

Logger32 User Manual

Version 4.0.352

August 2026

This manual has more than 1,000 pages and is frequently updated.

Do not print it out.

Save the planet. Hug a tree. Cuddle a whale.

If that ▲ was *ages* ago, scan this QRcode to grab the current edition from the Logger32 website ▶

Copyright © 2026 Bob Furzer K4CY

Clickable contents

Section A Logger32 basics.....15

1 Introduction 16
- 1.1 Using this User Manual *** START HERE *** 16
- 1.2 Logger32 functions, features and fun from A to Z 23
- 1.3 Basic Logger32 FAQs (Frequently Asked Questions) 26

2 Installing and updating Logger32 34
- 2.1 System requirements 34
- 2.2 Logger32 version 4 installation process 35
- 2.3 Initial configuration: six baby steps 35
- 2.4 Times and time zones 39
- 2.5 Bandplans: the Bands & Modes table 41
- 2.6 Logging bands and modes 47
- 2.7 *Upgrading* Logger32 from v3 to v4 49
- 2.8 Move an *existing* Logger32 installation to a *new* one 53
- 2.9 v4 installation and migration troubleshooting 55
- 2.10 Automated Logger32 v4.x updates 57
- 2.11 How to *un*install Logger32 68
- 2.12 Installation and updating FAQs 69

3 Importing and exporting logs and other data files 91
- 3.1 Importing ADIF logs 91
- 3.2 Exporting logs 95
- 3.3 Exporting other data files 100
- 3.4 Import-export FAQs 104

4 Getting started with Logger32 110
- 4.1 Screen layout 110
- 4.2 Main menu 111
- 4.3 Setup menu 120
- 4.4 The bottom panels 125
- 4.5 Getting started FAQs 129

5 Country, county and island databases.....132
- 5.1 Country and prefix database maintenance 132
- 5.2 Maintaining DXCC entities (layer A) 137
- 5.3 Alternative Prefixes (layer B) 138
- 5.4 Large Country Prefix Offsets (layer C) 139

Copyright © 2026 Robert C. Furzer K4CY

5.5 Recalculate statistics 139
5.6 Recalculate statistics *and* distances 142
5.7 The county database 143
5.8 The IOTA database 143
5.9 Database FAQs 145

6 The log entry pane 153
6.1 What we see in the log entry pane 153
6.2 Log entry pane right-click menus and functions 159
6.3 The ? (query or help) buttons 172
6.4 Logbook informational entries (pseudo-QSOs) 175
6.5 Database maintenance 178
6.6 Callsign and country information floater 179
6.7 Notification messages 180
6.8 Generic QSO pane 181
6.9 Find lost windows 183
6.10 Log entry pane FAQs 184

7 Scratchpad and notes 193
7.1 Notes 193
7.2 Scratchpad 193
7.3 Using the scratchpad 195
7.4 Log entry pane ⇄ scratchpad 201
7.5 Scratchpad and notes FAQs 202

8 The logbook 205
8.1 Logbooks and operators 205
8.2 Logbook right-click menus 210
8.3 Configure QSO tables (logbook and other 'grids') 219
8.4 QSO grid columns and layout 220
8.5 Sorting the logbook 224
8.6 Modifying/editing logged QSO information directly 225
8.7 Navigating the logbook 226
8.8 Rebuild (validate and optimize) your logbook 227
8.9 Changing and creating new logbooks 228
8.10 QSO numbering 230
8.11 Comments and notes 231
8.12 Updating QSO flags 232
8.13 Setup custom ADIF fields 233
8.14 Logbook FAQs 233

Copyright © 2026 Robert C. Furzer K4CY

Section B Logger32 intermediate topics.....246

9 Previous QSOs pane.....247
9.1 Configuring the previous QSOs pane 247
9.2 Selecting a QSO 248
9.3 Changing the appearance of the previous QSOs table 248
9.4 Grouping and sorting previous QSOs 248
9.5 Check/update QSLs 249
9.6 Previous QSOs pane FAQs 249

10 Worked/confirmed table 252
10.1 The power of worked/confirmed..... 252
10.2 Configuring the worked/confirmed table 254
10.3 Worked/confirmed table FAQs..... 262

11 Callsign lookups 265
11.1 Online callsign database lookups..... 265
11.2 CD-ROM callsign databases 269
11.3 Minimized display..... 271
11.4 Callsign lookup FAQs 271

12 Radio control (CAT).....273
12.1 Configure the CAT's COM port 274
12.2 Radio debug window..... 277
12.3 CAT control of radios (commands) 278
12.4 Controlling the radio's Push To Talk 279
12.5 CAT echo and slave ports..... 282
12.6 Interfacing examples 286
12.7 Command a slave receiver or transceiver..... 287
12.8 Use a cheap SDR RX as a panadapter..... 288
12.9 CAT FAQs..... 292

13 DX cluster 300
13.1 What *are* DX spots? 301
13.2 DX cluster pane appearance..... 303
13.3 TNC tab..... 304
13.4 Telnet tab 306
13.5 Messages tab..... 310
13.6 AGWpe tab..... 312
13.7 Localhost (and RBN) tab 312
13.8 Using Telnet to connect to other computers 315
13.9 Pseudo DX clusters with VE7CC's CC User software..... 315

Copyright © 2026 Robert C. Furzer K4CY

13.10 Pseudo DX clusters with Slice Master 6000 software 317
13.11 DX cluster FAQs 317

14 DX spots and BandMaps 319
14.1 DX spot processing overview 321
14.2 DX Spots pane right-click menu 322
14.3 BandMaps 345
14.4 DX/DXpedition station information 354
14.5 DX spot history 356
14.6 Posting (sending) DX spots to DX cluster 358
14.7 Adjust PSK/RTTY DX spots for audio frequency 360
14.8 *Pseudo* DX spots (bookmarks) 361
14.9 DX cluster and spots FAQs 362

15 The UDP BandMap and FT# 374
15.1 Open UDP BandMap 375
15.2 UDP BandMap menu 375
15.3 JTDX Control Panel 388
15.4 Opening & closing digimode software through Logger32 389
15.5 JTDX configuration settings 392
15.6 *Using* the UDP BandMap 395
15.7 RPTR (JTDX|WSJT-X UDP message repeater) 403
15.8 Sending DX spots 405
15.9 UDP message debug window 405
15.10 Event viewers 406
15.11 UDP port settings for GridTracker + Logger32 + JTDX 409
15.12 UDP logging from a Raspberry Pi 409
15.13 UDP BandMap FAQs 410

16 QSLing 426
16.1 Making QSL chores slightly less onerous 427
16.2 Responding to SWL cards received 428
16.3 Tracking confirmations received 430
16.4 Sending QSL cards and other QSO confirmations 430
16.5 Synchronize QSL (from an ADIF file) 437
16.6 QSLing FAQs 438

17 LoTW (Logbook of The World) 440
17.1 Identifying LoTW users 441
17.2 Upload your QSOs to LoTW 445
17.3 Download LoTW confirmations to update your log 447
17.4 Handling USACA county/LoTW discrepancies 464

Copyright © 2026 Robert C. Furzer K4CY

17.5 Correcting IOTA errors 467
17.6 Gridsquares from LoTW 467
17.7 General notes about LoTW synchronization 468
17.8 Diagnosing mismatched confirmations 468
17.9 Satellite LoTW 469
17.10 LoTW FAQs 470

18 Awards 473
18.1 Supported awards 473
18.2 Award tracking and reporting 480
18.3 Administering and customizing awards 485
18.4 Configuring the awards 496
18.5 DXCC awards 504
18.6 IOTA award 515
18.7 WAS award 522
18.8 CQ DX Field award 523
18.9 USA Counties Award 524
18.10 Awards FAQs 525

19 Tracking window (maps) 534
19.1 The tabs 535
19.2 The maps 536
19.3 Six tabs for six maps 540
19.4 Map FAQs 554

Section C Logger32 advanced topics 559

20 Add-on utilities 560
20.1 Pre-loaded utilities 560
20.2 Utility program setup 562
20.3 Ham CAP propagation prediction 564
20.4 NCDXF beacons 566
20.5 Grid squares (locators) 569
20.6 L32LogSync 571
20.7 Node-RED 571
20.8 CheckCall 572
20.9 JTAlert 573
20.10 N1MM+ Bridge 573
20.11 Add-on utilities FAQ 573

21 Sound card digital modes 574
21.1 Sound card data window panes 574

Copyright © 2026 Robert C. Furzer K4CY

21.2 Sound card data window menu 575
21.3 Sound card data window status bar 583
21.4 Digital communications modes 585
21.5 Selecting the soft modem 585
21.6 MMTTY 586
21.7 MMVARI 588
21.8 Using MMVARI/MMTTY with CAT 598
21.9 Receiving RTTY and PSK signals 600
21.10 Transmitting RTTY and PSK signals 605
21.11 Communicating with RTTY 613
21.12 Transmitting RTTY 619
21.13 Calibrating the sound card 624
21.14 Digital signal quality 627
21.15 2Tone 629
21.16 GRITTY 635
21.17 Sound card digimode FAQs 637

22 Data Terminal (TNC) 643
22.1 Data terminal setup 643
22.2 Mode macros 644
22.3 Data terminal logging 644
22.4 File transfer 644
22.5 Data terminal macros 645
22.6 KAM+ TNC 646

23 Macros 653
23.1 Macro strings and commands 653
23.2 Advanced macros 671
23.3 Macros FAQ 673

24 Programmable macro buttons and hotkeys 674
24.1 Programmable buttons and hotkeys 674
24.2 Hotkeys (keyboard shortcuts) 676
24.3 Logger32's default hotkeys 677
24.4 Macros and hotkeys FAQs 682

25 CW Machine 683
25.1 Connecting and configuring the CW Machine 684
25.2 Configuring and using the CW Machine 686
25.3 Using the CW Machine 696
25.4 Keying speed 697
25.5 CW Machine and CwGet (CW decoder) 698

Copyright © 2026 Robert C. Furzer K4CY

25.6 CW Machine macros and scripts 700
25.7 CW Machine FAQs 704

26 Digital Voice Keyer (DVK) 715
26.1 Setup the DVK 715
26.2 Setup radio DVK 716
26.3 Setup Logger32 DVK 716
26.4 Setup the *micro*HAM DVK 718
26.5 Using the DVK 719

27 Configurations 722
27.1 Defining and using a different configuration 723
27.2 Swapping configurations using a batch file 724
27.3 Editing Logger32's *.INI* files manually 726
27.4 Configuration FAQs 727

28 Casual contesting and special events 730
28.1 Contesting 730
28.2 Special events and callsigns 731
28.3 Contesting and special event FAQs 732

29 Printing 734
29.1 Using external print programs 734
29.2 Printing QSL labels using Microsoft Word 735
29.3 Printing award tables 736
29.4 Printing FAQs 739

30 Multi-language support 740
30.1 Using 'foreign' languages in Logger32 740
30.2 Generating (translating) language DLLs 741

31 Application Program Interfaces (APIs) 746
31.1 Establishing communication 747
31.2 Additional messages *to* Logger32 748
31.3 Additional messages *from* Logger32 750
31.4 ADIF text strings 751
31.5 Things to consider 754

32 The TCP Client-Server 755
32.1 About TCP 755
32.2 Configuring and monitoring TCP in Logger32 756
32.3 Sending ADI-formatted QSOs to other apps via TCP 759
32.4 Parallel logging via TCP 760

Copyright © 2026 Robert C. Furzer K4CY

32.5 TCP configuration for JTDX or other digimode software 766
32.6 Using JTAlert with Logger32 768
32.7 N1MM-Logger32 Bridge 774
32.8 TCP Server FAQs 777

33 UDP broadcasting of DX spots 781
33.1 Setup DX spot broadcasting 782
33.2 Using UDP broadcast of DX spots 782
33.3 N1MM+ protocol 786
33.4 How to create the SSDR CAT Flex Radio UDP Port 787

34 Computer security, performance and capacity 788
34.1 Windows security and privileges 788
34.2 Backups 789
34.3 Antivirus 796
34.4 Debugging 797
34.5 Patching and updating 798
34.6 Network security 799
34.7 Capacity 799
34.8 USB power capacity and quality 800
34.9 Computer tuneup 801
34.10 Security, performance and capacity FAQs 809

Section D Logger32 hardware interfacing 815

35 Shared ports 816
35.1 Port setup 816
35.2 Using VSPE to share serial ports 818
35.3 Logger32 and ARCP-590 or ARCP-590G (Splitter Device) 822
35.4 Logger32 and HDSDR (Splitter Device) 822
35.5 Logger32 and HDSDR (Connector Device) 823
35.6 LP-Bridge and LBP2 825
35.7 Logger32 and NaP3 (Splitter Device) 825
35.8 Logger32, NaP3 and LP-Bridge 827

36 Radio Control Panel (RCP) 829
36.1 Open the RCP 830
36.2 RCP menu 830
830
36.3 RCP (radio) macros 837
36.4 Automatic DX spot split operations 845
36.5 Alerts for split operation 850

Copyright © 2026 Robert C. Furzer K4CY

36.6 SO2R and SO2V RCP operation 852
36.7 RCP FAQs 856

37 Sound card-radio interfacing 859
37.1 Interfaces 859
37.2 Audio sources 864

38 Single Operator Two Radios (SO2R) 865
38.1 SO2R radio changeover 865
38.2 Quick switch 867
38.3 CW Machine 868
38.4 WinKeyer 868
38.5 Additional macros 869
38.6 Sound card enhancements for SO2R 870
38.7 Preliminary setup of MMTTY for SO2R 871
38.8 Sound card selection 872
38.9 Selection of audio channel 872
38.10 Radio PTT port selection 873
38.11 Setting up to use two radios 875
38.12 SO2R FAQs 877

39 Single Operator Two VFOs (SO2V) 878
39.1 Operating SO2V 878
39.2 Sound card SO2V 879

40 Open Two Radio Switching Protocol (OTRSP) 880
40.1 About OTRSP 881
40.2 OTRSP macro command format 881
40.3 OTRSP command examples 882
40.4 <Ctrl+T> and grave accent key switching 883

41 Antenna rotators (rotors) 885
41.1 Supported rotators 885
41.2 Which rotator for which band? 886
41.3 Rotator configuration 886
41.4 How to rotate the antenna 888
41.5 Linked rotators 894
41.6 *Micro*HAM ARCO controller 896

42 Antenna switching 899
42.1 Parallel port antenna switching 899
42.2 Serial port antenna switching 902
42.3 Default antenna 907

Copyright © 2026 Robert C. Furzer K4CY

42.4 Antenna names 907
42.5 CAT-controlled radio antenna switching 908
42.6 Testing the antenna switch 908
42.7 Antenna switching FAQ 908

43 Virtual SteppIR Controller (VSC) 909
43.1 VSC setup 910
43.2 Operating the VSC 913

44 Virtual UltraBeam Controller (VUC) 914
44.1 VUC setup 914

45 PK-232 916
45.1 Data terminal window setup 916
45.2 Mode macros 917

46 AGW Packet Engine 919
46.1 AGWpe configuration 919

47 Elecraft 924
47.1 Elecraft introduction 924
47.2 Elecraft K2 927
47.3 Elecraft K3/K3s 930
47.4 Elecraft K4 931
47.5 Elecraft KX3 933

48 ICOM 934
48.1 ICOM hex addresses 934
48.2 ICOM interfacing 935
48.3 ICOM filter setup 935
48.4 ICOM VFO B display 936
48.5 ICOM clock setting 937
48.6 ICOM CW and SSB bandwidth settings 938
48.7 ICOM radio “DATA” modes 939
48.8 ICOM radio settings for Logger32 939
48.9 ICOM IC-705 940
48.10 ICOM IC-706 Mk II 941
48.11 ICOM IC-735 941
48.12 ICOM IC-746 with OPC-662 interface 942
48.13 ICOM IC-7300 942
48.14 ICOM IC-7600 947
48.15 ICOM IC-7610 948
48.16 ICOM IC-9700 952

Copyright © 2026 Robert C. Furzer K4CY

49 Kenwood 953
49.1 Supported Kenwood radios 953
49.2 Kenwood CAT connections and commands 953
49.3 Kenwood TS-50, TS-60, TS-840, TS-940, TS-950SD and TS-950DX 958
49.4 Kenwood TS-440 958
49.5 Kenwood TS-480 959
49.6 Kenwood TS-570 and TS-870 959
49.7 Kenwood TS-590 961
49.8 Kenwood TS-890 973
49.9 Kenwood TS-990S 975
49.10 Kenwood TS-2000 976
49.11 Kenwood FAQs 977

50 TEN-TEC 978
50.1 TEN-TEC OMNI VI 978
50.2 TEN-TEC OMNI V.9 980
50.3 TEN-TEC OMNI VII 980
50.4 TEN-TEC ORION 981
50.5 TEN-TEC Paragon 585 982
50.6 TEN-TEC Paragon II 982

51 Yaesu 984
51.1 Supported Yaesu radios 984
51.2 Yaesu CAT connections 985
51.3 Yaesu CAT command protocols 986
51.4 Antenna selection for FTdx-5000, FTdx101D and FTdx101MP 989
51.5 Yaesu FT-100 990
51.6 Yaesu FT-450 & FT-450D 992
51.7 Yaesu FT-817 994
51.8 Yaesu FT-847 995
51.9 Yaesu FT-897 998
51.10 Yaesu FT-920 999
51.11 Yaesu FT-990 1002
51.12 Yaesu FT-991A and FT-891 1005
51.13 Yaesu FT-1000MP 1006
51.14 Yaesu FT-2000 and FT-2000d 1008
51.15 Yaesu FTdx-5000 1009
51.16 Yaesu FT-101D (FTdx101D) and FT-101MP (FTdx101MP) 1013
51.17 Yaesu FTdx10 1017

52 Software Defined Radios (SDRs) 1018
52.1 Pegasus 1018

Copyright © 2026 Robert C. Furzer K4CY

52.2 Power SDR 1019
52.3 Flex 6000 series 1023

53 Alpha 87A linear amplifier 1027

54 *micro*HAM MK2R+ 1030
54.1 Set up Logger32 for the *micro*HAM MK2R+ 1030
54.2 Set up the MK2R+ 1031
54.3 *micro*HAM macro examples 1032
54.4 Interfacing Logger32 to a *micro*HAM Keyer and FT-1000MP 1033

55 *Programmable* USB VFO knobs 1040
55.1 Reprogramming the Drok 1041
55.2 Configure Logger32 to use the Drok 1044
55.3 Test your Drok programming skills 1044
55.4 USB VFO knob FAQ 1045

56 Bits on the bottom 1047
56.1 The Logger32 team, a global community of volunteers 1047
56.2 Acknowledgments 1050
56.3 Glossary and abbreviations 1050
56.4 Additional information online 1056
56.5 Logger32's history 1059
56.6 Logger32 copyright, licensing, terms and conditions 1059
56.7 Logger32 guarantee and loyalty scheme 1061

Appendix: Logger32 color config cheat sheet 1062

Hinson tip: click whatever catches your eye in the contents listing to go there. There are more tips on navigating and searching this manual here (go on, click it: you know you want to …).

Copyright © 2026 Robert C. Furzer K4CY

Recent changes to this User Manual and Logger32

Version	Date	Changes
4.0.352	August 2026	Clarified (or at least amended) the description of log entry pane control-key shortcuts.
4.0.350 +.351	July 2026	Updated the FAQ on Logger32's warning about missing satellite data in the tracking pane. We can now do gridsquare/locator lookups.
4.0.349	June 2026	Several changes made for satellite tracking (*e.g.* get TLEs in DOS or Unix format, rename satellites and run radio macros when changing satellites) and logging sat QSOs.
4.0.348	May 2026	De-duping function can identify and remove duplicate QSOs from your log. Manual un-updated (downdated?) due to having other priorities.
4.0.347	April 2026	Refreshed the instructions for creating and using non-English language files. See when interesting DX spots were sent automatically to the Scratchpad while you were otherwise engaged. Added a note about Yaesu radio names.
4.0.346	March 10 2026	Mansplained Ham CAP. More support for FT2 *e.g.* known DXpeditioners CQing on FT# (FT8, FT4 or FT2) can be highlit on the UDP BandMap. We can now configure the UDP port broadcasting CAT data from the radio onto the shack LAN.
4.0.345	March 1 2026	New DX cluster Telnet connection progress messages. Limited support (so far) for FT2 mode. DXpedition stations can be highlit on UDP BandMap. Brief note about using Node-RED for shack automation. Added FT991 and FT891 CAT info. Current SSN and Kp values from WCY messages on DX Cluster are now passed to Ham CAP.
4.0.344	February 2026	New TCP client-server functionality added, with the capability to send ADI-formatted QSO data to another system or app as QSOs are logged. Flagging QSLs revived. Revised instructions for configuring the worked/confirmed table. Added a note about the value of DX cluster server-side spot filtering. Support added for programmable USB volume/VFO knobs. Spruced-up the glossary. Made tiny changes to the CW Machine chapter. Linked to the API programming VB example file. Ruthlessly hunted-down and eliminated *dozens* of broken internal hyperlinks, having multiplied like fleas on a feral CAT.

Hinson tip: this manual is revised whenever Logger32 is changed but, due to its bloat, it is *not* included in the program auto-updates. Download the current PDF manually to catch up with the latest, thrilling episode. Spare the planet: don't print it out unnecessarily. It is designed as an eBook with *lots* of hyperlinks and easy searching using <**Ctrl**+**F**> to **F**ind stuff in PDF reader apps.

Copyright © 2026 Robert C. Furzer K4CY

Section A

Logger32 basics

Copyright © 2026 Robert C. Furzer K4CY

1 Introduction

> "Every new beginning comes from some other beginning's end"
>
> *Seneca*

Welcome to Logger32, a 32-bit PC program written by Bob Furzer, K4CY. More than just a logbook, Logger32 is a **comprehensive shack operating system for radio amateurs**.

At the present time, Logger32 is available free-of-charge but it is *not* in the public domain. Legal information can be found in the copyright and licensing section at the end of this manual.

1.1 Using this User Manual *** START HERE ***

This user manual/help file has a progressive structure with four sections[1]:

- Section A covers the basics of installing and configuring the software – a gentle introduction to the main windows and core functions that most Logger32 users utilize.
- Section B leads through intermediate stuff to more advanced topics in …
- Section C, such as data communications via digital modes, macros and specialized automation functions.
- Section D concerns radio/shack hardware – various transceivers and peripherals that can be connected to your PC and controlled through Logger32, plus end notes including a glossary for the specialist terms and abbreviations used throughout.

If you are looking for general guidance on something, the contents listing is a good place to start. Find the chapter or section that seems most relevant, click it and see where it leads. There are underlined hyperlinks to related sections sprinkled *liberally* throughout the entire manual.

Bring up the relevant menus in Logger32 and try out the functions while reading the manual. Configuration options and settings are explained in the same sequence as they are listed in the menus, on the whole: I meant to explain every item on every menu. I may have missed some.

If you need help with something specific in Logger32, the search function in your PDF reader is useful. Open the search bar using the magnifying glass icon (if available) or <**Ctrl**+**F**> (while pressing down the Ctrl or Control key on your keyboard, tap **F** for **F**ind), then type in the word or phrase you're looking for – a puzzling menu item for instance, or a function, or your radio model[2], or

[1] Although it is not recommended, if you were to read the entire >1,000-page manual cover-to-cover, studying it like a textbook and trying things out along the way, you would become an expert in Logger32. We've done that so you don't need to! Having said that, there are doubtless obscure nooks and crannies that remain unexplored, so if you discover anything interesting (*e.g.* errors or omissions in the manual, or even better ways to configure and use Logger32), *please* share your tips via the Logger32 reflector.

[2] We *usually* include the hyphen if it is part of the 'official' radio name: try searching with and without it.

Copyright © 2026 Robert C. Furzer K4CY

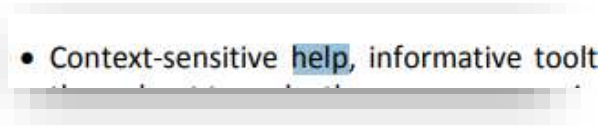
whatever. Try searching for the specific words or phrases used in Logger32 *e.g.* the wording of menu options that puzzle you.



◀ The PDF reader in, say, Google Chrome shows how many times the specific search term occurs.

The search is refined further as you type each character, so if there are *dozens* of hits (too many to browse easily), carry on typing a more specific phrase ▶





◀ The search term is highlighted with a colored background wherever it occurs in the text.

Click the up or down arrowheads in the search bar to go directly to the previous or next occurrence, or drag the slider on the righthand side of the window to any of the yellow bars where the term occurs bearing in mind the progressive nature of this manual: you'll generally find simpler, more basic explanations near the top, more advanced stuff in the middle and hardware-specific details towards the bottom ▶

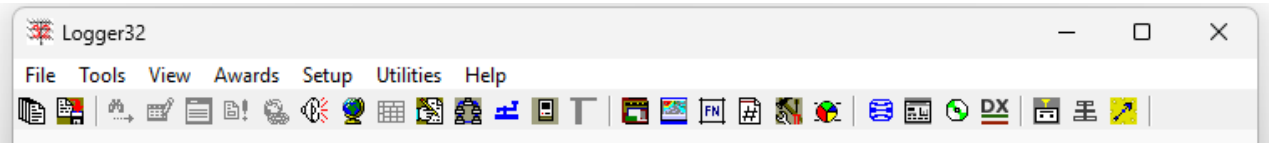


1.1.1 Typographical conventions in this manual

Menu items are shown in **bold** with ⇨ arrows linking to submenus *e.g.* **Setup ⇨ Radio** means click the word "Setup" on the menu to open a submenu, then click "Radio" on the submenu.

Buttons and individual menu items/options are usually shown in bold between angle brackets <**Like this**> to make them stand out from the text. The **bold** text is determined by Logger32, complete with its charming little quirks such as Inconsistent CapitaliZation. This means you can usually find additional information in the manual about obscure menu options by searching for the specific phrases that appear in those menus.

'Menu' usually means the main menu near the top of Logger32's main screen, with clickable words to open 7 subsidiary menus: **File Tools View Awards Setup Utilities Help** ▼



The main menu sits just above the **toolbar**, a row of 28 little pictorial icon buttons[3] ▲



◀ Mouseover the toolbar icons to see the number and name of each function - such as "#15 BandMap" for the set-square icon.

I have drawn red boxes on some of the screenshots, sometimes with **big bold red letters** or, in a few cases, brightly-colored highlighter pen marks to point out specific areas of the screen, plus red text for other important stuff. Generally, anything picked out in red is either a warning or something to take note of, especially the warning flags.

[3] If you really don't like them, hide them all by clicking to un-tick **View ⇨ Show toolbar**.

Copyright © 2026 Robert C. Furzer K4CY

By the way:

- **Mouseover** means point the mouse cursor at something and hesitate a moment.
- **Tooltip** is the mostly yellow box that magically appears (pops-up) with additional information.
- **Click** means tap the mouse button on the left[4].
- **Right-click** means tap the mouse button on the right[5].

I use **bold** for configuration items, options and menu items, usually surrounding them with angle brackets to make them stand out from the text, like this <**Band from bandplan**>. Individual bold letters pick out abbreviations (*e.g.* **C**ontinuous **W**ave = **CW**) and keyboard keys (*e.g.* <Alt-**X**>).

By the way, clicking <**Setup>** on the main menu swaps it for the longer setup menu[6] ▼



Click <**Main menu>** at the right-hand end of the <**Setup**> menu to revert to … the main menu. There are similar textual menus on other windows such as the CW Machine ▼



Hopefully, the context implies which menu I mean. Look at nearby images for further clues.

Folder[7] and file names are shown in *italics e.g.*[8] *C:\Logger32\Logger32.INI* Windows permits embedded spaces within folder and file names, so if the italics continue beyond a space, that is part of the same folder/file name. *"C:\Logger32\Export files"*, for instance, is the full name of the default file export folder (without the quotation marks). A trailing backslash is optional in Windows so "*C:\Logger32\Export files******"* works just the same.

File and folder operations are mostly performed through File **E**xplorer (formerly called Windows **E**xplorer). Open File **E**xplorer using any of these three methods:

- Click the Browse ... buttons provided in several Logger32 functions.
- <**Win**+**E**> *i.e.* press and hold the key to the left of the space bar ▶ (usually) with the Windows logo and tap the **E** key – that's **E** for **E**xplorer.



- ◀ Click the Windows <**Start**> icon at the bottom left or center of the screen, or the magnifying glass if shown[9], or else tap the Windows key then type **file explorer <Enter>**. The Windows search is case-insensitive (CAPITALS and lower-case letters are equivalent). Click the whatever you wanted from the list generated by Windows (assuming it's there!).

---

[4] Trust me, although it is more explicit, "left-click" would soon become annoying since it is used so often.

[5] Logger32 does not use other mouse buttons but the scroll wheel works to scroll some tables/lists, such as the logbook. Oh and lefties may swap left ⇆ right mouse buttons in their mouse software.

[6] There are similar toolbars within some windows *e.g.* MMTTY and MMVARI.

[7] Folders are also known as directories. They still are, in my little head. I'm old-skool.

[8] Abbreviated Latin terms are italicized as well. Numbered footnotes like this provide additional information and tips.

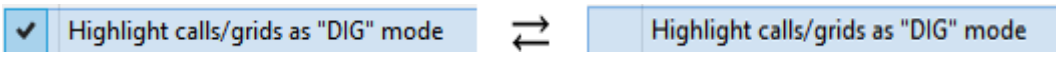
[9] These vary. The magnifying glass can be shown or hidden under taskbar options.

Copyright © 2026 Robert C. Furzer K4CY

Some messages and menu items vary according to the context. The variable bits are denoted in this manual by square brackets *e.g*. the DX spot configuration menu item shown as <**Save [**band**] settings**> may in fact say **Save 160m settings** or **Save 80m settings** *etc*. according to which band's DX spot filters are being configured at the time.

Some menu items and options are duplicates except for a specific parameter. To avoid needless repetition, I use the vertical bar |[10] to separate alternatives *e.g.* **Setup ⇨ Radio ⇨ Radio 1|2 configuration** refers to either of two menu items: **Setup ⇨ Radio ⇨ Radio 1 configuration** or **Setup ⇨ Radio ⇨ Radio 2 configuration.** And the tilde **~** means 'roughly', more-or-less.

"Select", "enable", "tick" and "check" are largely synonymous, referring to items such as:



Point the mouse cursor then click either one to change it to the other.
It is a kind of "toggle" or "flip-flop".

Solid triangles ◀ ▲ ▼ ▶ point from the text to relevant images nearby - mostly screenshots from my Logger32 installations on Windows 10 or 11 PCs … but be aware that your PC and its configuration is unique so don't be surprised if you see something a little different. For example, your Windows settings for the colors and fonts used in the captions (window title bars) and menus probably differ from mine, and various menu options concern the specific PC or radio hardware you are using (*e.g*. port numbers and radio models).

Hyperlinks are underlined.[11] Click them to jump directly to relevant sections in this manual, usually, or to Internet websites. Depending on your PDF reader software, the links may change color after you have clicked them, and you may have a "Back" button/arrow, alt-right-arrow or "Previous view" right-click option, or some other cunning means to retrace your steps if you get distracted or lost and bewildered. Check your Acrobat reader help for … help.

Hinson tip: since the screenshots throughout this User Manual are taken from Logger32 installations used by experienced Logger32 users on the support/beta team, they show the settings we are using routinely. Would they work for you too? Try them to find out for yourself. If you find something even better, please share your tips through the Logger32 reflector.

Hinson tip: suggestions like this are scattered liberally throughout the document, not all from that Hinson bloke. There are also FAQs at the ends of most chapters. Please forgive my clumsy attempts at humor – I'll settle for the occasional wry smile.

Quotations look like this – mostly apt comments from users, Bob K4CY and the support crew, shared through the Logger32 reflector. More are welcome!

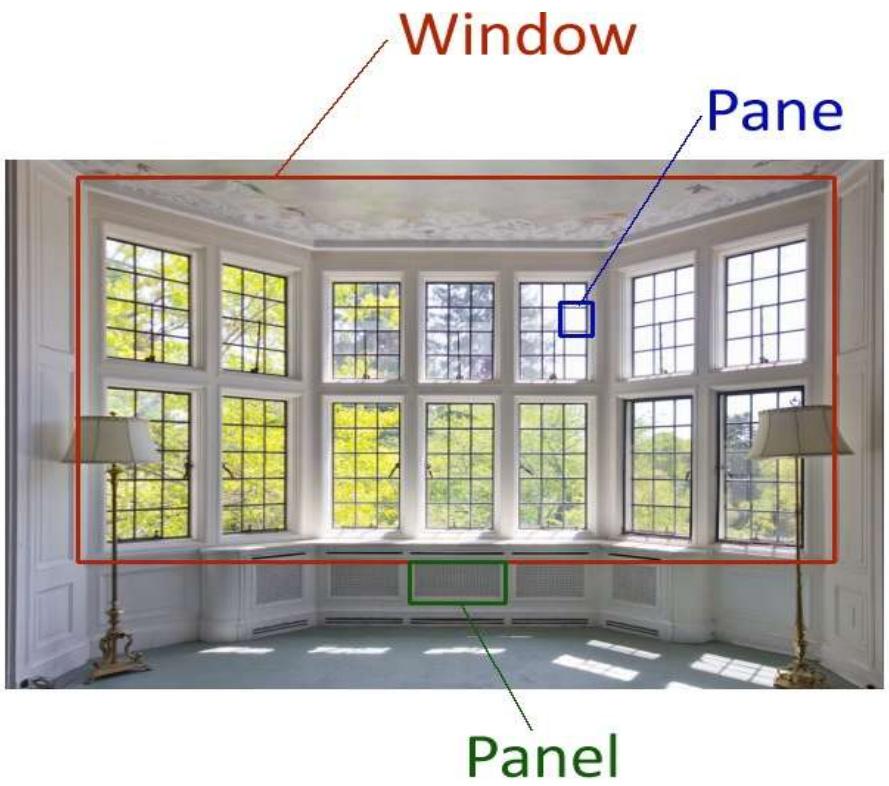
Gary ZL2iFB

[10] The vertical bar character | may be typed using <**Shift**+****> on most keyboards … but keyboards differ. It is rarely needed anyway. Personally, I prefer the kinds of bars propped up by 'characters' drinking beer.

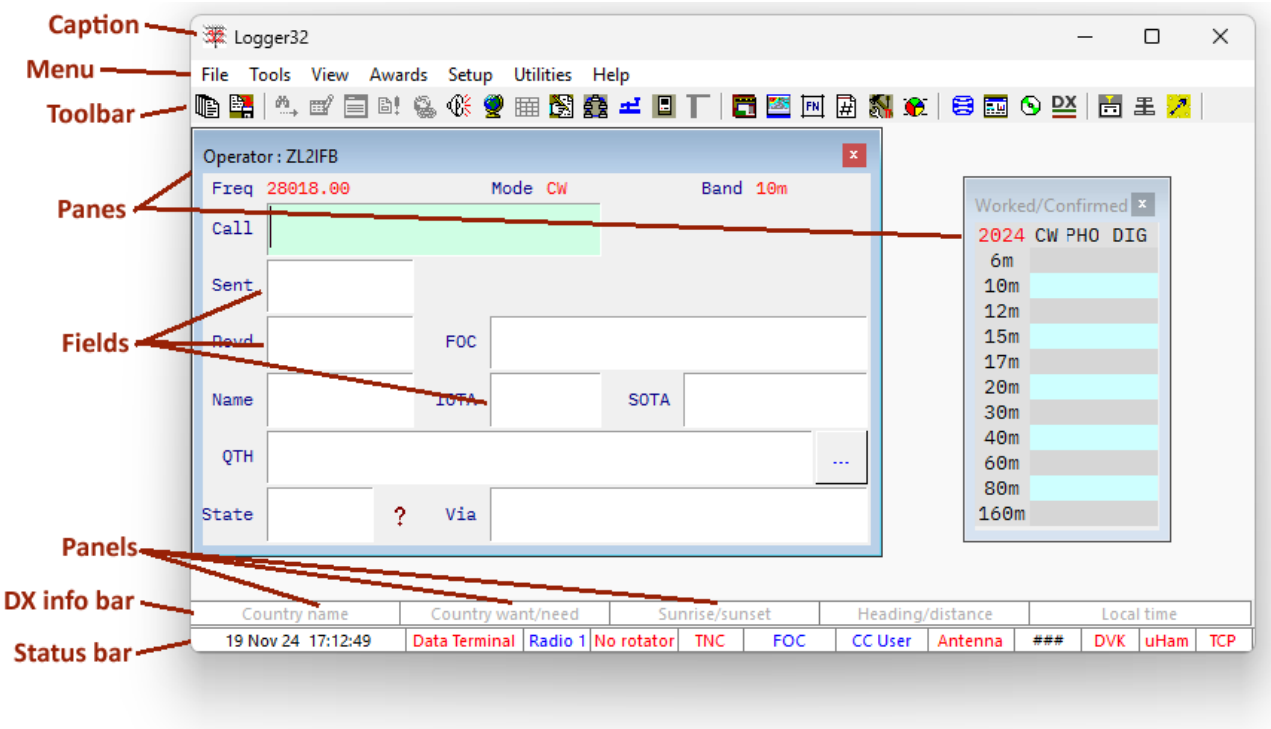
[11] All except the contents listing: MS Word doesn't underline those hyperlinks but they still work.

Copyright © 2026 Robert C. Furzer K4CY

1.1.2 Parts of the Logger32 screen



◀ The manual uses these architecturally-inspired terms to refer to various areas or elements of Logger32's display[12].



Hinson tip: the status bar has 15 useful panels ▲ most of which are right-clickable.

[12] Some little windows or panes are called tables or grids *e.g.* the worked/confirmed table.

Copyright © 2026 Robert C. Furzer K4CY

1.1.3 Seeking additional help

Aside from this User Manual, Logger32 is largely self-documented in the form of context-sensitive tooltips, plus error and warning messages that catch common issues and explain what's wrong. These are quite succinct, verging on cryptic at times, but they require virtually no effort to see.

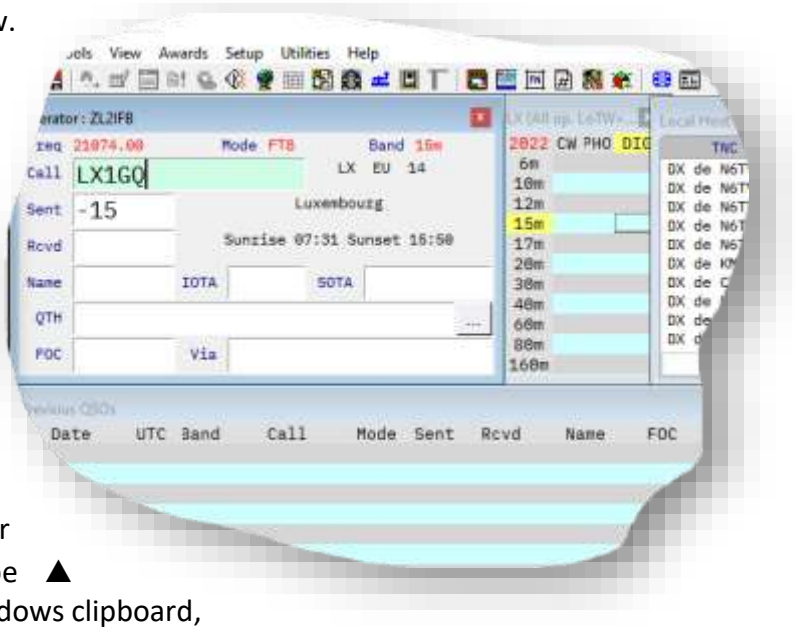
If something doesn't make sense, or you can't figure out how to achieve something, you are *very* welcome to email a query to the Logger32 reflector. Here's how …

It helps us help you if you take the trouble to explain:

- What it is that you are after, if anything. What are you trying to achieve? What is causing your problems?
- Why are you trying to do whatever it is? What is your situation?
- What have you tried already? What actually happened, as opposed to what you *expected* or *hoped* would happen?
- Have you browsed and searched this manual for guidance? What is it lacking or failing to explain?
- Are there any messages from Logger32 with specific information?

Screenshots are often the best way to show us exactly what you see:

1. Press <**Alt**+**PrtScn**> (hold down the Alt key and tap the 'print screen' key, often labeled PrtScn) to grab an image of the in-focus window. You may need to click the relevant Logger32 window first to get it 'in-focus'. If you need to show several windows in one shot, press <**PrtScn**> without <**Alt**> to grab an image of the entire screen. To grab a shot of a menu or window that disappears at the vital moment, try <**Ctrl**+**PrtScn**>, or better still launch the "**S**nipping tool" using <**Windows**+**shift**+**S**> and choose to capture the window, a rectangular section of the **S**creen or if you prefer, an irregular **S**hape ▲ The captured image is held in the Windows clipboard, ready for you to paste it somewhere …



2. If possible, paste the image into your favorite image editor to check and edit it *e.g.*
   - Crop or blank out irrelevant parts (*e.g.* anything personal/sensitive, such as passwords).
   - Use the paintbrush, pencil, crayon or draw tools to point out, emphasize or draw rings around things in a bright/contrasting color.
   - Perhaps show us what you *expected* to see as opposed to what you actually saw.
3. Compose an email to the Logger32 reflector – preferably an HTML email so you can paste and embed the image/s within your message text that explains your issue. If you can only send plain text email, please attach the annotated screenshot/s.

Copyright © 2026 Robert C. Furzer K4CY

4. Wait for the Logger32 support crew and other Logger32 users to respond, following up with answers to any requests for clarification.
5. If someone suggests doing something specific, **read and follow their instructions carefully**, line-by-line, as best you can[13]. If you get stuck or a certain step doesn't work, tell us where you are up to and what happened so far. If you cannot be bothered to follow instructions to diagnose and fix things, don't expect much help.
6. When your issue is resolved, *please* take a moment to tell us about the fix in a closing email. It's a nice way to thank those who responded, confirm that it was resolved, and inform others who may have been watching the exchange.

Hinson tip: occasional requests/comments fall between the cracks and fail to elicit a response. It's OK to chase up after a while (maybe a couple of days) if you hear nothing, ideally providing additional information or explanation. **Our lack of response is usually because we don't understand your issue.** Complex technical matters may take a while longer to address too. Please be patient with us. Above all, please remember that Logger32 is a free program used daily by thousands of hams from most of the DXCC list. We are volunteers with other things on the go – logging our own QSOs for instance! We're not rudely ignoring you, honest. Be cool.

In many Logger32 windows, click and drag the mouse to highlight some text of interest. When you release the mouse button, the highlighted text is saved to the clipboard. Then paste it wherever you like.

Bob K4CY

Suggestions for new functions/features or changes are always considered and may be discussed, but for various reasons they are not always adopted. To stand the best chance, think and write carefully about what you are trying to do, and how it might work. Alternative solutions may be suggested. Have you tried other approaches and found them lacking?

If you are asking for support for a new device (such as a new model of radio), can you direct us to an online technical manual with details about interfacing and commands for that model?

Hinson tip: aside from support queries, problems and complaints, it is gratifying to receive positive feedback from Logger32 users about the program, the add-on utilities, maybe even the documentation. Which bits work best for you? Have you any tips or suggestions for other users you'd like to share? What do you wish you had discovered earlier? Have you anything positive to add to other comments and responses to queries? Logger32 is a community project. What are you waiting for? Email the Logger32 reflector!

[13] It can be *very* frustrating when users fail to follow simple instructions – generally because they are skim reading and/or too anxious to leap directly to a solution. *Please* spare a thought for the people offering you advice: we don't owe you anything. We invest our time and effort for free. If you are rude or give us a hard time, we'll probably just give up and leave you to it. Life's too short for inconsiderate users.

Copyright © 2026 Robert C. Furzer K4CY

Gary, in the section 'Set your PC clock', it might be worthwhile to mention that Logger32 must be 'Run as administrator' in order for the Get atomic clock time right-click menu pick to function as described. This requirement is mentioned much later in the Windows security section, but the user would need to possess an extraordinary amount of ESP to know this in advance.

***Ed N4II** (Ed: thanks)*

1.1.4 Smiles welcome

Although there is no charge for Logger32, if you are keen for the Logger32 team to continue our efforts, compliments are most welcome. We love hearing about what Logger32 enables you to do, and about the creative ways you have found to use the software in support of your operating activities. Make us smile! Help spread the word on social media that Logger32 is the mutt's nuts. Tell your pals at the club all about it. Mention it on your QRZ page. Hire an aerial signwriter …

1.2 Logger32 functions, features and fun from A to Z

Logger32 is unique among loggers, functionally rich, *replete* with facilities requested, designed, tested, used, abused, misused and yet much appreciated by amateurs. With Logger32, the answer is probably 'Yes' … so what's the question?

- ADIF compliance – import and export your log in ADIF format.
- Audio alerting and visual highlighting for various new ones, DX cluster talkies *etc*.
- **A**utomatic antenna selection for the current VFO frequency/band on the active radio.
- Automatic log-on scripts for DX clusters.
- Automation of many routine functions and activities *e.g.* uploading QSOs to online logs, looking-up and downloading DX station information from online callsign databases.
- **B**andMaps displaying spotted, received or bookmarked DX callsigns by frequency across the spectrum of each band. BandMaps can be customized for regional, national and personal preferences *e.g.* showing only *DX* spots, optionally highlighting 'new ones' and LoTW users *etc*.
- Built-in **D**igital **V**oice **K**eyer (DVK) capable of playing .WAV files from the PC or triggering the radio's own DVK.
- **C**alculate and display sunrise/sunset times, short path distances, long and short path beam headings, and local times for the DX stations as we are working and logging them.
- Callsign lookups to find out about the hams we are working while we are in QSO.
- Capacity for more than a million QSOs per logbook, even with *copious* notes.
- CAT control of **C**omputer **A**ided **T**ransceivers, with CAT comms monitoring and programmable macro buttons.
- Cherry picker can tail-end FT# QSOs in JTDX and/or respond to CQs.
- Clock-setting macros for some Kenwood and ICOM radios.

Copyright © 2026 Robert C. Furzer K4CY

- Column-wise sorting of the logbook and other forms.
- Comprehensive QSO statistics and record tables for QSLing and awards tracking.
- Configurable colors, text sizes and fonts – *lots* of options, so many in fact that we compiled a cheat sheet.
- Configurable, dynamically populated table showing the bands and modes on which a DXCC entity has been worked or confirmed: either for all time or just so far this UTC calendar year.
- Consolidated statistics for digital modes and phone modes, as well as individual mode/sub-mode stats.
- Contest serial number generator for up to 999,999 contacts[14].
- Context-sensitive help, informative tooltip pop-ups, boilerplate text and other useful hints throughout, making the program intuitive and easy to use, yet powerful and flexible.
- **D**ata Terminal for TNCs with programmable, dynamic memories and macros.
- Definable Telnet/DX cluster connection scripts and shortcuts.
- DX cluster monitoring, merging inputs from multiple feeds and alerting for 'new ones'.
- DX spot maps with various projections and pop-up information on spotted callsigns.
- DXCC support with statistics for worked, confirmed and verified QSOs, LoTW synchronization and a facility to export relevant QSO data in ADIF format for QSL card submissions.
- DXpedition spot highlighting for known DXpeditions, linked to web pages with more info.
- **E**ditable DXCC entity (*e.g.* country), primary administration (*e.g.* state), secondary admin (*e.g.* county) and IOTA databases.
- e**X**tensions/add-ons from third parties for various additional features and functions (*e.g.* sending QSOs to Club Log as we log them, or highlighting colleagues from the club when they are spotted on-air).
- **F**ull ADIF compatibility maintained at the *current* release level for log imports *and* exports.
- **G**rayline maps with map projection and terminator options.
- Grid square (locator) support for directions, distances and Battleships-style DX grid awards.
- **H**ighlighting of logged QSOs with QSLs waiting to be sent, sent and received, plus those **W**orked, **C**onfirmed, **S**ubmitted and **G**ranted for various awards.
- **I**ntegration of, or interoperability with, digital mode software such as JTDX, WSJT-X, MMVARI and MMTTY, and (with an add-on) the contest logger N1MM+. Log your digimode or contest QSOs automatically and effortlessly in your consolidated station log.
- Intelligent split operating on RTTY using differential audio tones or CAT control, with macros.
- **J**elly and cakes whenever we bag a new one (in-game purchase, not available in all states).
- JTDX Control Panel: while JTDX has exclusive CAT control of your radio, click Logger32 buttons to change bands *etc*.
- **K**eyer built-in with programmable CW memories, keyboard sending and macros, and support for WinKeyers.
- Logbook 'percentage full' capacity indicator, QSO count and other logbook statistics.

[14] Wow, that's some contest!

Copyright © 2026 Robert C. Furzer K4CY

- **L**ogbook and previous QSOs pane with up to 47 user-configurable columns for all the usual QSO info plus notes, IOTA references, grid squares, satellite names, Ten-Ten numbers *etc*.
- Logging of QSOs, naturally! We can also insert, edit/amend and delete QSOs *etc*.
- **M**ulti-language support for translated menu items/options and messages.
- Multiple program configurations for everyday DXing, casual contesting, SWLing, training *etc*.
- ***N****umerous* bug fixes, usability improvements and tips to make Logger32 *even better*.
- **O**ptional data-entry fields to record IOTA references, QSL routes, club membership numbers *etc*. when logging QSOs.
- Output your log data in ADIF, UQF and CSV formats.
- **P**air-up two Logger32 PCs through the network, both logging your QSOs in parallel. Send logged QSOs to a backup system or to your QSL manager in real time.
- Pop-up list previewing previously-logged callsigns similar to the one currently being logged.
- Programmable keys to send radio commands and macros.
- **Q**SLing support for several QSO confirmation mechanisms (LoTW, QSL cards, eQSL, Club Log and more).
- QSO serial numbers added automatically, and can be updated on demand *e.g*. if QSOs are added to or removed from the log (as well as serial numbers used in contests).
- Quick Switch, a function that automatically launches mode-related software such as JTDX or the CW Machine, triggers macros and sends CAT commands to the radio when you change modes.
- **R**apidly search the logbook for QSOs of interest as well as automatically displaying previously logged QSOs with the same station while we are logging a new QSO.
- Rebuild logbook function to validate (check for) and correct database integrity issues.
- Re-sizable windows/panes and the capability to retrieve lost windows/panes *e.g.* when screen resolution is modified.
- Rotator control with *sensible* auto-pointing and 'park' (go home) capabilities.
- **S**cratchpad to jot down info from QSOs in progress, with the ability to recall jotted callsigns, frequencies and modes with a click.
- Selectable sound card waterfall and spectrum display characteristics (color, brightness, smoothing) and frequency markers.
- Selective filtering and de-duping of DX spots, including Skimmer/RBN spots flowing fast.
- Send and receive DX spots, announce and talk messages to DX cluster via various channels.
- Single-button compression and saving of backups of your log and configuration data.
- SO2V and SO2R support for **S**ingle **O**perators using **2 V**FOs or **R**adios, plus OTRSP.
- Split reminders with configurable logging of both RX and TX frequencies, even with SO2V and SO2R.
- Support for a second CAT-controlled radio with CAT echo and slave ports: add an external receiver to your main transceiver or turn a cheap SDR dongle into a panadapter.
- Support for Ham CAP propagation prediction software by Alex VE3NEA.
- Support for on-line HamQTH and HamCall callsign lookups built in, plus QRZ and other callsign lookups using optional add-on third-party utilities.

Copyright © 2026 Robert C. Furzer K4CY

- Support for single-user logs, of course, plus multi-user and multi-callsign logs (*e.g.* shared club or family stations; regular and contest callsigns; home, portable and mobile operations; DX station logs for QSL managers …).
- Support for SteppIR (VSC) and UltraBeam (VUC) antenna controllers.
- Support for add-on apps using external interfaces (APIs).
- Synchronization of downloaded LoTW and eQSL confirmations with the log, updating your award statistics accordingly for DXCC, IOTA and other awards.
- Synchronize your computer's clock to an online atomic clock reference using SNTP.
- **T**rack and map satellites in real time, showing AOS/LOS, azimuth, elevation *etc*.
- **U**nit conversion utility for temperatures, lengths *etc*.
- User-configurable column display, sequence and naming for the logbook and other QSO tables.
- User-definable needed/worked/confirmed colors for DX spots and digimode decodes.
- User-definable carry-forward of selected information fields from previously-logged QSOs with the same stations, even when they operate portable, mobile or overseas under CEPT.
- User-selectable VFO frequency displays in kHz or MHz down to 1 Hz resolution.
- UTC date and timestamps, with UTC, home shack and DX shack clocks calculated and displayed simultaneously (in real time!), using configurable date and time display formats.
- **V**alue for money: an *outstandingly* useful and cool bit of amateur radio software, with global support and this comprehensive User Manual, all supplied, maintained and supported **free of charge** … with a no-questions-asked cast-iron instant full money-back guarantee ☺
- **W**arnings and error messages that make sense, on the whole. Well, we try.
- **X**tremely configurable. Xtraordinarily fiddlable.
- **Y**tterbium-free. No lanthanides at all, in fact no added chemicals. Low fat, low cholesterol. Not tested on animals[15]. Depends on *natural* intelligence. Will run (albeit slowly) on a solar- or wind-powered knit-your-own organic computer of the appropriate specification for zero emissions. Eco-friendly: *please* save the forest by browsing, reading and searching this manual on-screen rather than fumbling through at least a ream of printed paper.
- **Z**ero *severe* flaws or bugs[16].

1.3 Basic Logger32 FAQs (Frequently Asked Questions)

Most chapters in this manual conclude with FAQs concerning issues that crop up repeatedly on the Logger32 reflector or in conversations between the development, beta test and support crew with hams struggling with various aspects of Logger32. It *is* a complex application, with a huge number of configuration options and some advanced functions that are tricky to recall and understand (even for those who designed and coded them, eh Bob?).

Hopefully the FAQs address/forestall the most common issues. Don't forget to search the text.

[15] Except for radio amateurs.

[16] Real-world software is never perfect, especially a program as complex as a real-time logging system with all the above features. However, the programmers, beta testers and users have done, and continue to do, a *fantastic* job with Logger32. **Bug reports are welcome via the Logger32 reflector.** Help us edge Logger32 ever closer to absolute, glorious perfection!

Copyright © 2026 Robert C. Furzer K4CY

Q. I'd like to do [something] but how? I've forgotten …

A. A consequence of Logger32's extreme flexibility is that there are *lots* of functions, configuration settings and options. Lots and lots. Loads, shack-loads in fact. Part of the joy of using (and supporting and documenting!) Logger32 is finding cool new options, sometimes discovering things you didn't even know you wanted until you accidentally tripped over them.

Fiddling around with Logger32's screens, colors and settings give you something to do in the shack when conditions are flat and there's not much DX about.

All that complexity means you'd need to be a memory-freak[17] to remember what every last option does, and more importantly where to find and configure it.

If you're lost and bewildered, **try right-clicking the window or pane/part of the window that you'd like to change or reconfigure**. All of Logger32's windows have right-click menus to access configuration options, sometimes more than one (right-click in different places such as in data-entry fields or on the field labels and headings). Even small items like the digital clock have right-click menus with options for you to fiddle with. If that doesn't work, hunt for a <**Config**> or <**Appearance**> menu item.

Also, of course, you can browse or search this very manual for clues. Try <**Ctrl**+**F**> to **F**ind stuff. We have incorporated the exact text of most items on most menus and forms for instance, including any Spurious Capitals *etc.*, so if you're confused by something in Logger32, search the manual for the actual wording used. The contents listing is another useful starting point.

Lastly feel free to ask on the Logger32 reflector. You'll almost certainly get an answer, possibly something helpful and correct … and we may have another FAQ to add to this manual.

Q. What happened to Logger32 v3's 'Help' function?

A. It was replaced by this User Manual for Logger32 v4. The v3 help didn't simply disappear though: it was used as the basis for this v4 manual. 1,300 pages of v3 help content had been built up piecemeal over many years by a diverse team of Logger32 beta testers and other contributors … but there were significant structural, stylistic, readability and usability issues, hence it desperately needed revising and updating. So I (Gary, ZL2iFB) foolishly volunteered.

Going forward, I would really appreciate your help:

- If you find things in the manual that are plain wrong, *please* let me know. Factual corrections are an obvious priority, especially anything important of course.
- If there are parts that you find confusing, don't make sense, don't read well, or could be better expressed, *please* let me know. Aside from helping you achieve whatever it is that you're trying to do, I will gladly take another look at the relevant sections.
- If there are gaps – things that don't seem to be covered at all, at least not in sufficient detail to be helpful and useful – *please* let me know. Likewise dupes (duplicated text) and any broken links ▶



- If you would like to contribute content, tips, better ways of doing things, clearer screen images, cooler diagrams, shortcuts *etc.*, comments and asides, *please* let me know.

[17] If you *think* you know Logger32 inside-out, explore *C:\Logger32\Logger32.INI* and the other *.INI*s for obscure config settings you didn't even appreciate Logger32 was dutifully saving and recalling for you. These are plaintext files but not plain text files. Fiddle at your own risk (backups are for wimps, right?).

Copyright © 2026 Robert C. Furzer K4CY

'*Please* let me know' means:

- For problems and feedback on this manual, email me directly: Gary@isect.com.
- Post your technical queries, bug reports, comments, complaints and improvement suggestions/requests on the Logger32 reflector. This is the quickest way to get technical support since your message goes to the whole team – author Bob K4CY, the Logger32 support crew and other power users distributed liberally around the globe.
- Mention Logger32 to any of the friendly team behind this, when in QSO with us on the amateur bands. We *love* hearing from you! Tell us about your experiences. Let's talk!

Q. Why is this manual so HUGE? >1,000 pages! Overwhelming! Krazy!

A. TLDR: we know. Sorry. Too bad.

> If you can't explain it simply,
> you don't understand it
> well enough.
>
> *Albert Einstein*

The fundamental issue is that Logger32 has so much functionality and flexibility that it takes a *lot* of pages to do it justice – over 1,000 in fact with more than 320,000 words, some **long**.

Please bear in mind that, like radio amateurs generally, Logger32 users are a diverse bunch. Some of us are propeller-head technophiles who get our kicks from exploiting and meddling with technology, including IT. We wouldn't be satisfied with superficial or incomplete coverage. We ~~want~~ *need* to know what makes things tick. At the other end of the scale, some of us are technophobes who 'just want to get on with it': to us, the logging software is merely a tool that saves us having to scribble in a paper logbook … but the 'it' that we want to get on with is not the same 'it' that other users want, plus 'it' will undoubtedly change over time.

If you don't believe me, glance back through the list of functions and features, ticking the ones that matter to you: *your* tick-list is probably unique. Check it again in a month or three to see what's changed. Oh and by the way, do you know *exactly* which lines to tick, and does anything else on the list intrigue you? Have we caught your imagination? Read on!

This User Manual was explicitly designed to be a comprehensive, detailed, no-holds-barred, *definitive* guide to version 4 of Logger32, enabling users to dig deep, explore its inner workings, and discover the shiny golden nuggets hidden within … but we fully appreciate that some of you don't have the time, inclination or English reading skills for all this. We hear you, honestly we do. We have tried hard to make the manual accurate and usable. You will find *hundreds* of relevant, annotated screenshots, with carefully-phrased descriptions and explicit step-by-step instructions supplementing the pretty pictures.

A further part of our cunning plan was to develop more succinct, basic, shorter, lightweight and again *usable* documentation, using this manual as a basis, a source of inspiration and content, and a reference. We had in mind a quick-start guide to Logger32, maybe a series of short YouTube videos covering installation and configuration at least, translations into various languages, focused pieces covering specific aspects, and … well, what do you want?

Copyright © 2026 Robert C. Furzer K4CY

To be clear, we can't *promise* to do everything asked of us. Your requests and suggestions, though, will be well received if:

- You know what you want, and why. Explain your need or vision, well enough to persuade us that it has real value for all Logger32 users (not just you!). Examples or donor content showing the style and nature of the content (including materials from other logging programs, or completely different kinds of software) are very helpful, and demonstrate your commitment to get involved.
- Your ideas are practical, feasible, workable. A suite of SCORM-compliant eLearning modules with animated graphics, multi-lingual voice over and accompanying exercises sounds wonderful but, sorry, it's beyond our capabilities … unless …
- You are willing to participate in the generation of additional documentation, videos or other collateral. Seriously, if you have the drive, energy and passion to make stuff happen, we'll gladly help out. If you expect the world but aren't willing to lift a finger, don't be surprised at the, ahhh, lukewarm response. 'We' are a team of volunteers with various interests and priorities – not least making QSOs, and logging them with Logger32, naturally! The best way to get things done is to put in the effort. Show us you mean it. Give us a head-start.

On a smaller scale, focused, specific contributions are easier all round. We can usually adapt and adopt corrections, amplifications, additional examples, shortcuts, tips, anecdotes, better screenshots and diagrams, FAQs (and rare answers!) and so on. Don't worry about the formatting and layout, or even the grammar and spelling. We can handle all that. Please be clear about what changes you suggest, specifically it helps to know **which section number/s you think should be updated** (page numbers are not much use to us as they often change during the course of drafting and maintaining the content: section numbers are more stable).

Q. Why is this manual so out of date?

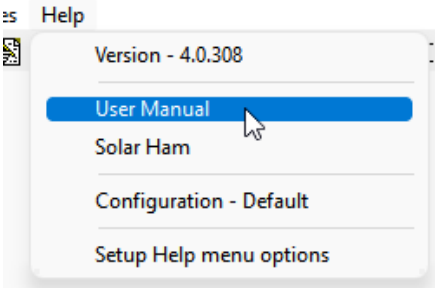
A. It isn't, at least we sincerely hope not in any material aspects! It is proactively maintained and the official online version is updated every month or so when new versions of Logger32 are released. For convenience, the download URL[18] …

https://www.logger32.net/files/Logger32_v4_User_Manual.pdf

… remains the same but the version number, date and document history section *within* the User Manual tell you about changes.

This User Manual is *not* included in Logger32 auto-updates because it is so big that some people would complain about the updates draining their mobile data and dimming the lights.

Have you tried downloading the current User Manual from the link on the Help menu lately? ▶



You *may* need to ⟳ reload the page ⟳ in your browser in order to grab the very latest edition.

[18] You're welcome to bookmark this URL in your browser, making it easy to grab the latest version.

Copyright © 2026 Robert C. Furzer K4CY



◀ Try holding <**Ctrl**> and tap function key <**F5**> or click the browser's curly-arrow page-reload button ⟳. We remind Logger32 users to do this from time to time on the Logger32 reflector, so long as *we* remember!

Deliferate mistale!

Bsoy and Girls! Somewhat in this blok,thre is a deliferate mistale. It may be a pocture or a world or ever the pange gumbers could be wrong. If you cal spot the DELIFERATE MISTALE, write to the Weditor, and you will delieve £24,000 by returb of nost.

Bert Fegg
Nasty Book for Boys & Girls (1974)

Q. Is there some reason why [a certain option, function, screen, form or whatever] isn't described in the manual?

A. Yes. I (ZL2iFB) accept responsibility as editor of this document. Often it is simply an oversight. I may not have noticed something amiss, including changes that have occurred in the software since the screenshots and descriptions were prepared. Sometimes there are differences due to the particular versions of Windows and PC/radio hardware we are using. Often I don't know enough about a given topic to spot errors and omissions, so I rely on Bob and the Logger32 beta team/support crew for assistance, along with feedback from readers.

Occasionally, being an old geezer, I lose my train of thought, distracted from writing by a cherry on my BandMaps …

Whatever catches *your* beady eye, by all means email me about it. I might not drop everything to address it *immediately* (cherries take precedence!) but I am grateful for complaints and improvement suggestions, using them in the course of editing the manual – an ongoing task.

Please check that you are looking at the current version of the User Manual before commenting, and please tell me which **section number**[19] concerns you.

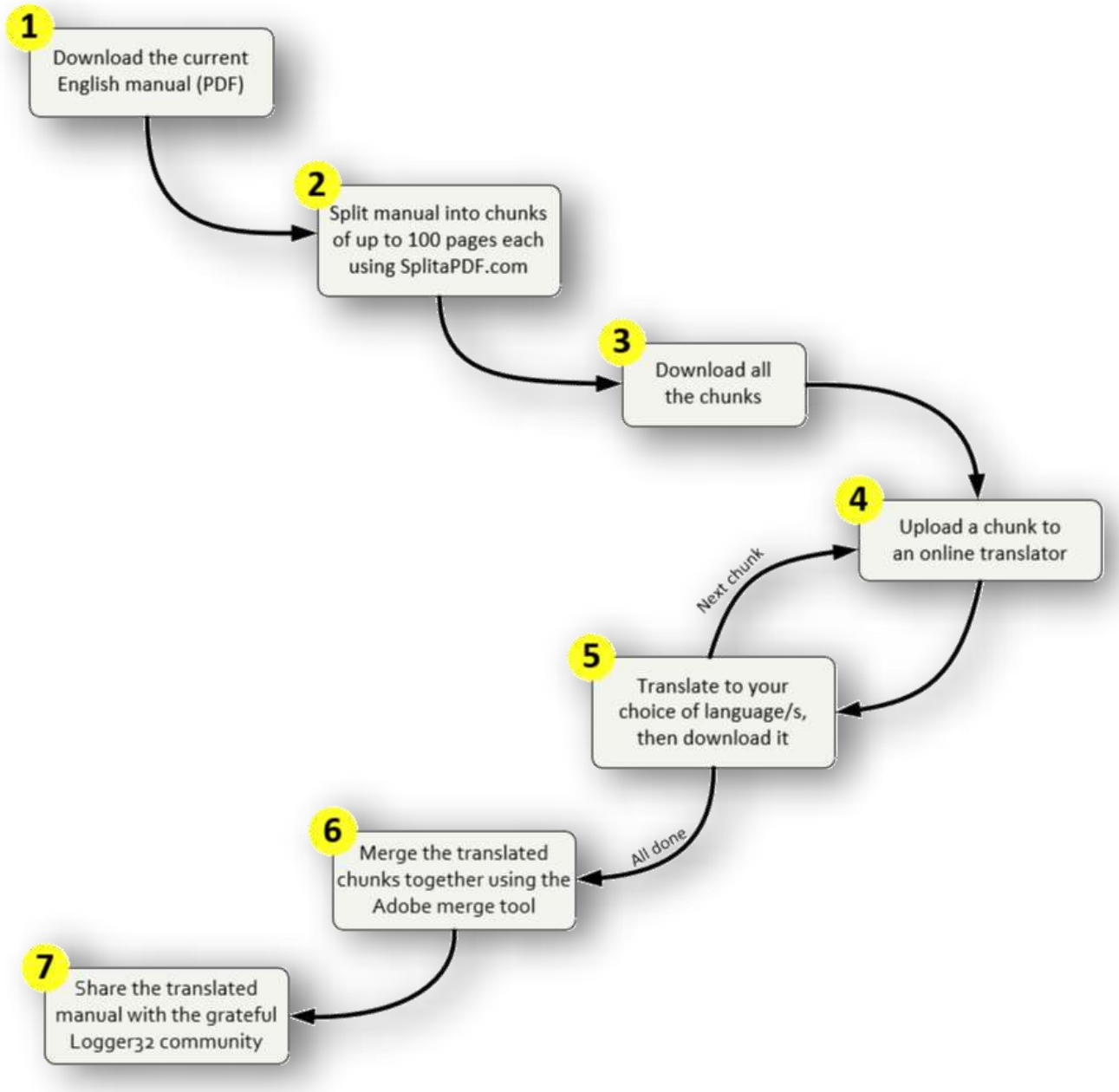
[19] Chapter and section numbers remain relatively stable whereas page numbers *often* change when editing and layout changes ripple through to the end.

Copyright © 2026 Robert C. Furzer K4CY

Q. Is this manual available in other languages?

A. This manual has been translated from English into French, Japanese and Spanish *so far*, thanks to the efforts of André F5JBR, Aki JA1NLX and John K8YC. Whereas André did the hard work manually, machine translations are *much* easier. Automated translation struggles a bit with the text formatting and placement of the numerous screenshots and other graphics throughout the document, but it ends up good enough for most purposes. However, internal hyperlinks between various parts of the English manual are mostly broken in the machine translations – sorry, but that's just how it goes until someone finds a translating machine willing to take on the whole of this 1000+ page monster of a PDF file in one go.

If *you* would like to machine-translate this manual to another language, here is the process:



Neatly numbered notes next …

Copyright © 2026 Robert C. Furzer K4CY

1) Download the current English version of this manual as a large PDF file to your PC.
2) Upload the large PDF to SplitaPDF.com and split it into chunks of 100 pages or less[20].
3) Download all the chunks to your PC as a *.ZIP* file, then unzip them into a convenient folder.
4) Upload one of the chunks to an online translator – whichever works best for you and your language. Aki suggests OnlineDocTranslator.com or DeepL.com but there are others.
5) Translate the split PDF to your chosen language and download it. **Read the translated file to make sure the translation and formatting are acceptable**[21], then upload, translate and download the remaining split PDFs in the same way until they are all done.
6) Merge the split PDFs into one large PDF again *e.g.* using the free Adobe merge utility, and check it once more for good measure.
7) *Please* share the translated PDF freely with your fellow hams in the same way as we are sharing this English version with you right now. You are welcome to publish and share it on the web yourself, or send it to Gary@isect.com to share with Logger32's global user community through the *Logger32.net* website.

The master (English language) version of this manual is updated most months, so please be prepared to update or re-create your translated version every so often.

Le paragraphe de l'aide est très bien fait

Merci beaucoup
Jean-Christophe
FESIND

Q. I grabbed a screenshot of an error message but forgot to paste it somewhere before capturing something else, and now it's gone. What can I do?

A. Ideally, please try to repeat whatever you did to trigger the error in the first place, and grab another picture. Bob and the beta test crew *love* tasty readily-repeatable errors that are well described since diagnosing and fixing the bugs is reasonably straightforward, and we can easily check that Bob's coding fixes have actually solved the problem.

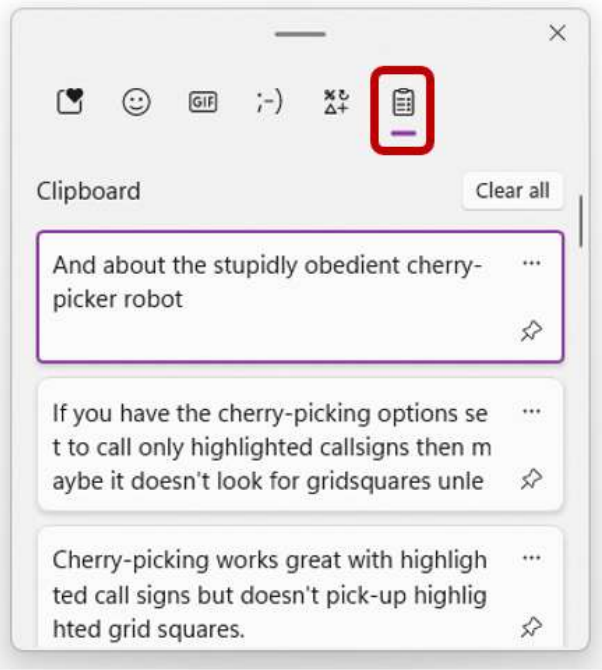
Yum yum, give us more.

[20] The original PDF is far too big for the free online translators we have found so far. If you find a utility that will translate the entire PDF file in one go, do please let us know.

[21] Although the text is generally translated quite well, the formatting is tricky due to our complex page layouts with lots of embedded images. Hyperlinked cross-references *between* chunks will be broken. The result may look ugly but hopefully it is worthwhile – a bit like me, really.

Copyright © 2026 Robert C. Furzer K4CY

If you cannot reproduce the error, things get hard for the fixer-uppers but hopefully you should still have a copy of the earlier screenshot in the Windows clipboard, provided you haven't rebooted since it was captured. To review previous contents of the clipboard, press <**Win**+**V**> to **V**iew a scrollable list of clips, something like this ▶



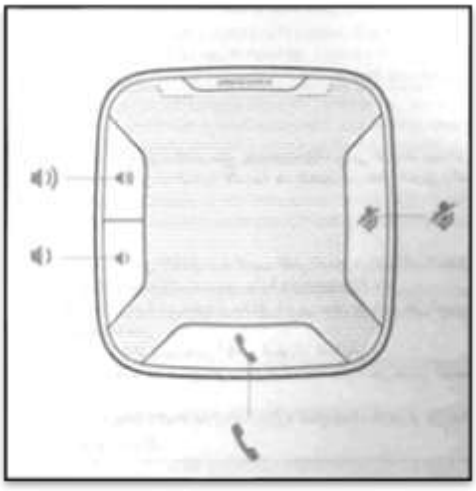
If not already selected, click the little clipboard icon on the right to show the history of recent clips, then drag the scrollbar through them to find the one you want and finally click it to paste it into your image editor or email program.

"Easy-as", we Kiwi DXers say.

My thanks to Bob and all the others who had a hand in this. I am not a power user of Logger32 but I've never had a problem the manual did not help me solve, plus all the good info from the reflector. I especially want to say how wonderful the latest manual is. I've been retired for 25 years as an engineer mostly working with technical documentation for military, telecom, and computer projects. We were just beginning to work on online documentation when I retired and I never imagined how far documentation could come in 25 years. I can barely imagine the huge amount of work that you all put into the manual (and the software). WOW!!! Thank you – thank you – thank you!!!

David N4DE

You're welcome David. On balance, we thought it best not to leave the documentation to the developers responsible for this example from a user manual ▶



Copyright © 2026 Robert C. Furzer K4CY

2 Installing and updating Logger32

> "Any sufficiently advanced technology is indistinguishable from magic"
>
> *Artur C. Clarke*

2.1 System requirements

Logger32 is a 32-bit application program (app) that runs on both 32 and 64-bit Windows systems. The current incarnation, version 4 (first released in 2021) has been tested and runs well on Windows 8.1, 10 and 11[22].

The program plus its attendant files and databases needs approximately 100 Mb of disk space. A logbook with 100,000 QSOs, plus backups *etc*., may take the total up to about 250 Mb or so.

> *I have been a Logger16 and Logger32 user since 1998. When anyone asks me about a logging program, I always recommend L32 to them and have helped several local amateurs setup their L32. I have also given a few presentations on this program to our local radio club. I find this program to be excellent: it does everything, even more than what I look for from a logging program.*
>
> *Bob VE1RSM*

[22] Logger32 *may* run under Windows emulators on Linux or other platforms, and on older Windows platforms such as Vista and Windows 7 … but **no support is offered**. Much like old hams, old computer hardware generally lacks both performance and capacity. Please don't be surprised if Logger32 refuses to install or operate as expected on your original IBM PC, just as Fred Flintstone should not expect a modern GPS system to work in his stone-age car. Getting Logger32 to run may take effort and skill. You may have more luck with another logging program designed and built for your old system. Failing that, we suggest pencil and paper, crayons, wax or stone tablets maybe with a cuneiform font. [\Sarcasm]

Copyright © 2026 Robert C. Furzer K4CY

2.2 Logger32 version 4 installation process

If you are using version 3 of Logger32 already, skip ahead to
the instructions on *upgrading* to Logger32 version 4
to avoid having to configure Logger32 from scratch
… unless you really want to start over, that is.

Once you have verified that your computer and operating system is compatible, download the Logger32 v4 full installer.

Run the installer and read the *Read.me* file before proceeding[23].

The installer creates all the necessary folders, installing the files in the right places and registering the **D**ynamic **L**ink **L**ibraries with Windows.

The default installation folder is ***C:\Logger32***. You *can* install it elsewhere (*e.g.* in, say, *C:\Ham Radio\Logger32* or *D:\Logger32v4*) but **avoid *C:\Program Files* and *C:\Program Files (x86)*** since those are special system folders, viciously guarded by Windows.

In this manual, we presume you have installed to *C:\Logger32.*

It is not absolutely necessary and reduces system security, but it makes things easier if you run Logger32 as an administrator. In File **E**xplorer, navigate to *C:\Logger32*, right-click *C:\Logger32\Logger32.exe*, click <**Properties**> then on the <**Compatibility**> tab, enable <**Run this program as an administrator**>, and finally click <**OK**>. Do the same for *both* autoinstaller programs: *C:\Logger32\Logger32autoinstaller.exe* and *C:\Logger32\autoinstaller.exe*

If your antivirus program creates false-positive alerts and blocks the installation, you *may* want to add *Logger32.exe* and/or the *C:\Logger32* folder to the exceptions (whitelist, trusted apps) in your antivirus package. We can't really help you with that: see the antivirus package's help or support information. Despite strenuous efforts, nobody can *guarantee* that Logger32 is totally benign and virus-free, just as there is always a risk of COVID-19 infections in real life.

When you are ready, launch Logger32 by clicking the Logger32 icon on your desktop, or double-clicking *Logger32.exe* in the *C:\Logger32* folder.

2.3 Initial configuration: six baby steps

With Logger32 installed and running, there are six things to do first:

1. Enter your **callsign** when prompted on the splash screen, read the conditions of use, and click <**Agree**> *if* you agree with them.
2. Set up your screen/display/window layout.
3. Define your station location.
4. Set your computer clock.
5. Update to the latest v4 release.
6. Import your existing log if you have one *e.g.* one exported as an ADIF file from whichever logging software or online logging service you used previously.

[23] This *may* contain important late-breaking installation instructions, so please don't just skip it.

Copyright © 2026 Robert C. Furzer K4CY

Step 1: Callsign setup



When Logger32 is run for the very first time, the initial screen asks for your callsign ►

Enter your callsign (the call you are using on-air) then click the button/box containing the words "I will strictly comply with the Terms and Conditions of use" … provided you do agree, that is.

Step 2: Screen/display/window layout

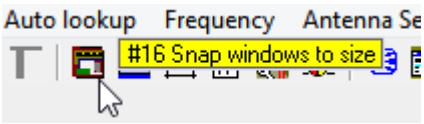
Logger32 initially opens a main window with eight smaller 'child' windows or panes. Adjust the main window to your preferred size and location, then position and size the child windows *within* the main window: child windows can *only* reside within the main Logger32 window (an 'MDI form'). Windows such as the Sound card data window, Tracking window, BandMaps, CW Machine, Data Terminal and **R**adio **C**ontrol **P**anel are grownups - not child windows - and hence can play anywhere on your monitor/s.

To find out which windows are or are not child windows, click and attempt to drag the caption (title bar or header area) of any window beyond the main Logger32 window. Child windows cannot pass the outer boundary of the main window (the play pen), whereas non-child windows can.

Child windows (such as the log entry pane, logbook, worked/confirmed table and previous QSOs pane) may be closed using the X in the top right corner and re-opened using the <**View**> menu or toolbar icons.

With a dual monitor setup, the main Logger32 window with its panes can be positioned and sized to fit on *either* monitor or to span *both* monitors. Non-child windows can be placed anywhere.

Aki tip: Aki JA1NLX maintains a gallery of Logger32 screenshots demonstrating how various Logger32 users lay out their panes, and which panes they find useful enough to display routinely. Aki's gallery is a source of artistic inspiration to get you started. In due course, consider emailing a snap of your own unique screen layout to Aki for inclusion in the gallery, to inspire others.



◄ Having positioned and sized the windows as you like, click icon #16 on the toolbar to fine-tune them and then use the layout for a while to confirm it suits you, sir.

Once you are completely happy with the layout, under <**View**> you can enable <**Lock child windows**> to avoid accidentally moving them. If you change your mind, simply disable that setting temporarily while you make adjustments.

The most clear and comprehensive user manual I have ever seen. Congratulations on a brilliant product. Keep up the great work.

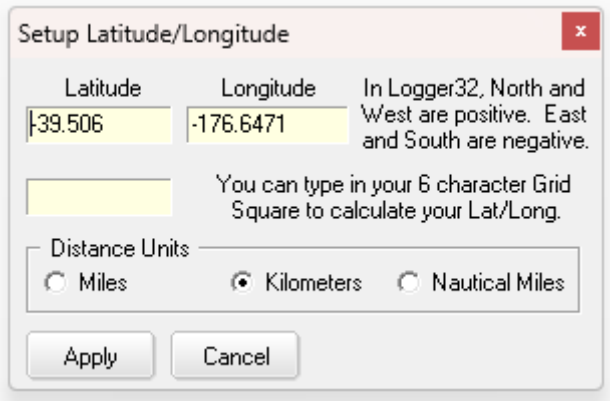
Tnx Peter VK6RZ

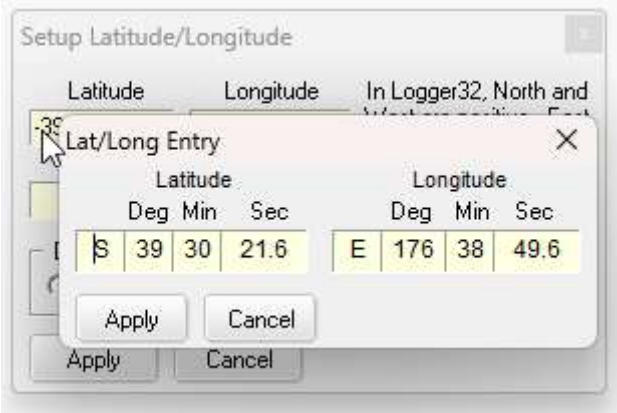
Copyright © 2026 Robert C. Furzer K4CY

Step 3: Define your station location and units

In order for certain functions to work correctly (*e.g.* to calculate beam headings and distances to DX stations or satellites), Logger32 needs to know whereabouts in the world you are.

To set up or change the location of your station, right-click any data entry field in the log entry pane, then click **Setup** ⇨ **My QTH Lat/Long** to display this configuration form ►





Enter your **latitude** and **longitude** in degrees and decimal degrees. North and West are positive values, whereas East and South are negatives (precede the numbers with a hyphen/minus sign).

◄ If you prefer degrees, minutes and seconds, right-click in the latitude or longitude fields then enter those values. When you click <**Apply**>, Logger32 converts Degs, Mins and Secs to decimal degrees.

If you don't know your latitude and longitude but you do know your six-character grid square ("Maidenhead locator"), you can use that instead: Logger32 calculates the position of the middle of the square, even if your station is located near an edge or corner. For most practical purposes, that's close enough.

You can always come back to update this later, after you've had the chance to check your GPS or zoom-in on your QTH using one of the online satellite imaging programs. In Google Earth for instance, the latitude, longitude and elevation of the area under the pointer are shown at the bottom edge of the window, and can be configured to show either decimal degrees, or degrees, minutes and seconds ▼

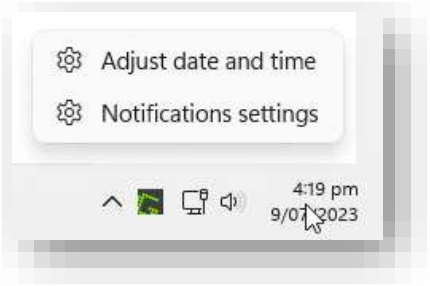


Copyright © 2026 Robert C. Furzer K4CY

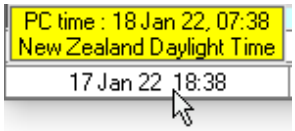
Now choose the units in which you want distances to be displayed in Logger32**.** The ADIF standard specifies that distances are to be *recorded* in kilometres, hence Logger32 always calculates and uses those units internally. However, if you prefer, the software can be configured to *display* distances in miles, kilometres or nautical miles[24]. Finally, click <**Apply**> to complete this step.

Step 4: Computer clock setup[25]

As per the ADIF standard, Logger32 records dates and times in UTC**. It is important that your PC clock is *reasonably* accurate and Windows has the correct time zone:**



1. Right-click the clock display in the Windows system tray (bottom right corner of the screen, usually), then click <**Adjust date and time**> to open the Windows Settings **Time & language ⇨ Date & time** form ▶
2. Select the appropriate **Time zone** for your location.
3. Click to slide the <**Adjust for daylight saving time automatically**> slider to <**On**>.
4. Slide the <**Set the time automatically**> slider to <**On**> to have Windows periodically correct the PC clock to align with a convenient NTP time server on the Internet, assuming you are Internet-connected. Otherwise, check and if necessary manually adjust the PC clock to the current *local* time in the shack using the <**Change**> button beside <**Set the date and time manually**>.
5. Click the corner X to close the Settings form.Logger32 calculates and displays the current **UTC** time at the left-hand end of the status bar at the bottom of the Logger32 window. If it is incorrect, double-check your PC clock settings.



◀ As a final check, mouseover Logger32's UTC time panel at the lower left to display a popup tooltip showing your PC (local) time and time zone.

With the correct time zone configured in Windows, you *should* not need to change your computer clock again provided the PC clock itself is sufficiently accurate. Read on for advice on how to get your PC clock to keep good time, correcting itself automatically by reference to Internet-based time standards using atomic clocks, or GPS, or a Cesium clock.

Step 5: Update to the latest v4 release

Although the 'full install version 4.0' downloaded from Logger32.net gets you up and running on version 4, there may have been important updates to Logger32 since that release, including various bug fixes and new functions. You don't need to run through the entire update sequence one-by-one: simply install the latest update using **Setup ⇨ Updates ⇨ Logger32 version updates** then click to tick **Setup auto updates** (to offer future updates whenever they are released) and then click **Look for updates now** to get Logger32 smack bang up-to-date.

[24] We're just waiting for some bright spark to suggest displaying distances in yards and feet, furlongs and chains, fathoms, London buses, cricket pitches or string-lengths …

[25] The instructions here are for Windows 11. On other versions of Windows, the process is a little different. Flounder around for yourself or Google it: honestly, it's not hard.

Copyright © 2026 Robert C. Furzer K4CY

Step 6: Import your log and get going!

If you have one, import your existing ADIF log file using **File ⇨ Import Logs ⇨ Import ADIF (.adi) file**. Simply follow the prompts or read the chapter on importing and exporting logs.

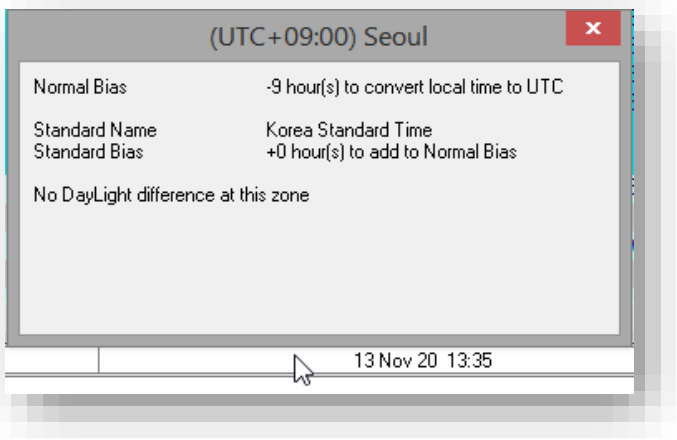
You can now start to log your contacts … but you may wish to read on and set up the logbook, the log entry pane, the previous QSOs pane and your bandplan first. If you will be using the DX cluster network or a CAT interface to your radio, those functions will also need setting up. There is a *lot* more advice in this manual – about 1,000 further pages to peruse at your leisure.

2.4 Times and time zones

Aside from the UTC date and time shown in the lower left-hand corner of Logger32's main window, over on the right-hand side near the bottom is another date and time panel that is only populated while there is a callsign (or at least a recognizable prefix) in the log entry pane ▶



It shows the **local time in the shack of the DX station you are working**. In fact, that entire row of information panels relates to the DX station.



◀ Right-clicking the DX station's shack clock at the lower right shows *his* time zone information.

Clearly, then, Logger32 knows about dates, times, time zones *etc*. but how? Internally, Logger32 uses UTC. It uses the computer time (typically maintained by a battery-backed **R**eal **T**ime **C**lock unit on your PC's motherboard) and the time zone settings in Windows[26] to calculate UTC, adjusting as necessary for Daylight Savings Time (a.k.a. Summer Time *etc*.) at the appropriate period in the year.

2.4.1 Set your PC clock

Logger32 uses your PC clock as its time reference. If your PC clock is wrongly set, inaccurate or unreliable, all the time calculations based on it will be equally wrong, inaccurate or unreliable.

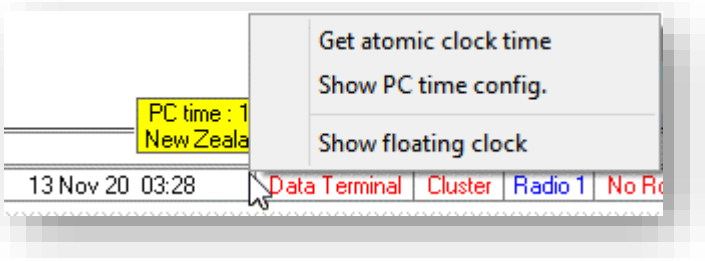
Extreme "scientific" clock precision is unnecessary for virtually all amateur radio purposes … but it pays to set and maintain the PC clock *reasonably* accurately.

If your PC is online, Logger32 can adjust the computer's built-in clock automatically using one of the many Internet-connected atomic clocks.

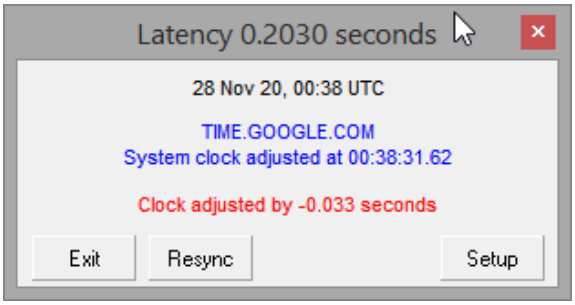
[26] For some reason, the 'Norway' regional settings in Windows evidently cause JTDX QSOs to be logged via UDP with the wrong dates in Logger32 … but a simple workaround is to change the regional setting to 'Sweden'. It appears to be a failure to translate the date to Julian format using the Norway setting. There *may* be unwelcome side-effects with this workaround.

Copyright © 2026 Robert C. Furzer K4CY

Right-click the UTC clock at lower left for a menu with three options ▶



- **Get atomic clock time**: queries an Internet-connected time standard (time.google.com by default) using SNTP[27], setting the system clock accordingly.



◀ The 'latency' figure in the heading is the measured round-trip time delay between sending a request to the time server through the Internet and receiving its response. The latency value is used to adjust the time value as received back to what it presumably was when sent.

Whereas normally Logger32 checks and adjusts the system clock periodically[28], <**Resync**> does an immediate check and adjustment.

- The <**Setup**> button lets us configure a different SNTP time server and choose when to check and update the PC clock ▼



There are options to sync the clock when the UDP BandMap is open because timing is a little more important for synchronous digimodes such as FT# than for legacy modes such as CW and SSB. Digimode software such as JTDX|WSJT-X copes with discrepancies up to about 2 seconds, allowing for a second or so of inaccuracy in the clocks at each end plus the transmission, path/propagation and reception delays[29].

[27] Resetting the PC clock is a privileged Windows thing, meaning that you need to run Logger32 as an administrator for this function to, errr, function. Naturally, also, you must be connected to the Internet. If you have no Internet access, you can use GPS or HF radio time standards such as WWV. At a pinch, the average timing of FT# transmissions on a busy band can get you within a second or so of the correct time.

[28] It checks when Logger32 starts up, and re-checks every 30 minutes while the UDP BandMap is open.

[29] Latency is a concern with **S**oftware **D**efined **R**adios. Complex digital signal processing takes time!

Copyright © 2026 Robert C. Furzer K4CY

- **Show PC time config.**: displays the computer's time settings from Windows, for your information ▶

  Should you need to change anything here, do it through Windows.
- **Show floating clock**: brings up a faded black and white photograph of a mysterious clock found on a piece of driftwood in the South China Sea.



2.5 Bandplans: the Bands & Modes table

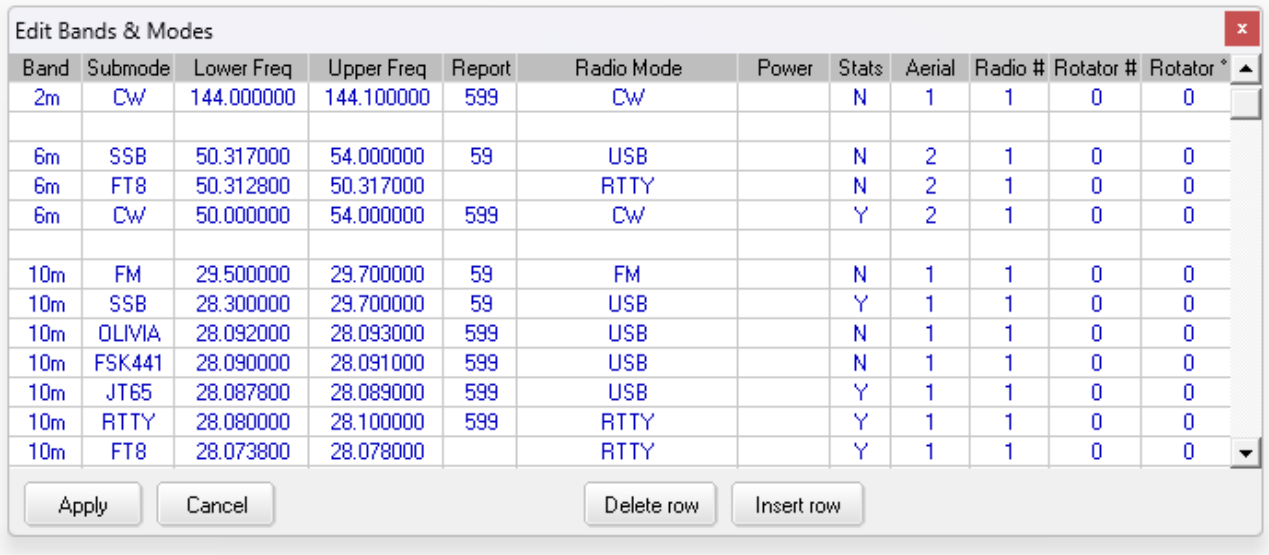
Logger32's bandplan (configured in the **Bands & Modes** table) helps identify from, say, a DX spot that gives only a callsign and frequency, which mode the spotted station is *probably* using, plus the corresponding mode settings for your radio/s and even the appropriate antenna and rotator for that band if you have an automatic antenna switch and digital rotator.

The table determines which amateur bands and communication modes/protocols are shown in the log entry pane, logbook, worked/confirmed table and various other places in Logger32 *e.g.* it associates a valid ADIF band with a VFO frequency (*e.g.* 3750 kHz is in what ADIF specifies as the "80m" band), and specifies which aerials and rotators to use on various bands, or rather band-segments.

Although the default bandplan is a reasonable start, you can customize it to your specific requirements with up to 200 entries, for example if the frequencies of mode segments in your part of the world differ from the defaults, or to add preferred frequencies ('watering holes') for new digital modes as they come into use.

2.5.1 Setup Bands & Modes

To check, modify[30] or delete a default BandMap entry, or add a new band or mode segment, open it with **Tools ⇨ Setup Bands & Modes** ▼



[30] Make as many changes as you need, and *then* click <**Apply**> to save and implement them.

Copyright © 2026 Robert C. Furzer K4CY

- **Band** for this segment – this must be a valid ADIF band[31]. See the FAQ for clues.
- **Submode** for this segment - must be a valid ADIF mode or submode. See the FAQ for clues. This 'operating mode' is normally shown at the top of the log entry pane and is logged with your QSOs: if not, check the ADIF MODE and SUBMODE section for further inspiration.
- **Lower (sub-band) Freq** for this mode segment: the frequency in MHz of the very bottom (LF) end of the segment (*e.g.* for the 20m CW segment, the lower edge is 14.000[32]). **Reading the table top-down, the lower sub-band entries in the Bands & Modes table *must* be in descending order of frequency.** Trailing zeros are automatically added if needed.
- **Upper (sub-band) Freq** for this segment - frequency in MHz of the top (HF) end of the band (*e.g.* for 20m SSB *and* CW it would be 14.350).

  The order in which the segments are entered is important - for example:

  - 20m SSB **14.150** 14.350 59 USB 100 Y
  - 20m RTTY **14.080** 14.120 RTTY 50 Y
  - 20m PSK **14.065** 14.075 USB 50 Y
  - 20m CW **14.000** 14.350 CW 100 Y

  The lower (LF) sub-band edges are (in descending order) **14.150, 14.080, 14.065** and **14.000**.



Hinson tip: *strange* things happen if Logger32 lacks a default mode for the current VFO frequencies … so **define a CW segment from the very bottom edge to the very top of every band you use**. Without exception. Just do it. Trust me, I'm a doctor.

To add, say, the 20m SSTV sub-band, you would need to add a new line to the table. Click to place the cursor on the line *above* 20m SSB (starting at 14.150), then click <**Insert row**> and enter data for the additional line, ending up with:

- **20m SSTV 14.220 14.230 599 USB 100 Y**
- 20m SSB **14.150** 14.350 59 USB 100 Y
- 20m RTTY **14.080** 14.120 RTTY 50 Y
- 20m PSK **14.065** 14.075 USB 50 Y
- 20m CW **14.000** 14.350 CW 100 Y

Click <**OK**> to enact the changes. By adding this new line in the correct position:

- You have successfully added SSTV as a valid mode to Logger32.
- When your CAT-connected radio is tuned to a frequency in the SSTV sub-band, the mode will default to SSTV so that new QSOs are logged as SSTV QSOs (unless you deliberately change the mode).
- If you click a DX cluster spot with a frequency between 14.220 and 14.230, the radio will QSY and the radio mode will be set to USB.

[31] Visit https:/adif.org/ for the current ADIF definition including the enumeration of bands, modes *etc*. If we *need* to log QSOs on a non-ADIF-defined band, we can add it to *C:\Logger32\ADIFbands.txt* … but be aware that strict ADIF compliant systems may reject our exported ADIF logs outright, or (if we are lucky) they will skip the applicable QSOs, perhaps warning us about this anomaly.

[32] Whereas radio frequencies are stated in kHz elsewhere in Logger32, they are in megs here.

Copyright © 2026 Robert C. Furzer K4CY

- If you tune the radio HF across the 20m band from the bottom edge with auto-poll on and mode controlled by the radio, the mode in the log entry pane will initially show CW, regardless of which mode is set on the radio. As you tune upward to reach 14.065, the log entry mode changes to PSK. At 14.075 the mode changes back to CW, then at 14.080 it changes again to RTTY. At 14.120 the mode reverts to CW until, at 14.150 it changes to SSB. Between 14.220 and 14.230, it changes to SSTV, then back to SSB.

- **Report**: this column sets the default signal report (*e.g.* 59 or 599) for each band+mode segment. If you don't want the standard reports to appear as you enter QSOs in the log entry pane, or to be saved by default when you log the QSOs, leave the field blank.
- **Radio Mode**: this determines which mode commands are sent by CAT to the radios for this band segment. Check your radio's manual. Old radios may accept just the basic SSB, CW, RTTY, AM and FM mode descriptions here but newer ones typically need more detail *e.g.* Flex radios need either CWL (CW with LSB) or CWR (CW with the reverse sideband *i.e*. USB), rather than plain CW. Some radios need a dash in the mode descriptions *e.g.* CW-L and CW-R.

  Sending an invalid mode command to the radio will set it to USB above 14000 kHz or LSB below 14000 kHz.

  Hinson tip: CW nuts like me who prefer CW-LSB might try "CW-R" (CW reverse) in the hope that our particular radios understand the CAT command. If not, it may be possible to configure the radio itself to select LSB automatically, by default, when in CW mode. Good luck.

  Single **O**perator **2** **R**adio operation is supported: see this SO2R chapter section for more information. In summary, the syntax is: **radio 1 mode | radio 2 mode** *e.g.* RTTY|USB

  Crossband QO-100 satellite operation is supported: see the FAQ.

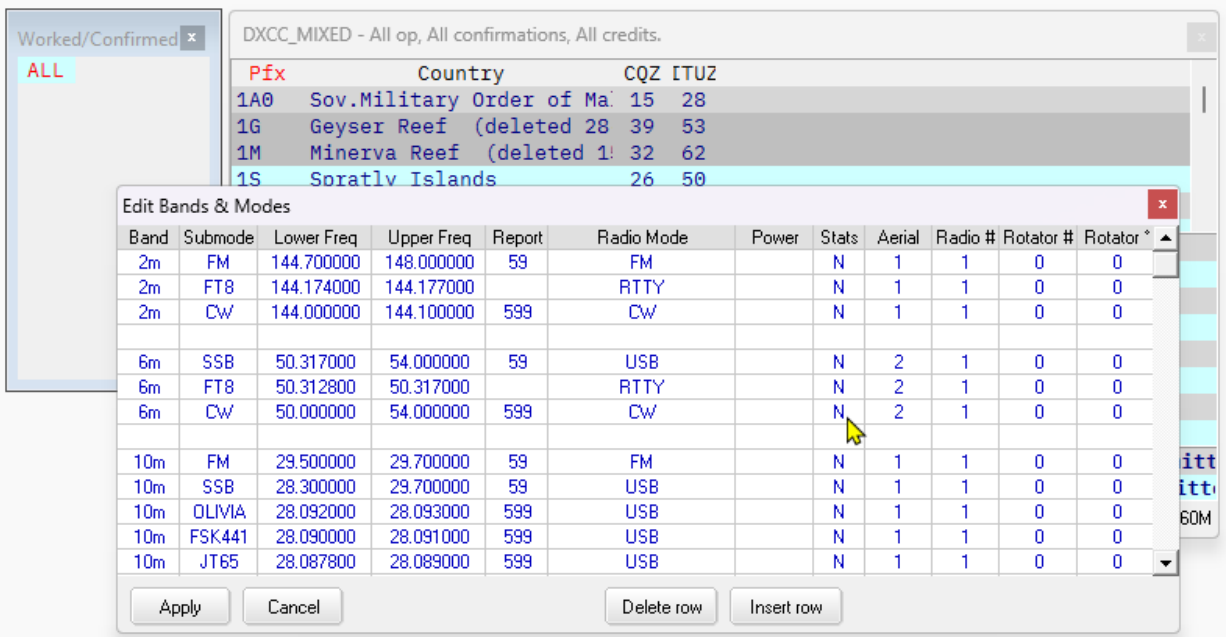
  Whereas new radios with new command sets and options appear on the market all the time, the Logger32 team does not have the inclination, time or budget to play with all the very latest toys. New/updated versions of software for **S**oftware **D**efined **R**adios present similar challenges. All popular radios are supported, plus many others. If your specific radio model is not listed, it is worth trying other models from the same manufacturer since the commands are *usually* similar, although some functions may not operate as expected[33].

  Some radios have a DATA mode, typically used in conjunction with audio connectors on the rear panel for receiving and transmitting digital information. DATA mode generally mutes the radio's microphone and disables speech processing so it will not interfere with the computer-generated audio during transmit, and a constant audio output level may be provided on receive, regardless of the volume control. Read the radio's instruction manual for hints about whether your radio has this useful facility, and if so how it works.
- **Power:** your transmitter power to be logged by default for this sub-band.

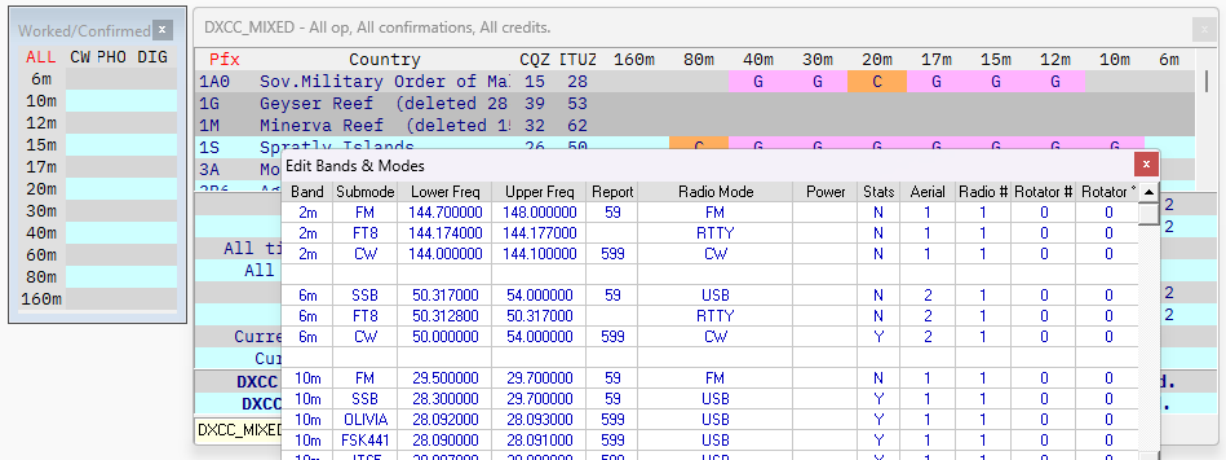
[33] You are welcome to *request* CAT support for new radios, SDR updates *etc*. through the Logger32 reflector. Your request is more likely to succeed if you provide explicit details of the commands and format *e.g.* a link to the CAT section of the manufacturer's technical documentation, citing the applicable section or page number. Please be patient as it takes time to specify and code the changes, and we are busy people, not sitting around idly waiting for your instruction. In due course, when the changes are ready, you will probably be invited to check that everything works as it should with your shiny new radio before general release.

Copyright © 2026 Robert C. Furzer K4CY

- **Stats**: an indicator to show if statistics are to be calculated/displayed in the worked/confirmed table. If *all* the rows in this column are left blank or marked "N," the worked/confirmed table ▼ will be mostly blank and your DXCC award report will have no band columns ▼



With "Y's" in the Stats column, Logger32 knows which bands and modes to show you in the ▼ worked/confirmed table and which bands to list in the DXCC report ▼

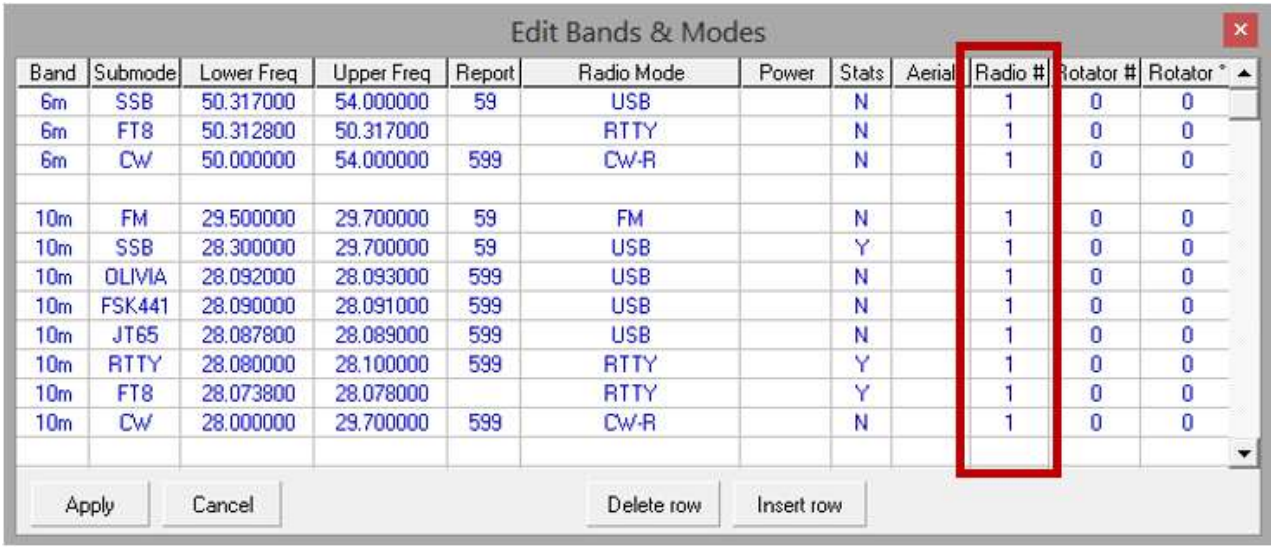


The sequence of "Y" entries affects the 'default' worked/confirmed mode column order … but the mode columns can be manually re-sequenced if you wish.

- **Aerial:** the antenna number to be selected by automatic antenna switches. If you have the capability, different antennas may be selected for radio #1 and radio #2 for the same band segment (*e.g.* on 20m, radio #1 uses antenna #1, a monoband 20m beam, whereas radio #2 uses antenna #2, a tribander). The entry in the aerial column for this would be "1|2" (one bar two, without the quotes).

Copyright © 2026 Robert C. Furzer K4CY

- **Radio #:** which number radio to be used for this particular band segment ▼



This function was originally designed to allow automatic control of separate HF and VHF/UHF radios. These days, you might like to run a dedicated barefoot radio for FT# alongside your main HF DXing station. You'll find plenty of information about interfacing and controlling radios in section D of this manual.

Some radios do not automatically switch their selected antenna if the band is changed by CAT command as opposed to manually changing bands from the radio front panel. A workaround has been implemented in the bandplan: a selection in the Radio # column (column 10) can specify *both* the radio *and* the antenna to use[34]. With a syntax similar to the radio mode column (column 6), the format is **x, xy|z** or **x y|z** where **x** (either 1 or 2) is the default radio to use, **y** (0 to 9) is the aerial port on Radio 1, and **z** (0 to 9) is the aerial port on Radio 2. If no aerial is to be selected, use 0 as the aerial port number. The vertical bar character may be the shifted backslash key on your keyboard … and I appreciate that might not help you find it! Good luck!

Some examples:

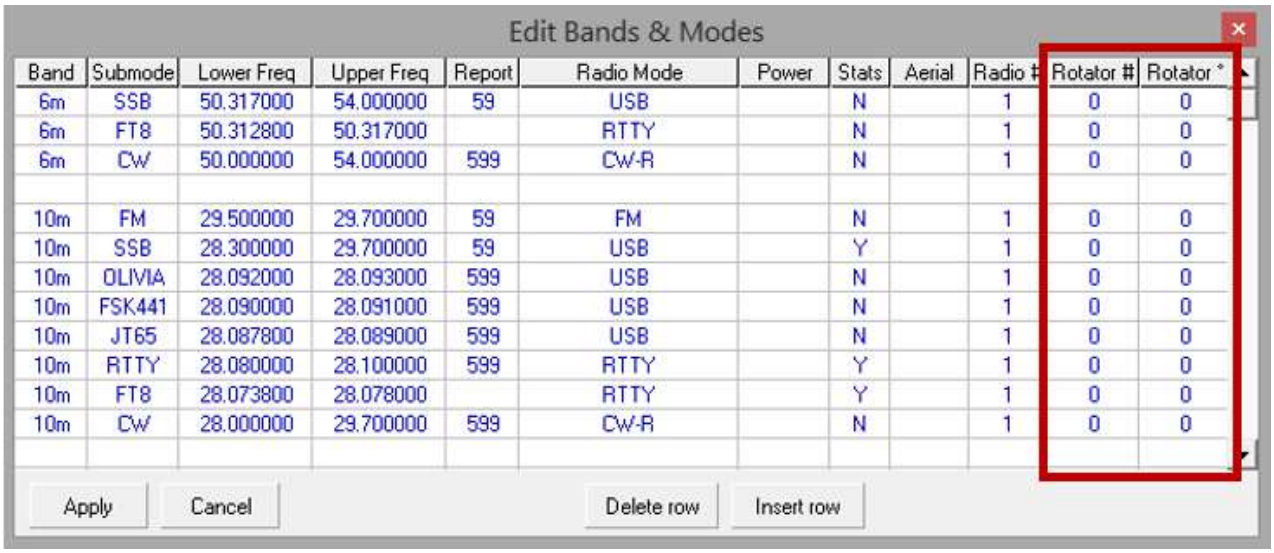
- "2" – simply defaults to Radio 2 without switching either Radio 1 or Radio 2's built-in aerial connectors.
- "2 0|0" - same as above.
- "17|8" - sets Radio 1 as the default with aerial 7. If Radio 2 is selected (say for SO2R operation), aerial 8 is to be used.
- "1 7|8" - same as above (the space is optional … but makes it a little easier to comprehend!).
- "2 0|3" - sets Radio 2 as the default radio with aerial 3. Switching to Radio 1 does not issue any aerial selection command to Radio 1.

As Logger32 does not read the radio's antenna port, reverting to the previously-selected radio or antenna (by menu selection) will *not* reset the radio to the previously-selected antenna.

[34] <**Automatic radio change**> must be selected in the radio configuration for this feature to work. Right-click the Radio panel on the status bar to find this option.

Copyright © 2026 Robert C. Furzer K4CY

- **Rotator #**: define which rotator number is used for each band/frequency segment. If a rotator is not present or used for any band/segment, put a 0 in this column ▼



- **Rotator ***: this column ▲ lets you compensate for directional antennas mounted sideways to the main beam and azimuth indicator on the rotator control box. For instance, if you have a little 6m beam mounted 90˚ *clockwise* from your HF tribander (looking down on it from above), put -90 in the Rotator * column on 6m in order to have Logger32 subtract 90˚ from the intended headings when pointing the antenna on that band. If no compensation is needed (*i.e*. the antennas are aligned, pointing the same way), put 0 in this column.

  See the Antenna rotator chapter for advice on compensating for mast/rotator slippages, declination and magnetic variation, and lots more besides.

Hinson tip: take care! When you've finished editing the Bands and Modes table, double-check anything you've changed *before* <**Apply**ing> it. Proofread carefully. Subtle little data errors in this table can cause weird hard-to-pin-down side-effects in Logger32 such as indicating the wrong bands for given frequencies, and rotators failing to rotate on some bands. If you don't have time to check it carefully now, click <**Cancel**> and come back when you're not so rushed … or anticipate weirdness ahead.

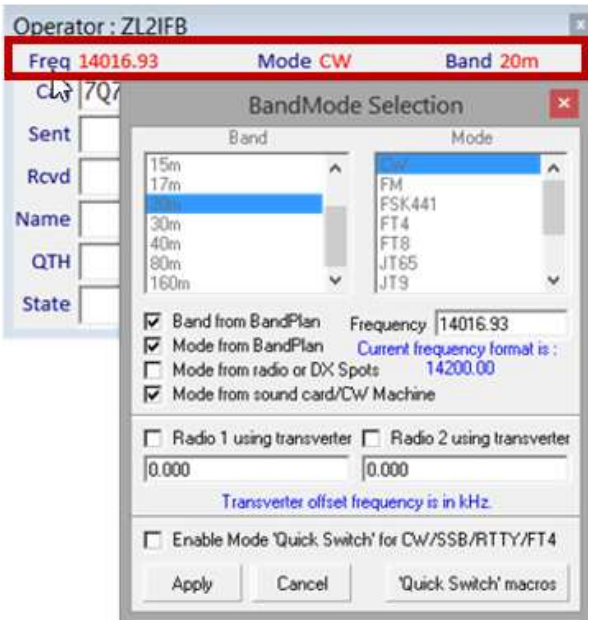
Here's a solution to a problem in case any other noob needs it: all of my 4m contacts were rejected on import as they were 'not in the 4m band'. Actually, the 4m band had not been defined in the Bands & Modes table. Entering the relevant info there caused all the QSOs to be imported correctly.

Mike MOAGP

Copyright © 2026 Robert C. Furzer K4CY

2.6 Logging bands and modes

Click anywhere on the top line of the log entry pane where the Freq, Mode and Band are shown to configure how Logger32 determines and logs the frequencies, modes and bands you are using ▶



2.6.1 Automatic logging of bands and modes

Both the band and mode can be updated automatically as you tune through the bands on a CAT-connected radio. No more forgetting to log the correct details as you hurriedly change band or mode to bag a new one!

Exactly how the automated band and mode logging functions work is determined by:

- **Band from bandplan**: the band recorded in the logbook is determined from the Bands & Modes table. If the table is set up correctly, a QSO on a frequency of 14.070 MHz will be logged with a band of "20m", for example, because that frequency is between the lower and upper frequency limits for "20m".

> **Hinson tip**: if you are using JTDX|WSJT-X with the UDP BandMap option <**Allow WSJT/JTDX to set Logger32 frequency/mode**> ticked, Logger32 takes the **mode** from the digimode software *provided* the mode is listed in the Bands & Modes table *somewhere* (not necessarily within the present band). It looks-up the **band** in your Bands & Modes table corresponding to the frequency reported by the digimode software. In other words, make sure both the band *and* the mode exist in your Bands & Modes table.

- **Mode from bandplan**: the "submode" column of your Bands & Modes table determines what 'operating mode' is logged for a QSO according to its frequency. For example, a QSO on 14.074 MHz is probably FT8, even if the radio is reporting its 'radio mode' as USB or DATA via CAT.
- **Mode from radio or DX spots**: normally determines the 'operating mode' from your CAT-connected radio *e.g.* LSB, USB or DATA. This option better supports the mode sent in the comments field by Skimmer nodes such as those in the **R**everse **B**eacon **N**etwork. DX spots that state a specific mode in the comments field are treated as such with the corresponding highlighting if you 'need' a QSO on that mode and band.

  Say for example you have your bandplan setup with 20m CW defined from 14.000 to 14.350 and RTTY from 14.080 to 14.100. A basic DX spot on, say, 14.0801 would be *assumed* to be RTTY. However, if the DX spot comment said "FT8 fox-n-hounds", it is obvious (to those of us who read cluster comments) that the DX is using FT8 … but:

  - With <**Mode from bandplan**> selected, clicking the DX spot sets the radio to RTTY (or whatever radio mode that particular model uses: maybe DATA or FSK) and prepares to log a RTTY QSO using MMVARI.

Copyright © 2026 Robert C. Furzer K4CY

- With <**Mode from radio or DX spots**> selected, clicking the spot sets the radio to DATA mode (or whatever it uses) and, with Quick Switch enabled, sets things up for an FT8 QSO using JTDX|WSJT-X.

- **Mode from Sound card/CW Machine** takes the actual mode from MMVARI (*e.g.* RTTY or PSK) or from the CW Machine (CW, obviously), whichever you are using to make QSOs. If the UDP BandMap options <**Allow WSJT/JTDX to set Logger32 frequency/mode**> *and* <**Mode from sound card/CW Machine**> are *both* selected, CW takes precedence while the CW Machine is open. On closing the CW Machine, the mode in the log entry pane reverts to FT# when Logger32 receives the next update via UDP from JTDX|WSJT-X.

2.6.2 Manual logging

If your radio has no CAT/computer interface or for some reason you prefer to enter frequency[35], band, mode and/or submode yourself, un-tick <**Band from bandplan**>, <**Mode from bandplan**> *and* <**Mode from Radio**> giving you complete manual control over what appears in the log entry pane, and hence gets recorded in your logbook. If that's what you want, that's what'll happen.

2.6.3 Using transverters

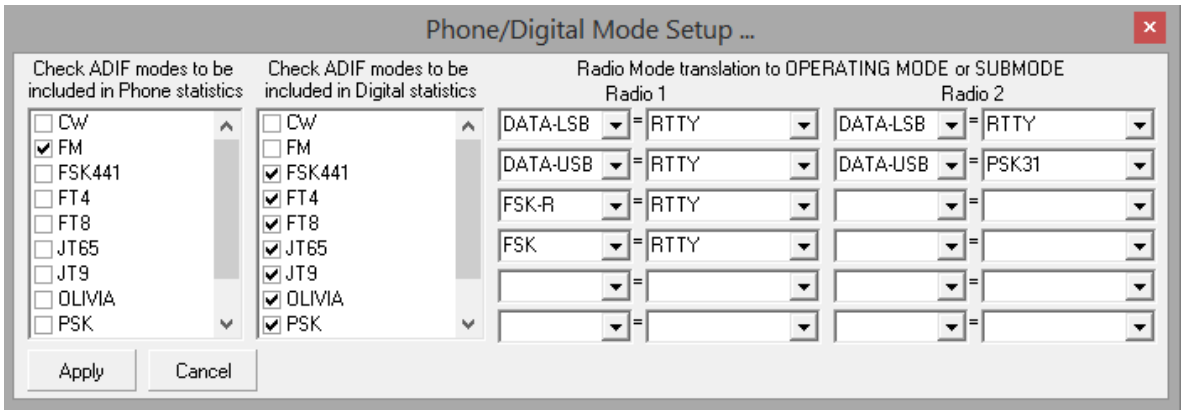
If your CAT-enabled radio is used as the tunable IF for a fixed-frequency up or down converter (called a transverter if it works on both transmit and receive), the radio typically reports its VFO frequency to Logger32, rather than the ultimate radio frequency. You can adjust for this on the BandMode form which you open by right-clicking the word "Mode" in the log entry pane.

The offset for radio #1 is applied when radio #1 is active, and likewise for radio #2. Enter the frequency offset (either positive or negative) in kHz and place a check mark for the radios as appropriate to tell Logger32 to activate the offset(s). For example, if radio #2 operating on 28 MHz drives a 2m transverter, the frequency offset is +116000 kHz *i.e.* 116 MHz gets 'added' to the RF frequency generated on transmit, and 'subtracted' on receive.

2.6.4 Phone/digital mode setup

You can decide which modes get aggregated towards awards under the general "DIG" (digital) and/or "PHO" (phone) banners, and which modes are logged if you are using <**Mode from radio**>.

Set these from the Logger32 main menu **Tools ⇨ Database Maintenance ⇨ Setup Phone/Digital Modes ▼**



[35] The "Current frequency format" shown here reflects your settings under **Setup ⇨ Frequency**.

Copyright © 2026 Robert C. Furzer K4CY

Modes that are ticked in the two left-hand lists count towards your PHO and DIG statistics, respectively. So, for instance, you can choose whether QSOs using some form of digitized speech count as phone, or digital, or both.

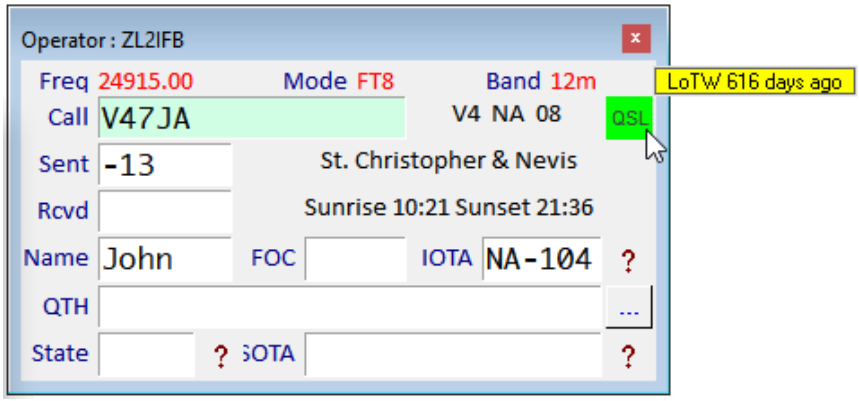
The order and content of these lists are dependent on the Bands & Modes table. If a mode you wish to include does not appear here, then it is not (yet!) in your Bands & Modes table.

The right-hand side of the table concerns the <**Mode from radio**> option. Logger32 can translate modes used and reported by each radio to different modes recorded in the log and exported (*e.g.* in the ADIFs): for instance, you may use a radio 1 mode of DATA-USB to make RTTY contacts, but the same DATA-USB mode on radio 2 is used for, say, PSK31 contacts. For the radio modes on the left of the equals sign, the modes on the right will be logged.

Hinson tip: don't fret about occasional oddities, for instance if you whistle up a CW QSO on a sideband radio, or tap one out using the PTT key on an FM radio [yes, I've done both!]. You can always edit the modes recorded for QSOs in your logbook. Logger32 uses sensible defaults for the usual, boring, routine stuff, leaving you to manage the rare and interesting exceptions.

2.6.5 LoTW/OQRS user indicator (the colored blob!)

Provided you have loaded the LoTW user file into Logger32 (a plain text file listing callsigns, one per line), the log entry pane will display a colored[36] marker blob *if* the callsign currently being logged is found in the file.



A tooltip pops into view when you mouseover the blob ▲ showing how long it is since they last uploaded their log to LoTW, if so configured. If the LoTW user file came to you via Club Log, the tooltips can also show if the station uses Club Log's **O**nline **Q**SL **R**equest **S**ystem.

Read about configuring the blob in the LoTW chapter, the DX spots chapter, and the UDP BandMap chapter.

2.7 *Upgrading* Logger32 from v3 to v4

2.7.1 Why should I *upgrade* to v4?

Version 4 is a major upgrade from prior versions of Logger32. Bob K4CY changed some 10,000 lines of source code under the hood. In particular, the database engine (a proprietary ISAM database) was upgraded and increased in capacity, and the ancient serial port driver was upgraded to a

[36] Blob color is a configurable option.

Copyright © 2026 Robert C. Furzer K4CY

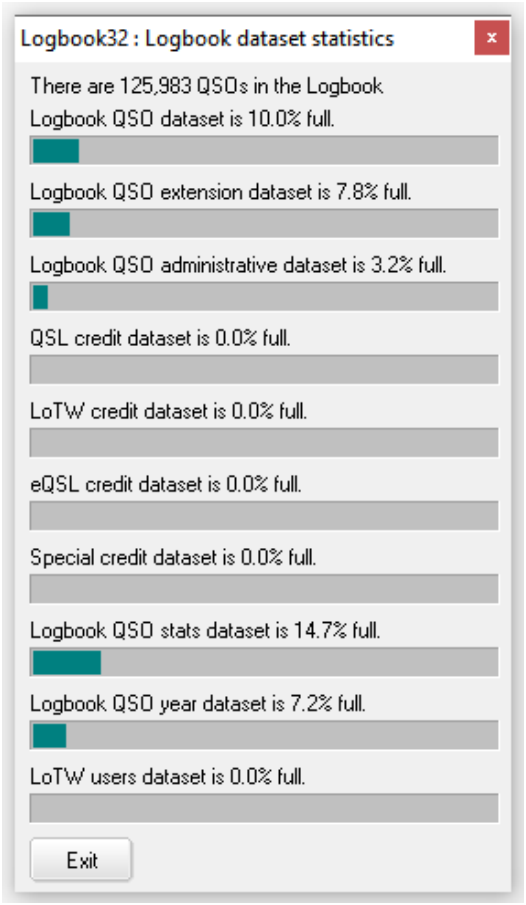
supported version that allows up to 256 serial/USB ports to communicate with rigs, amplifiers, rotators and antenna switches – more than enough for the most complex of stations.

Usability improvements were made in various areas *e.g.* clarifying the purpose of data entry fields, the Quick Switch function, and a simple way to align the BandMaps.

Version 3 and previous versions are no longer maintained or supported. Exciting new Logger32 features are *only* available on version 4, and are not retro-fitted to older versions. Support for v4 is available as always on the Logger32 reflector.

Users report that v4 starts faster and is generally more responsive in use. It's a hit! A 17,000 QSO logbook that occupied nearly 6% of available database capacity under v3.5 uses just 1% under v4. The LoTW/OQRS user database that previously took 5% now takes just 1%.

ZL2iFB's version 4 installation is good to log over a million QSOs even if he carries on recording copious details and notes about the stations he contacts ▶



The process for *migrating* Logger32 v3.5 to v4 is painless. **It retains your current Logger32 configuration so you will not *need* to reconfigure the screens and options from scratch** but you can do so.

2.7.2 Prerequisites for *migrating* v3.5 to v4

You will need:

- A working Logger32 v3.5 installation, specifically version 3.50.421 or later.
  - If you are running an even older version, or if you don't presently have Logger32 installed and running at all, you need to do a **full v4 installation**, *not* a migration!
- "Removable storage" such as a USB flash memory stick, CD-ROM/CD-RW or external hard drive, on which to save a safety backup of your valuable log/s. Don't skip this!
- The Logger32 v3.5 to v4 upgrade zip file: download it from: https://www.logger32.net/files/logger32_upgrade_3.50.421_to_4.0.132.zip

I finally got around to migrating from 3.5 to 4.0 (following the upgrade instructions to the letter) and my compliments to Bob and the rest of the development team for a totally pain-free experience. Thanks again for a great product!

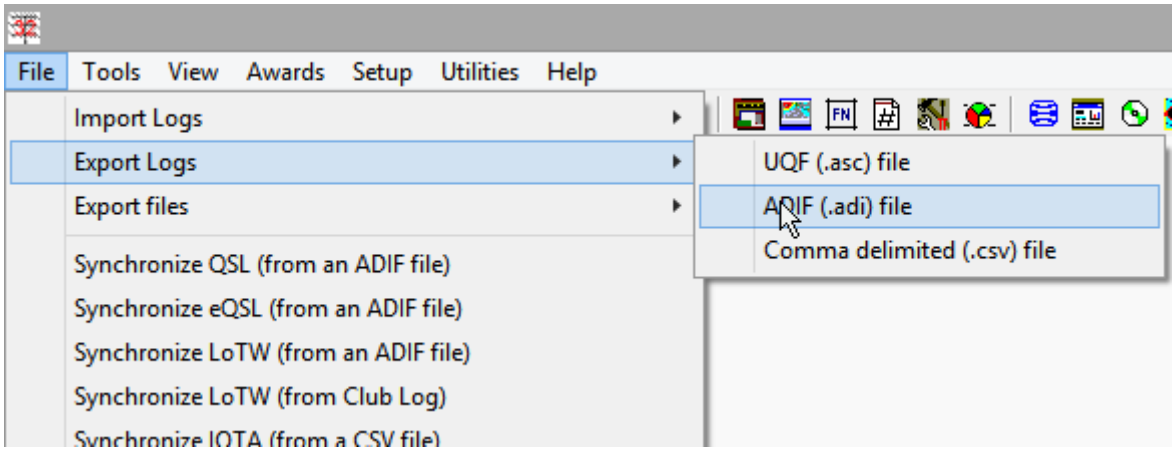
Duncan G3WZD

Copyright © 2026 Robert C. Furzer K4CY

2.7.3 Backup v3 first!

We *strongly* recommend making backups of your valuable log/s *before* doing the v4 migration, just in case. Swim between the flags.

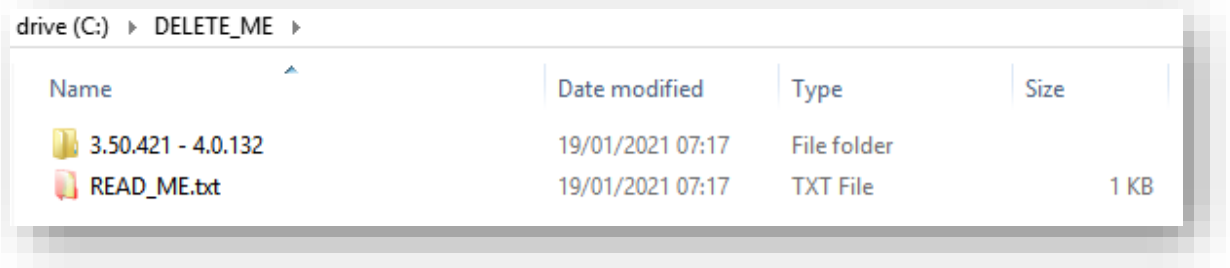
1. Make an ADIF export of your entire log, and save it safely to removable storage[37] using **File ⇨ Export Logs ⇨ ADIF (.adi) file** ▼



2. If you run several logbooks (*e.g.* for your normal, contest and club callsigns), open each one in turn, making ADIF exports and saving them to removable storage in the same way. Be sure to backup every log or risk losing them.
3. Close Logger32 v3[38] for the very last time. Relish the moment.
4. Remove/unplug/disconnect the removable storage, label it and stash it safely away. You may need it one day - hopefully not today though!

2.7.4 Migrating to v4

1. Unpack the Logger32 v3.5 to v4 upgrade zip to a temporary folder *e.g. C:\DELETE_ME* ▼

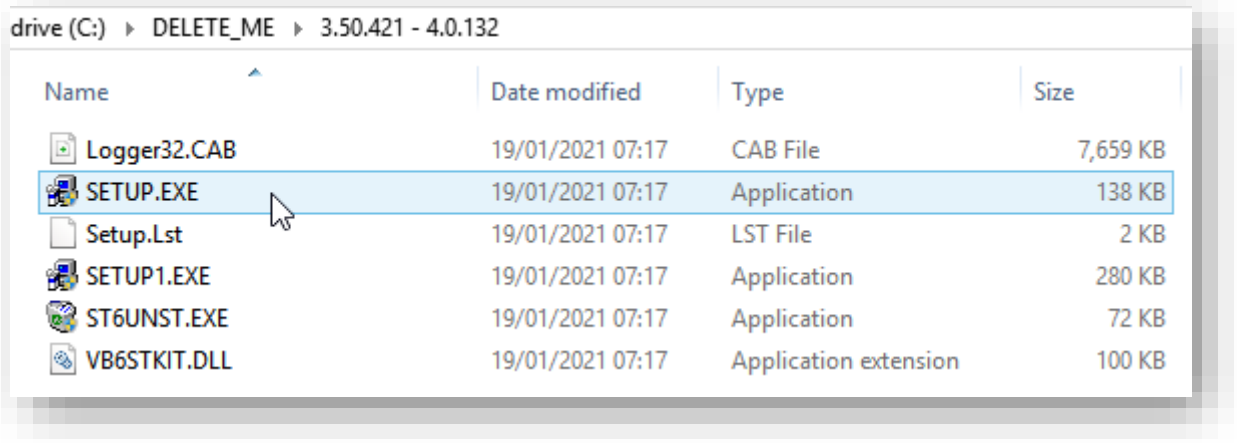


[37] Such as a USB memory stick, a CD-ROM/CD-RW or an external hard drive – something you can physically remove from the PC and store safely. Cloud storage *may* be fine *provided* the cloud is there and the cloud app works reliably … Your log is valuable, so take good care of it!

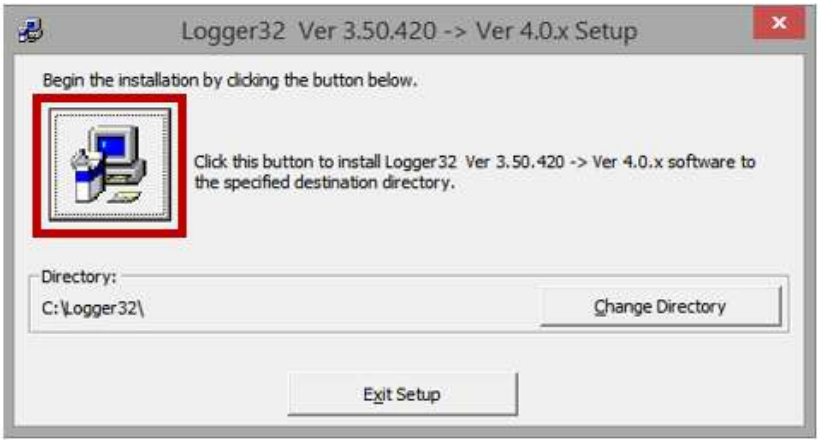
[38] These instructions apply to upgrading Logger32 from any v3.5 subversion to v4. Once at v4, Logger32 can update itself automatically to the very latest and greatest v4 subversion, soon after its release.

Copyright © 2026 Robert C. Furzer K4CY

2. In File **E**xplorer, click to open the *C:\DELETE_ME\3.50.421 – 4.0.132* folder, then double-click *SETUP.EXE* to run it and so install version 4 ▼



3. If Logger32 is not currently located in the default folder *C:\Logger32* on your computer, click the <**Change Directory**> button, then find and select your Logger32 folder[39] ▼



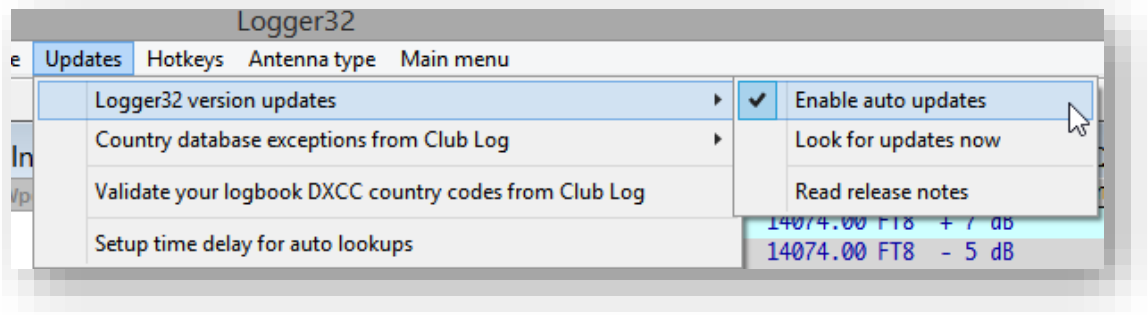
4. Click the big square computer image button to continue the installation ▲
5. Answer the installation questions. For most of us, the default answers are fine. That's why they qualify as 'defaults'!
6. Wait until the installation is complete. It shouldn't take long.
7. Start Logger32 and notice on the startup splash screen that you are running version 4.something. It will launch and run faster with much greater QSO capacity, shiny new serial port drivers and various other improvements.
8. Logger32 will immediately open and migrate your current logbook to the new version 4 internal database format, a quick one-time irreversible migration[40] - so quick, in fact, that you probably won't even notice.

[39] The v4 migration process works well if you upgrade Logger32 in the original folder. If you try to move it to a new/different folder, you are on your own! In particular, take note of the warning in section 2.2.

[40] Yes, irreversible. Changes under the hood, once made, cannot be reversed (just as you can't *un*scramble an egg!). You may be able to recover a working 3.x system and log from backups taken before the migration, but your recovered 3.x log will be missing any QSOs logged on the 4.x system, and it *should* not be necessary in any case.

Copyright © 2026 Robert C. Furzer K4CY

1. Provided it is configured to auto-update, Logger32 will also update itself to the very latest version 4 release. To enable auto-updates (as recommended) click to tick **Setup ⇨ Updates ⇨ Logger32 version updates ⇨ Enable auto updates ▼**



   If for some reason you choose *not* to enable auto-updates, *try* to remember to find and click <**Look for updates now**> on that updates menu from time to time, in order to take advantage of bug fixes and other improvements to Logger32. If you drop too far behind, you will no longer qualify for sympathy and support on the Logger32 reflector.

9. Recalculate statistics[41] using **Tools ⇨ Database maintenance ⇨ Recalculate statistics.**
10. Check it out! If you run into problems, see the next section. If it all works well, please help us spread the word by telling your friends, contacts and social media followers that you are *thrilled* with Logger32 version 4. Oh, and please spare a thought for its author Bob K4CY and the Logger32 beta testing and support team.

> *Migration to V4 was flawless here. Have tried other logging programs before, and occasionally I test a new/renewed one, but I keep returning to L32. TNX all for the efforts.*
>
> *Marc ON7SS/0090*

2.8 Move an *existing* Logger32 installation to a *new* one

Imagine you are a deliriously happy Logger32 version 4 user. You've been using the software for quite a while. You've gradually customized and fine-tuned your installation, chosen your favorite fonts, colors and audio alerts, connected your radios and peripherals, laid out the panes and logbook columns just-so, and you like it just the way it is … like a comfortable old shoe.

Then, out of the blue one fateful, disastrous day, a nearby lightning strike blows your PC motherboard and wipes all the hard drives. ***Oh no!*** The old shoe is toast!

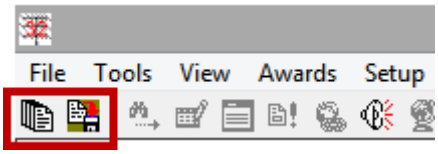
Less dramatically, you finally bite the bullet and invest in a (relatively) new computer to replace that old wreck of a machine you've been painstakingly nursing along for years. Your new shoes may be a bit stiff and uncomfortable, but *boy* do they shine! They gleam and glint in the sun!

[41] This step *may* not be necessary but several upgraders have reported that it helps, and it won't do any harm so we are recommending it for everyone. It can take a while to run, though, so if you are desperate to get going with version 4, you can run the recalc later when it is more convenient, no worries.

Copyright © 2026 Robert C. Furzer K4CY

Either way, you quite reasonably expect to run Logger32 on new/replacement PC hardware, retaining not just your log and statistics, but ideally all those lovingly-crafted Logger32 customizations you have accumulated over the years too. Here's how.

1. If it is still usable, make backups on the old system of your logbook and user files, using the two left-most icons ▶



   Preferably, make ADIF exports of your logs too, and stash them somewhere safe.

   If your old computer, or at least its disk, is unusable (*e.g.* burnt to a frazzle, fried to crisp, making weird clicks and scraping noises, eaten by your dog, stolen by burglars …), you will be able to recover your Logger32 installation and log using your most recent backup.

You *do* have backups, right?

2. Do a **full install** of Logger32 on the new PC, replacement disk drive, new partition or virtual machine, or whatever. Follow the instructions. Do what it says. Be patient.
3. Verify that the new Logger32 installation works OK. It's up to you what you check and how much verification you do but here are some suggestions:
   - Run it! Does Logger32 launch and complete its start-up process without obvious errors or warnings? Marvel at how quickly it starts on your shiny new setup!
   - Once running, the default screen layout is a bit of a mess, so move the windows around and resize them roughly, enough to complete the remaining checks anyway.
   - Run an auto-update. How does that go? How about a Club Log exceptions update?
   - Create a test logbook and log a few fake QSOs. Does the log entry pane work as expected? Do the QSOs appear OK in the test log?
   - Configure and open the CAT connection to a radio – just one will do for now. Does the VFO frequency appear in the log entry pane? Is it being logged correctly?
   - Configure and try out your favorite peripherals such as a WinKeyer or *micro*HAM box, and add-on/utility programs such as MMVARI, JTDX/WSJT-X, L32 LogSync and CheckCall.
4. **Provided things have gone well so far**, attempt to make the new installation resemble what you had before. There are three ways to achieve that:
   - Manually reconfigure everything to your liking. This *is* tedious but it's also an opportunity to make sure that the new installation is properly configured in every way, making good use of the shiny new hardware. There are literally *hundreds* of pages of well-meaning advice in this very manual to help you achieve this! The more you have already done to set up and verify the new installation, the less remains to complete the reconfiguration.
   - Restore the *.INI* files from the old installation's *C:\Logger32* folder to the new one, restart Logger32, and generally mess around with things until you get it working satisfactorily, dealing with various challenges along the way (such as missing fonts and different port addresses). Sorry to be so vague here but it's hard to know what might come up. You've chosen a shorter but rockier path.

> **Hinson tip**: whereas most Logger32 settings are simple enough to reconfigure manually, some take more effort – such as the colors, macros and message buttons, and logbook columns. Personally I find it quicker and easier to copy the relevant sections from

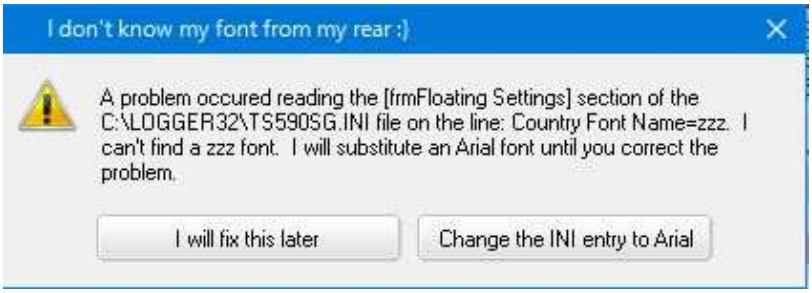
Copyright © 2026 Robert C. Furzer K4CY

archived, known good copies of the relevant *.INI* files from the old computer into the corresponding sections in the corresponding *.INI* files on the new one, replacing those sections one-by-one, opening Logger32 to check progress at each stage.

- Alternatively, simply copy the entire contents of *C:\Logger32* from the old computer into *C:\Logger32* on the new computer, and hope for the best. You may be copying old files that are no longer needed (wasting a few kilobytes), and some of the original settings may not be appropriate for the new computer, so be prepared to fiddle around with things until you have it working to your satisfaction.

2.8.1 Missing fonts

Logger32 sometimes complains about missing fonts on your system - for instance if you transfer the Logger32 installation from a PC that has certain fonts installed, to a different PC *without* those fonts ▶



In this case, you have two options:

1. **I will fix this later**: missing fonts are *temporarily* replaced by Arial; however, this setting is not saved permanently in the *.INI* file[42] so the error message will reappear the next time you start Logger32 *until* you change the fonts configured in Logger32 or locate and install the missing fonts through Windows.
2. **Change the INI entry to Arial**: missing fonts are replaced by Arial and *Logger32.INI* is updated accordingly. You can still change fonts, install additional fonts *etc*. later on, if/when you decide to do so, but meanwhile Logger32 should start up and run without any font errors.

Hinson tip: Logger32 checks the Windows registry to determine which font to use for its menus. If the registry doesn't specify a particular font, Logger32 *presumes* you are in fact using "Microsoft Sans Serif" font – specifically, that font, exactly as written. You may well get errors if "Microsoft Sans Serif" font is not installed on your PC, or if it has even a *slightly* different name.

2.9 v4 installation and migration troubleshooting

If this happens …	… try this
You can't *migrate* Logger32 to version 4 … because you are not yet running Logger32 …	Don't *migrate*! Instead, download and *install* Logger32 version 4 from here.
Your antivirus program complains about a virus in *Logger32.exe*	Don't panic! See our advice on antivirus.

[42] The fonts configured in various parts of Logger32 are specified in *Logger32.INI* and other *.INI* or *.ini* files.

Copyright © 2026 Robert C. Furzer K4CY

If this happens …	… try this
You get error/warning messages from Windows about security issues in the installation process	Read them! The installer *needs* privileges to install Logger32 version 4, and Windows is just letting you know. Once it is installed, Logger32 does *not* need privileges routinely *except* for the final part of auto-updates and the PC clock sync function.
Add-on utility programs no longer work properly	Check that they are being launched from the correct folders; make sure you have the current version of the software; ask whoever provided the programs if they are compatible with Logger32 version 4.
When you try to edit a QSO already in your log to, say, 20M, an error message tells you "20M is not a valid Band in your C:\Logger32\MyBandMode32.db file"	In version 4, bands are now denoted with an SI-compliant lowercase **m** for metres rather than capital **M** as was used in version 3. So try using 20m not 20M.
FT# isn't working on [at least one] band	Check your Bands & Modes table
FT# QSOs logged in JTDX|WSJT-X aren't being transferred to the Logger32 logbook	Ensure the UDP port settings in Logger32 match those in JTDX|WSJT-X; the UDP BandMap should preferably be open - well, the UDP port at least
Some panes/windows that you used to see in Logger32 v3 are no longer open in v4	Open them from the <**View**> menu, and if necessary resize and reposition them: they *should* behave themselves and reopen correctly in future
Some functions/ports that used to work in Logger32 v3 are no longer open and functional in v4	Open them from the applicable toolbar icons, status bar panels or menus, and if necessary check/update the configuration details: in future, they too *should* reopen and behave themselves obediently
My log has been trashed, somehow	Recover it from those backups you *thought* were an unnecessary waste of time and disk space …
Importing my ADIF log from another program fails after about 100 QSOs, filling BAD.ADI with errors	Are you trying to log to a cloud drive? The cloud synchronization process may be interrupting the import. Try logging to a local disk drive instead.
You're stumped! Something just doesn't work as described: the installation or upgrade fails or Logger32 v4 simply refuses to start and run properly	Turn it off and back on again! In more detail: 1. Try stopping and restarting the problem program. It may help to *temporarily* suspend your antivirus software. Don't forget to restart it in due course. 2. If that doesn't work, shutdown and restart the PC (a soft reset).

Copyright © 2026 Robert C. Furzer K4CY

If this happens …	… try this
	3. If that doesn't work, try a hard reset: shutdown and power-off the PC completely, then power it up and boot it, avoiding the temptation to kick it where it hurts. 4. If that doesn't work, try reinstalling Logger32 v4 using a full installation, having first saved a backup of your log and any *.INI* files just in case. 5. Try renaming *C:\Logger32\Logger32.INI* to something else *e.g. Logger32-DUBIOUS.INI* As it starts up, Logger32 will then create a new *Logger32.INI* with all the defaults. If that works OK, there was evidently a problem with your original *.INI*. Here are the instructions on what to do next. 6. If you are still having problems, the Windows system file checker tool *may* find and fix missing or corrupted Windows files and is worth a go. 7. Fire up the magick crystals and explore the occult. 8. As a last resort, email the Logger32 Reflector about your situation and we'll help you find a solution. Please carefully describe what you do and what you see, especially any error messages. We *need* the clues. Note: "refreshing" and especially resetting or reinstalling Windows plus all your apps from scratch can take hours and may not solve anything. Hold that option in reserve as an absolutely desperate very last resort ever once you have exhausted other, more promising approaches. Twice.

There are further notes about installation problems and error messages in the FAQs at the end of this chapter.

2.10 Automated Logger32 v4.x updates

Logger32 can be configured to check for and (if found) download and install program updates from the Logger32.net website, automatically, whenever you launch Logger32.



◀ Enable the auto updater under **Setup ⇨ Updates**

Three options are available:

- **Enable auto updates**: check online for program updates each time Logger32 starts up[43].

[43] If Logger32 runs 24x7, it does not *periodically* check for updates. It only checks once when started up.

Copyright © 2026 Robert C. Furzer K4CY

- **Look for updates now**: check online *immediately* for any Logger32 updates. If an update is available, Logger32 tells you, giving you the option to install it at your leisure (read ahead for more advice).
- **Read release notes** opens *C:\Logger32\Release Notes\Auto-update release notes.txt*, a text file briefly outlining changes made in the most recently-installed update.

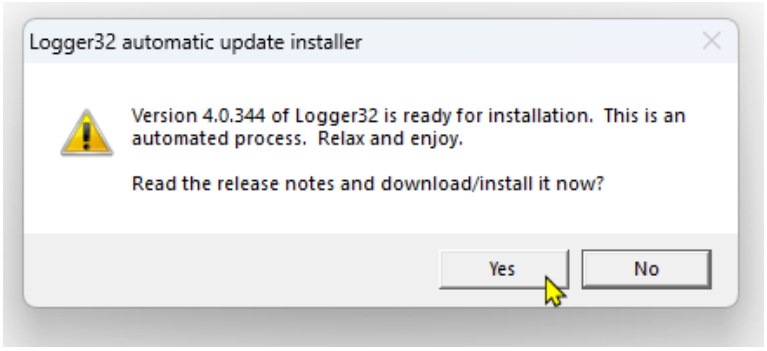
When the update checker runs, messages appear in the bottom right corner of the screen, near the Windows clock *e.g.* ▼

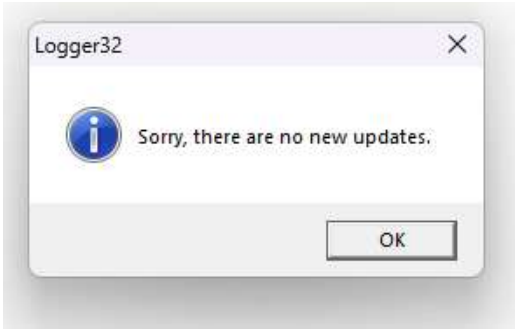


… followed by …



Logger32 also politely informs us in further messages whether updates are available … ▶





◀ … or not. Occasionally, you may see ▼

Maybe the server is down or overloaded, or your Internet connection is disconnected.



If an auto-update is offered, click <**Yes**> to read about it in the release notes as you decide whether to install it.

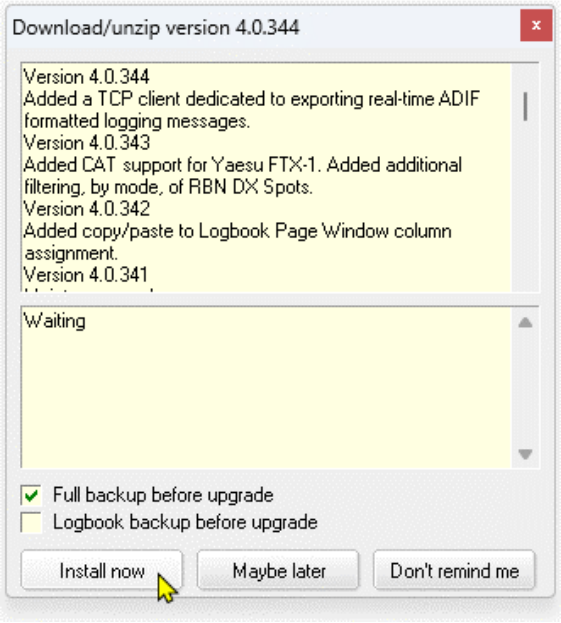
Hinson tip: don't ignore the release notes. On rare occasions there will be specific warnings or explicit instructions, such as "Recalculate your statistics following changes to Logger32's internal database". Ignore these at your peril!

When an auto-update is installed, its release notes are stored in *C:\Logger32\Release Notes*[44].

If an auto-update is *not* offered, maybe there isn't one at the moment, or maybe something fell over somewhere along the path between Logger32 on your PC and the *Logger32.net* server Out There on the Internet. Maybe there is no DX propagation at the moment. See below for further clues.

[44] If an update causes you problems, please take *another, even closer* look at the release notes before even thinking about complaining on the Logger32 reflector.

Copyright © 2026 Robert C. Furzer K4CY



◄ The release notes outline key changes in each version with the very latest one at the top of the list, in Bob K4CY's charmingly succinct matter-of-fact style.

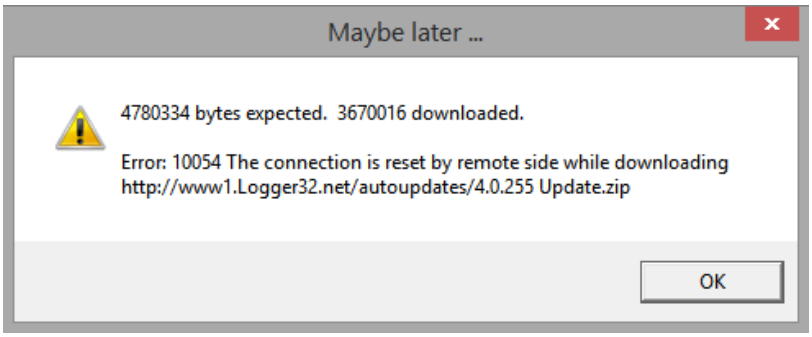
◄ Progress information about the auto-update will appear here after you click <**Install now**>.

◄ A check box offers you the chance to make full or logbook-only backups[45] prior to updating. **Do it!** No regerts!

▲ When you're ready, click <**Install now**> to download and install the update.

Logger32 checks the auto-update file size before attempting to download it, so you may see an error message like this if it fails to download correctly[46] ►

Wait a bit and try again.



Thank you Bob for all your hard work. I installed version 4 with no problems and it's already updated. Piece of cake. Of course I backed up my 3.5 files just in case <grin>. I've been using this program for years and it has never let me down. A quality piece of software.

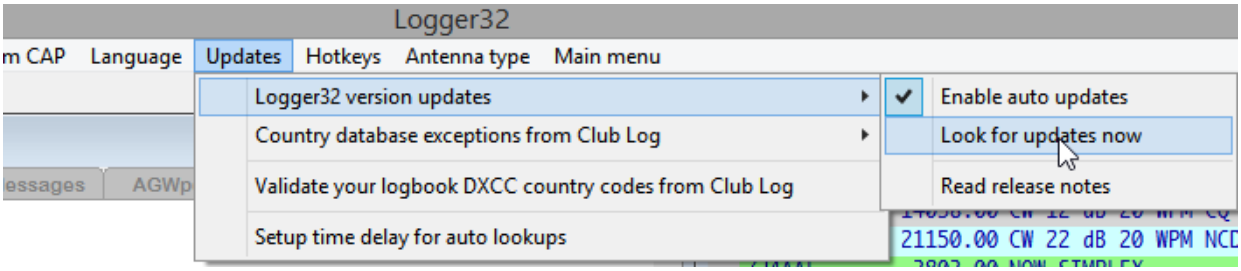
Dave WB2PJH

[45] If you save your backups to the cloud (*e.g.* Dropbox), updates may fail to download due to the time needed to upload backups. If this happens, you might have more luck saving backups on a local drive that is automatically synchronized with the cloud service in the background (*e.g.* One Drive or Google Drive). To be safe, you might like to initiate a backup manually before starting the auto-update process.

[46] Network errors are generally transient, so wait a moment and try it again. You might be lucky.

Copyright © 2026 Robert C. Furzer K4CY

When the download succeeds, click <**Install now**> to proceed with the auto-update. If now is not a good time, click <**Maybe later**> to be reminded again about this update when you next start Logger32, or when you click **Setup ⇨ Updates ⇨ Logger32 version updates ⇨ Look for updates now**▼



If you decide to skip an update altogether (*e.g.* if you tried it but found it didn't work properly on your system), clicking <**Don't remind me**> tells Logger32 not to bother you any more with the notifications for the skipped update … but it will pester you about *subsequent* updates unless you choose to skip them too. Eventually – *hopefully* – you'll take the hint and update to the very latest, fully-supported version.

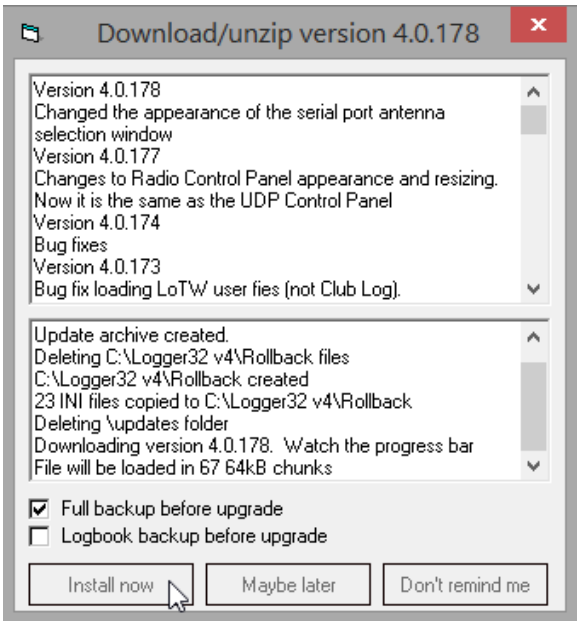
Hinson tip: ~monthly Logger32 updates generally address reported issues, and sometimes add new features. Rarely, there are internal database/system changes to improve performance and capacity, to comply with updated ADIF specifications, or to work properly with current versions of Windows. All updates have been coded, tested and refined by the development/support crew. If you are as risk-averse as me, consider that *not* updating may be riskier than updating … and before you update, make another offline ADIF backup of your log just-in-case.

During the process:

- The progress pane is updated with messages about what's going on … ►
- Logger32 creates a new rollback folder in *C:\Logger32* (if necessary deleting the oldest one to retain no more than ten folders), then copies into it all the files needed to restore the present version if the upgrade doesn't work out.
- A progress bar shows the backups ▼

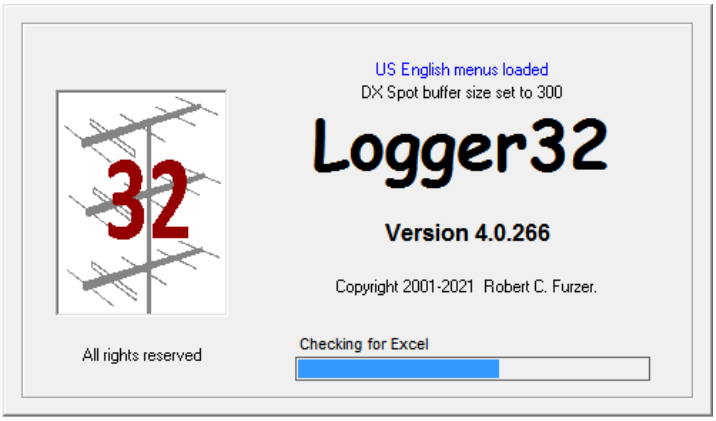


- A spinning pin-wheel image appears as Logger32 closes and then re-opens itself, running the updated version.

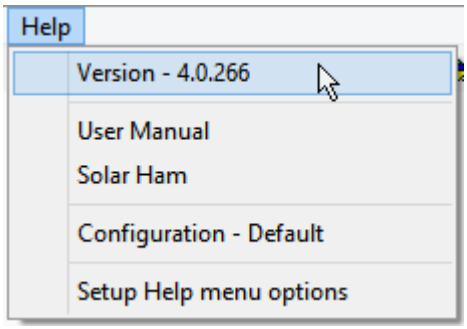


Copyright © 2026 Robert C. Furzer K4CY

The startup splash screen ▼ briefly tells you which version is now running[47].

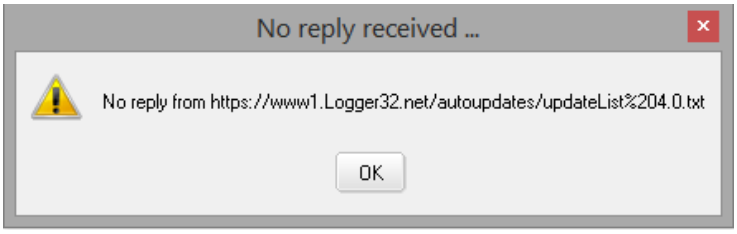


You can also check the current version number under **Help** ▼



2.10.1 Auto-update problems

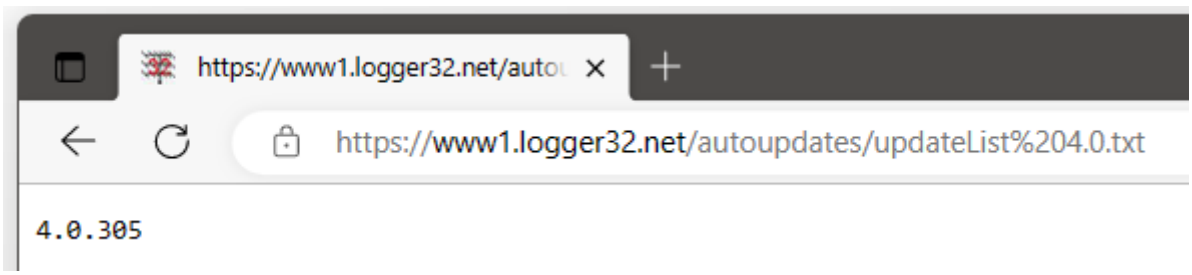
If Logger32's automatic updates fail, the following suggestions *might* just work for you.



◀ If your Internet connection is as unreliable as mine, you will see a message like this from time to time.

The *No reply received …* message tells us Logger32 tried but failed to grab a small text file from the Logger32 website containing the current version number.

- If that happens, check your Internet connection and, when it is working, try it again. Simply **repeat the auto-update process**. Internet connectivity or performance issues occasionally slow, interrupt or corrupt the downloading of updates. Usually, these are only temporary glitches unless you live in the Back Of Beyond … in which case you are probably well aware of the problems.
- If auto-updates fail repeatedly, click this underlined HTTP hyperlink right here in the User Manual: does it open in your browser? If the PDF reader is working properly on your PC, the link should open a mostly blank web page in your default web browser, with just the current Logger32 version number shown in the top left corner, something like this ▼

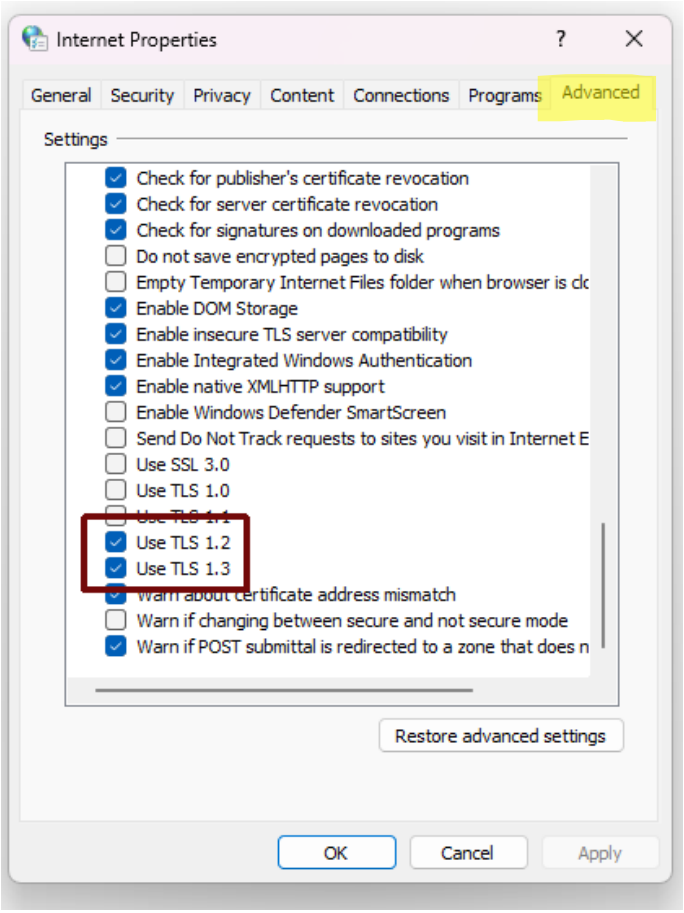


[47] Occasionally, on some systems, the update process fails for some reason (*e.g.* if your firewall or antivirus software blocks the downloading and/or unpacking of the update files). If the problem persists, you *may* need to add exceptions to the security software, or download, unpack and update Logger32 manually.

Copyright © 2026 Robert C. Furzer K4CY

- Notice that the current Logger32 version number is shown (4.0.305 when that screenshot was taken in April 2023). Logger32's auto-updater function magically determines whether a new version of Logger32 is available by comparing the version number stated in the downloaded text file against the version number of Logger32 currently executing.
- Notice also that the http:// URL in the hyperlink was automatically redirected to https:// by the web server, hence the padlock icon is shown. The browser address field may show the full URL including the https:// bit. Like most current websites, the Logger32 web server *insists* on using TLS version 1.2 or later encryption. If it doesn't open the page, you may need to check/enable TLS 1.2 and 1.3 on your PC:
  1. Open the Windows Control Panel.
  2. Click <**Internet options**>.
  3. Click to open the <**Advanced**> tab.
  4. Scroll down to the bottom of the settings:

     <**Use TLS 1.2**> and <**Use TLS 1.3**> should both be ticked[48] ▶
  5. Click <**OK**> when you're done.



  - If your computer does *not* support TLS 1.2 (*e.g.* an ancient version of Windows that only supports SSL or an older version of TLS), you are bang out of luck. Sorry. You won't be able to auto-update Logger32, and your creaky old computer is vulnerable to privacy and security compromises. Think yourself lucky you are even able to read these very words. You are running on borrowed time.

- If you don't see a page like that, try carefully retyping or copying-and-pasting the following HTTP URL directly into your browser's URL address field, exactly as shown:

  `http://www1.logger32.net/autoupdates/updateList%204.0.txt`

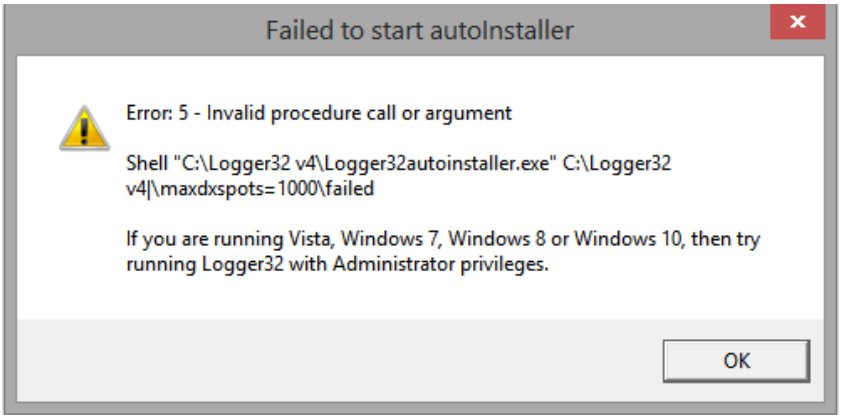
  Does it show the mostly-blank web page with just the version number now?
  - If so, it seems that something (a firewall or your antivirus package, probably) is blocking Logger32 from accessing the web page. Check the security configuration on your PC.
  - If not, try browsing to the Logger32 website's home page https://www.logger32.net If you cannot access the Logger32.net website at all, that suggests a more fundamental problem with the web server or your Internet access such as a DNS or firewall issue … but we've taken you as far as we can here. Sorry, you're on your own now. GL.

[48] SSL and earlier versions of TLS are insecure, deprecated and should be *un*ticked.

Copyright © 2026 Robert C. Furzer K4CY

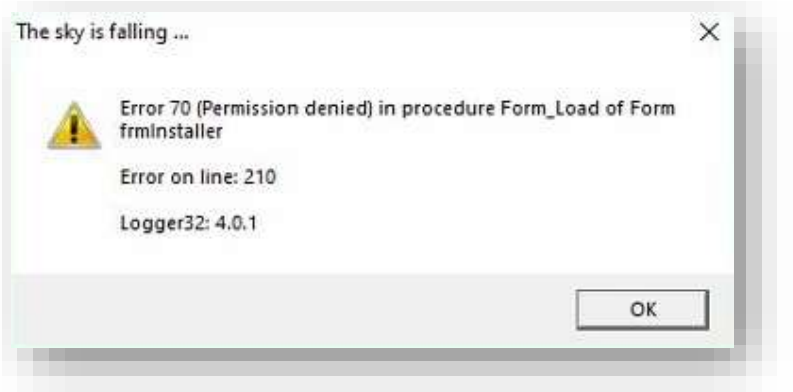
- If your Internet connection is working OK and the auto-updater can access the version number text file, something else is causing problems.

An error message along these lines is a big clue ▶

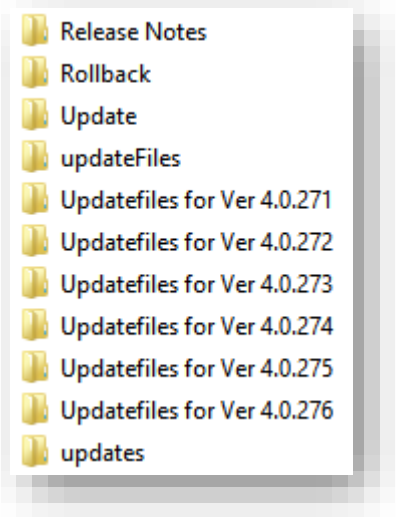


- In this case, configure Logger32 in Windows to run as administrator and try again to auto-update.

- 'Permission denied' errors suggest that *Logger32.exe* or *Logger32autoInstaller.exe* are running without administrative privileges, or some other cause ▶



Updating Logger32 involves several folder and file operations, resulting in a folder structure like this under *C:\Logger32* ▼



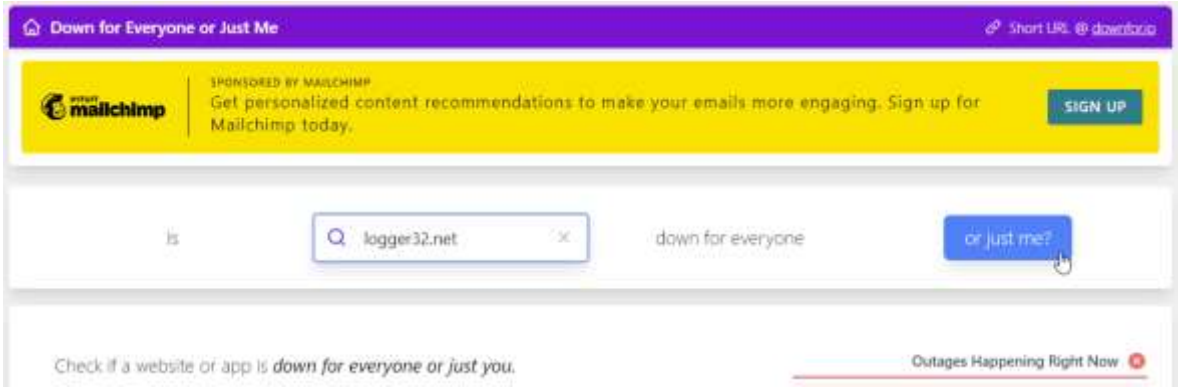
Try moving the relevant folder/s from *C:\Logger32* somewhere else such as *C:\Logger32\DeleteMe* (just in case you ever need them) … or simply bite the bullet and manually delete them all.

When the process works properly, your system will recreate the structure during future auto-updates, saving the release notes, rollback files, update files *etc*.

Hinson tip: all the folders shown here are safe to delete, but don't get *over*-enthusiastic with the delete button!

Copyright © 2026 Robert C. Furzer K4CY

- Other things to try if the auto-updates *still* aren't working properly:
  - Confirm that the Logger32.net website is currently running and accessible from your PC by using DownForEveryoneOrJustMe.com or the shortened form downfor.io. Type logger32.net into the query box and click the blue *or just me* button to run the check ▼



  - Add *C:\Logger32\Logger32.exe* and *Logger32autoinstaller.exe* or even the entire *C:\Logger32* folder as exceptions (whitelist) in your antivirus software.
  - Turn on Logger32's debugging function, watch the screen carefully as you try yet another auto-update, and then ask for assistance through the Logger32 reflector, explaining what happens. If errors are generated within Logger32 (as opposed to the auto-installer), the exact line of code that fails is identified with the debugging error trapping enabled.

Read the text file

C:\Logger32\Auto-update progress.txt

for further clues about what happened ▶

```
Building C:\Logger32 v4\Updatefiles for Ver 4.0.187
Update archive created.
Deleting C:\Logger32 v4\Rollback files
C:\Logger32 v4\Rollback created
26 INI files copied to C:\Logger32 v4\Rollback
Deleting \updates folder
Downloading version 4.0.188
.  Watch the progress bar
File will be loaded in 70 64kB chunks
4547262 bytes transferred
Unzipping downloaded file
Unzipping file: LOGGER32.EXE
Unzipping file: LOGGER32ROLLBACK.EXE
Unzipping file: AUTOINSTALLER.EXE
Unzipping file: QRX.EXE
Unzipping file: ADIFBANDS.TXT
Unzipping file: ADIFMODES.TXT
Unzipping file: CW.EXE
Completed : 7 files unzipped
Now going to auto-backup (if enabled) and close Logger32
```

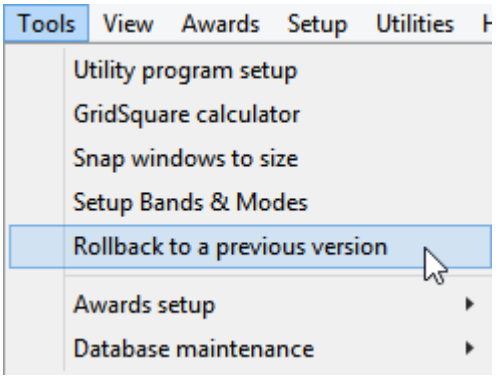
- If you are *still* having trouble, by now beginning to lose all hope that auto-updates will *ever* work for you, try to download Club Log's LoTW users or prefix exceptions files, or load the Keplerian elements, or set the PC clock through Logger32. All of these operations use essentially the same network mechanisms as the auto-updates.
  - If all the diagnostic checks fail, there is evidently something fundamentally wrong with the network mechanisms on your computer (in software, firmware or hardware) or on the network itself (probably a connectivity problem on your shack LAN, possibly something wrong at your ISP, conceivably the Internet itself is broken … in which case Logger32's auto-update failures are the least of your worries!).
  - If the diagnostic checks are OK but auto-updates *still* fail, you appear to have discovered an obscure bug in Logger32 … so report it on the Logger32 reflector and wait patiently for further instructions. Meanwhile, try turning-it-off-and-on-again, or sacrifice a goat.

Copyright © 2026 Robert C. Furzer K4CY

If you are plagued by issues, here's a trick everyone should know by now … Roll back to the previous version. You are now using a known good Logger32.exe, and a known good Logger32.ini. If everything is working as it should, then auto-update. If the problem persists, then simply reinstall Logger32 over the top of what you have, then auto-update.

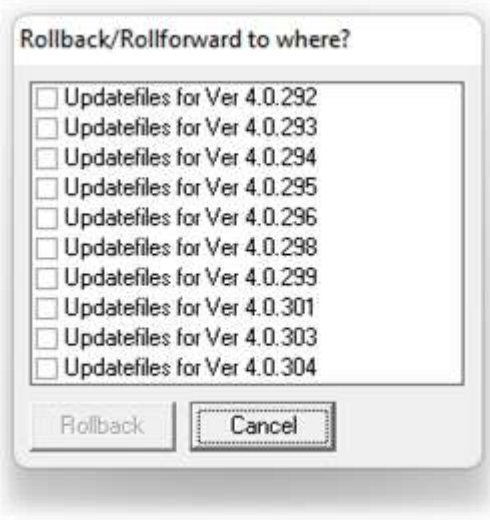
Bob K4CY

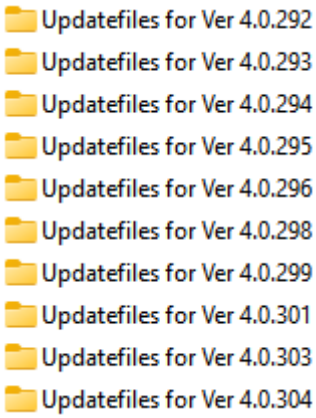
2.10.2 How to rollback *automatically* to a prior version



◀ If Logger32 does not work properly after an update, you can rollback (revert) automatically to a prior version using **Tools ⇨ Rollback to previous version** in Logger32, *provided* Logger32 doesn't crash and burn before you get the chance to use the menus and functions, in which case you can rollback manually instead.

Click to tick on the list whichever version you now wish to load, then click <**Rollback**> to set it loading ▶





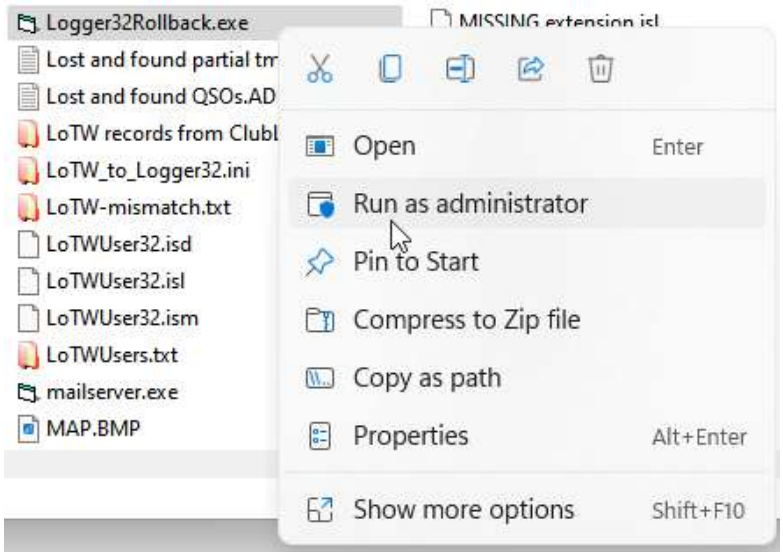
◀ Up to 10 prior versions are available, stored in the corresponding *Updatefiles* folders under *C:\Logger32*

The menu item runs *C:\Logger32\Logger32Rollback.exe*, instructing it to copy the contents of the chosen *Updatefiles* folder into *C:\Logger32*.

Hinson tip: in addition to the previous subversion's *Logger32.exe* program, auto-update also saves a copy of *Logger32.INI* and other configuration files in *Updatefiles* … so that's another backup in case you later mess up your configuration. If something that used to work has stopped working and you can't figure out why, rolling back through the versions one-by-one will hopefully revive it. Auto-updating from that point will skip any intermediate versions, taking you directly to the very latest *Logger32.exe* but with your original (working) configuration – which hopefully works OK.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: if the automated rollback fails, try running *Logger32Rollback* with Windows administrator privileges. To do that, press <**Win+E**> to open File **E**xplorer, navigate to the *C:\Logger32* folder, right-click the *Logger32Rollback.exe* file to open a menu, then click <**Run as administrator**> ▶



I did the manual update to v273, but still no joy. Rolled back to v269 and everything is good. This may well be a problem unique to me, since Bob can't replicate it. The rollback feature is great – very helpful.

K4DR

2.10.3 How to rollback *manually*

You know you are having a *really* bad day when, not only do you miss out on working P5DX, but a Logger32 auto-update fails spectacularly, such that the newly-updated Logger32 won't even start up properly or it crashes before you can open the menu to launch an automated rollback.

Before pressing ahead, perhaps making a rotten situation even worse, take a moment to reflect quietly … while you *reboot* your computer[49].

After the computer reboots and settles down, but *before* you run Logger32 again, **make a safety copy of your entire Logger32 folder** (usually *C:\Logger32*) including your logbook and Logger32's backups, to removeable media such as a CD-ROM or USB memory stick, then physically remove the backup media from the computer to a place of safekeeping *e.g.* under the mattress. That way,

[49] Turn it off and back on again is more than just a wry joke about inept IT support. A simple reboot fixes things surprisingly often, resetting the entire system to a stable state. Apart from applications and utilities that autostart at boot time, the computer should be relatively quiescent after booting. A reboot stops any runaway processes and may also complete the installation of Windows or other program updates, new drivers *etc*. **It's worth a try!** Things *could* be much worse: the dreaded **B**lue **S**creen **O**f **D**eath means your system won't even boot cleanly. Safe Mode might just let it boot … otherwise it's a full operating system rebuild for you, reinstalling Windows. Those backups you *hoped* you'd never need may save the day, unless your computer is totally bricked, in which case you have our sincere condolences.

Copyright © 2026 Robert C. Furzer K4CY

there's a pretty good chance your entire logbook/s and Logger32 setup can be recovered to its present state *even if* whatever you do next totally screws things up.

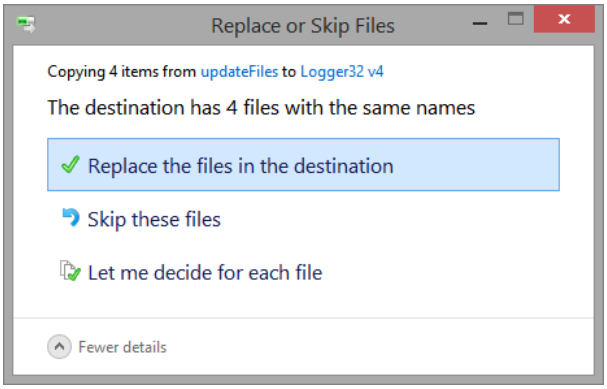
Hinson tip: trust me, accidents happen, even for those of us who ought to know better. This warning comes to you courtesy of the School of Hard Knocks. You're welcome.

Now try starting Logger32 once again, and running the automated rollback function following the earlier instructions. It may work this time! In fact, having rebooted, the updated version of Logger32 might even work so well that you don't actually need to rollback, so check that first.

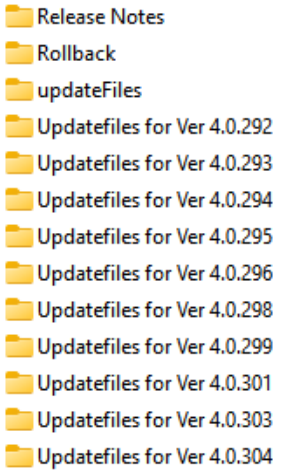
If you still have problems and *need* to rollback manually, here's how:

- Close Logger32 if it is running. Wait patiently for it to shut down fully.
- In File **E**xplorer, navigate to your Logger32 folder (probably *C:\Logger32*).
- You should find a subfolder in there called Rollback containing a copy of your *C:\Logger32* folder contents immediately prior to the latest auto-update. Double-click to open it.
- Select everything in Rollback using <**Ctrl**+**A**> then copy it using <**Ctrl**+**C**>.
- Click the left arrow or use <**Alt**+**LeftArrow**> or click the parent folder to go back to the Logger32 main folder.
- Paste all the files and folders from Rollback using <**Ctrl**+**V**>.
- Windows will ask you whether to overwrite files with the same names ▶

  If you are happy to proceed, happily click <**Replace the files in the destination**>.



- That's it! Start Logger32 in the normal way. You should now be running the version *prior* to the failed auto-update.
- If Logger32 *still* won't run properly, you could try rolling back manually to even earlier versions using the contents of the *Updatefiles for Ver 4.x.xxx* folders instead of the *Rollback* folder ▶
- As a last resort, download and re-install Logger32 version 4 from scratch, and *hope* that it runs OK! With luck, it may even auto-update itself successfully to the latest release. If so, you can restore your log/s and *.INI*s from backups into *C:\Logger32* to pick up where you left off … unless it was one of those *.INI*s that caused your problems in the first place, in which case you'll have to reconfigure Logger32 manually.

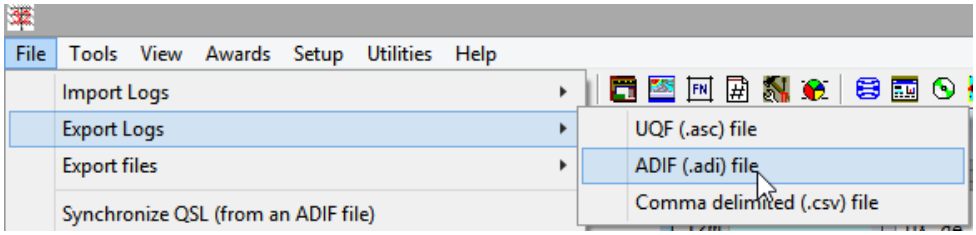


Good luck.

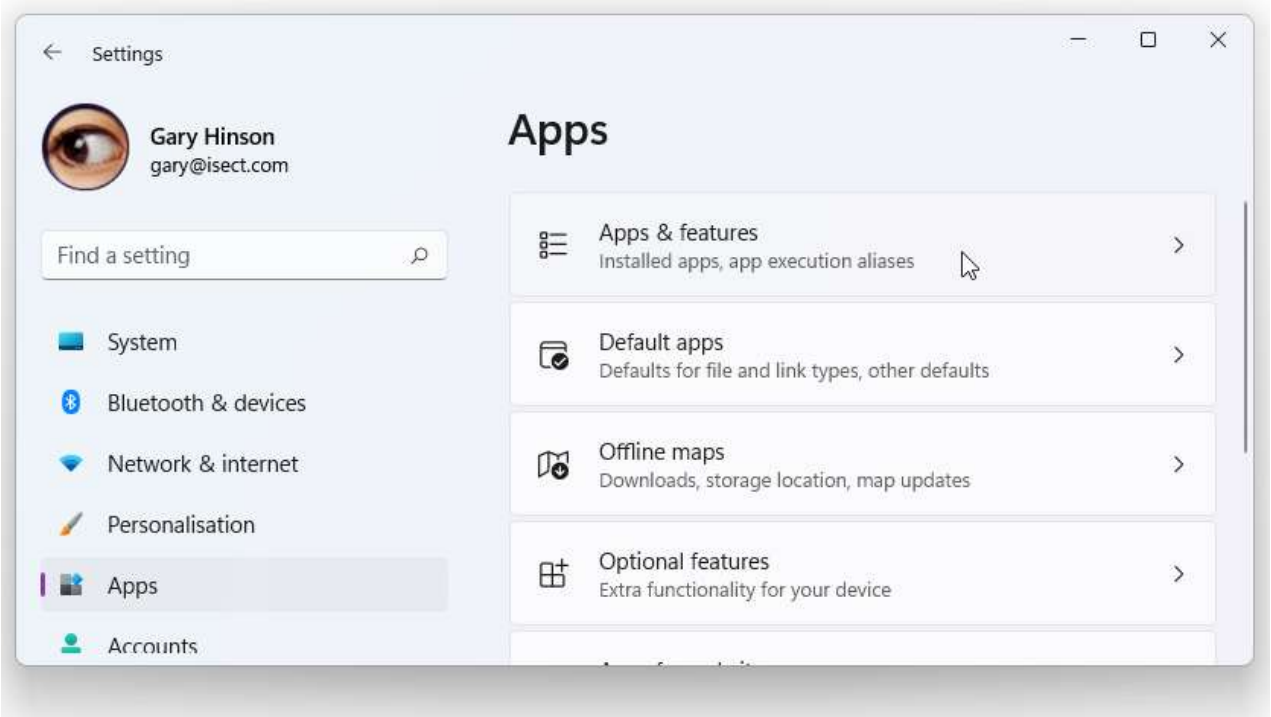
Copyright © 2026 Robert C. Furzer K4CY

2.11 How to *un*install Logger32

If possible, run Logger32 for one last time. Export your log in ADIF format[50] using **File ⇨ Export Logs ⇨ ADIF (.adi) file** ▶



Again, if possible, using **File ⇨ Change Logbook** open any other Logger32 logs in turn, exporting each of them as ADIFs too in the same way. Close Logger32, then uninstall Logger32 using the **Apps** pane, **Apps & features** function in [**Win**+**I**] ⇨ **Settings** (Windows 11). ▼



In Windows 8.1 and Windows 10, use the **Programs and Features** pane in the Windows Control Panel. If you don't see Programs and Features listed, click <**Small icons**>.

Either way through Apps & features or Programs and Features, scroll down then click to select the Logger32 entry, then click <**Uninstall**>.

Hinson tip: we're truly sorry if Logger32 didn't work out for you. Good luck finding a replacement logger that meets all your requirements. If that mission doesn't exactly go to plan, you can always come crawling back to Logger32 with your ADIF logs in hand. We won't say a word - promise.

[50] Save the ADIF file/s somewhere safe. That way, should you ever need to restore, check or update your log/s for some reason, you won't have to re-install and run Logger32 just to access the proprietary database format that only Logger32 understands: *any* ADIF-compatible program (meaning almost any logger) *should* be able to open and read the saved ADIF file/s just fine.

Copyright © 2026 Robert C. Furzer K4CY

2.12 Installation and updating FAQs

Q. When I launch Logger32, it starts up but then disappears without a trace – no error message, no annoying 'ding', no clues: it simply vanishes. What now?

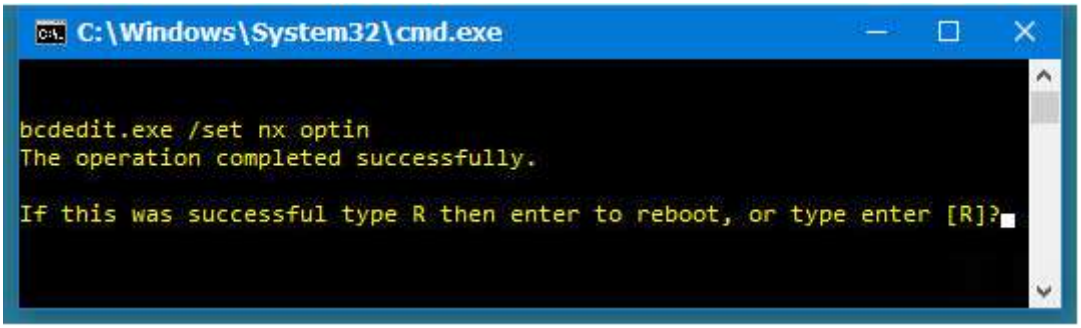
A. Most likely Logger32 is triggering a Windows security feature called **D**ata **E**xecution **P**revention. DEP is *meant* to prevent malicious or buggy software from executing by designating certain memory ranges as 'data only' (no execution). However, it has a hair-trigger and suffers false-alarms on *some* PCs[51].

Read about DEP here.

To get around this, disable DEP using any one of the following *four* approaches:

Option 1 – simplest of all … *provided* you have previously run and updated Logger32

1. Use <**Win+E**> to open **Win**dows **E**xplorer.
2. Navigate to your Logger32 folder – by default, that's *C:\Logger32*.
3. Double-click to run the Windows batch file called *Change DEP.bat* ▼

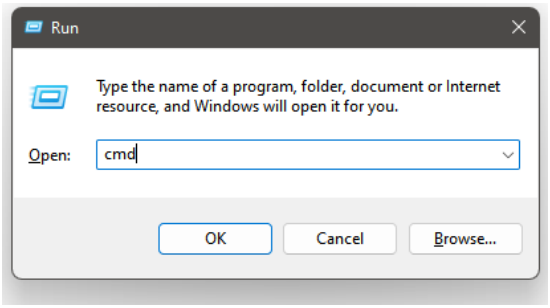


4. Do your best to follow the cryptic instruction ▲

Option 2 – almost as simple

Option 1 automates the following commands but you can also execute them manually:

1. Use <**Win+R**> (press the Windows key on your keyboard and tap the R key) to open the **Win**dows **R**un box ▶
2. Type **cmd** into the Run box, then press <**Ctrl**+**Shift**+**Enter**> to run the command processor as an administrator[52].

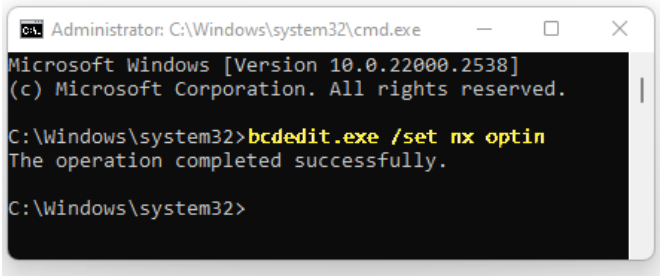


[51] A few dozen users have had to disable DEP in order to run Logger32. We don't know what makes them so special, and we're frustrated because we cannot reproduce and so diagnose or fix the problem: *none* of the Logger32 beta/support crew have suffered the DEP issue, so far. We blame Bill Gates.

[52] If you just hit plain <**Enter**> by mistake, you will not have picked up the administrative privileges necessary to disable DEP … so close the unprivileged, downtrodden command processor and have another go.

Copyright © 2026 Robert C. Furzer K4CY

3. If asked, read and accept the security warning by clicking the <**Yes**> button ▶ to open the command processor ▼

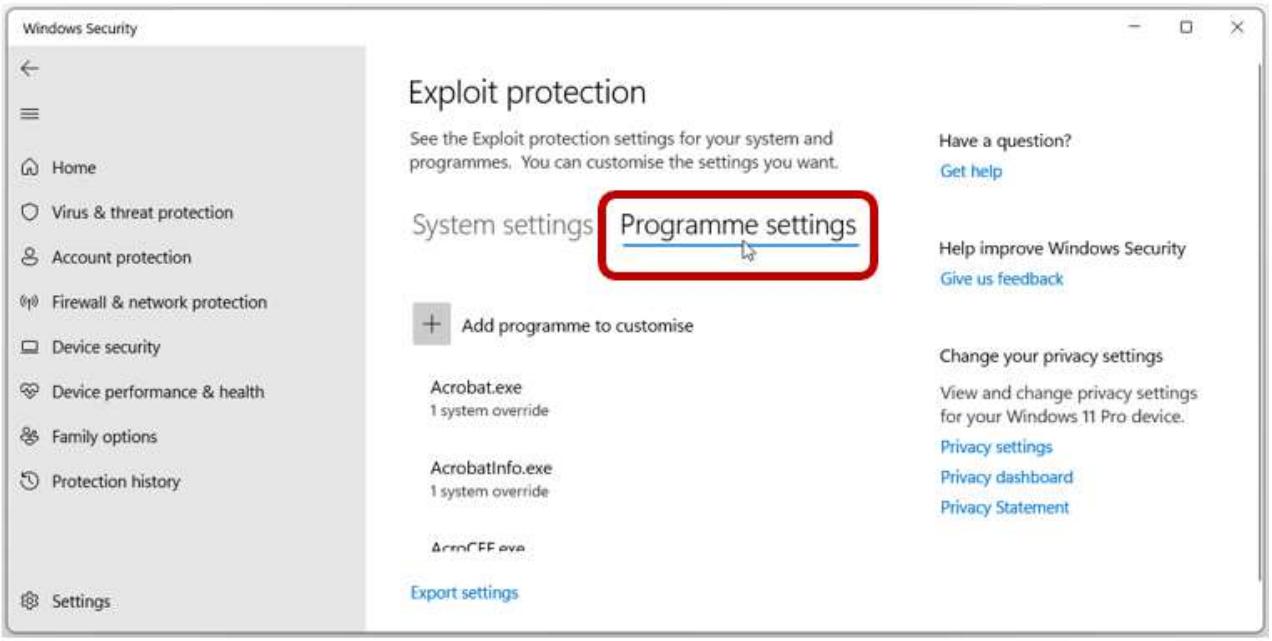




4. Into the command processor ▲ type **bcdedit.exe /set nx optin <Enter>**
   You should shortly see "The operation completed successfully", a not-exactly-helpful message indicating that DEP has been disabled.
5. Type **exit <Enter>** or click the corner X to close the command processor.
6. Reboot to pick up the new DEP-off setting.
7. Check and hopefully celebrate your ability to run Logger32 normally. If it didn't work and you *still* can't run Logger32, re-enable DEP as that was evidently not your problem. To do that, use the command **bcdedit.exe /set nx optout** to reverse the change, and reboot.

Option 3 – the safest: disable DEP just for Logger32 (*and* CW machine if necessary)

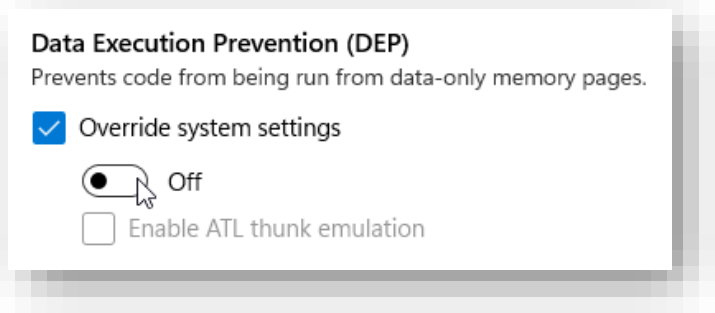
1. Tap the Windows key on the keyboard or click the Start button on the screen.
2. Type **Exploit protection** <**Return**>.
3. Click to open the <**Programme settings**> tab in the middle ▼



4. Click the + button to the left of <**Add programme to customize**>.
5. Click <**Add by programme name**> then type *Logger32.exe* and click the <**Add**> button.
6. Scroll down to the Data Execution Prevention (DEP) section.

Copyright © 2026 Robert C. Furzer K4CY

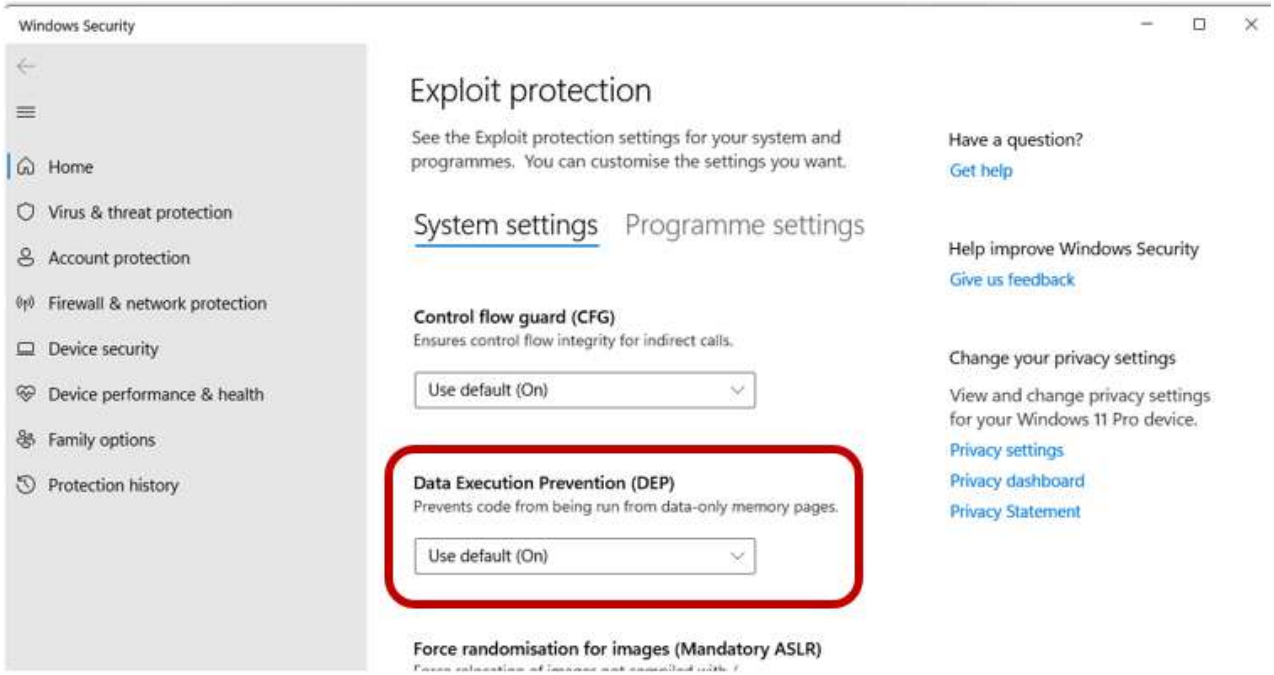
7. Click to tick <**Override system settings**>.
8. Click to set the slide-switch to <**Off**> ▼ … which disables DEP but *only* for Logger32.



9. Click the <**Apply**> button at the bottom.
10. Click <**Yes**> to acknowledge the Windows security warning.
11. Go back to step 4 and **repeat the process for *CW.exe*** if you find that Logger32 also crashes when you run the CW machine, otherwise skip it and go on to step 12.
12. Click the corner X to close the DEP function.
13. Check whether Logger32 now loads and executes normally. If not, reboot to make sure DEP is truly disabled, then check Logger32 again. If you *still* can't run Logger32, re-enable DEP by setting the slide to <**On**> as DEP was evidently not your problem … and consult the stars for inspiration. Try magic crystals, yoghurt, chicken bones or a sacrificial anode.

Option 4: disable DEP *completely* – potentially risky

1. Tap the Windows key or click the Start button.
2. Type *Exploit protection* <**Return**>.
3. In the Data Execution Protection (DEP) section, click the selector ⌄ button and change the setting from <**Use default (On)**> to <**Off by default**> ▼



Copyright © 2026 Robert C. Furzer K4CY

4. Check whether Logger32 now loads and executes normally. If not, reboot to make sure DEP is truly disabled, then check Logger32 again.
5. Finally, decide whether you are or are not willing to take the risk of leaving DEP completely disabled on your PC.

And, to be on the safe side, wear safety goggles when sitting at the keyboard: it could blow at any moment

Bob K4CY

Hinson tip: DEP is just one of several system security controls. Personally, being a professionally paranoid information security geek by day, I rely heavily on commercial antivirus software and risk-avoidance, *plus* DEP since I don't experience this issue with Logger32 on my PCs. If you decide to disable DEP completely and leave it disabled, you are even more reliant on other security controls. Be careful out there.

Hinson tip: some users report that DEP is re-enabled by Windows updates, so if Logger32 resumes crashing, you might like to check those DEP settings again.

Hinson tip: if, like me, you are unwilling to leave DEP disabled but you want to run Logger32, you might try running Logger32 in Windows 7 or 8 compatibility mode or fiddling around with Windows internals, perhaps running the **s**ystem **f**ile **c**hecker from an elevated command prompt and maybe deinstalling and reinstalling the .Net framework for good measure. If those cryptic hints leave you bewildered and none the wiser, don't even attempt them. This *Logger32* User Manual is not about to map out Windows internals for you, partly because I'm lost and bewildered but largely because it is dangerous territory. There be dragons.

Q. I want to save my logs to the cloud so I can access them from anywhere. Can I do this with Logger32?

A. Yes … and no. The way Logger32 uses the Microsoft database subsystem to manage your log seems to cause problems for cloud synchronization – essentially, there are conflicts when multiple apps attempt to open and read/write the same log files.

The 'solution', such as it is, is to have Logger32 log to a file on your hard drive or some other local drive, one that it can open and hog *exclusively* while it is running. To copy, synchronize or backup the log files to cloud storage (*e.g.* Dropbox, OneDrive, Google Drive, Box …), you'll have to close Logger32 first. It might be worth adding the relevant post-execution commands to the end of a Windows script (a batch file) that launches Logger32.

Q. My Logger32 is at v3.5*something*. The stupid auto-updater tells me there are no updates, which I know is a flat lie … so how *do* I get Logger32 v4?

A. See the migration section of this very chapter!

Copyright © 2026 Robert C. Furzer K4CY

Q. I gather there is a new update for Logger32 v4 available but for some reason the auto-updater won't download it. What do I do now?

A. Manually download the update file for a specific version from:

`https://www1.Logger32.net/autoupdates/4.0.NNN%20Update.zip`

… where **NNN** is a recent v4 sub-version number *e.g.* 324. Then double-click to open the zip file and extract the contents to *C:\Logger32*, overwriting the existing files.

If that scares you, make a backup of the folder first – no, hang on, backup the folder anyway. You *should* be scared. Overwriting stuff *is* scary.

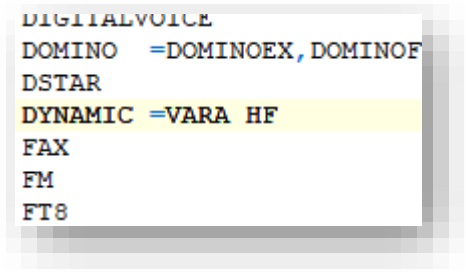
Q. Can I add a new mode or submode to Logger32?

A. Yes. There are several steps. **Be sure to complete the sequence in the order shown.**

1. Check the current ADIF specification to determine how the mode **or** submode is defined[53].

> **Hinson tip**: do not accidentally define it as **both** a mode **and** a submode!

2. Open *C:\Logger32\ADIFmodes.txt* in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad. Add the new mode(s) or submode(s) - one per line, in the same way as the original modes (and =submodes, if any). Save the edited *ADIFmodes.txt* file into *C:\Logger32* and, preferably, save backup copies wherever you routinely back up your Logger32 *.INI*s *etc*. (your user files) and logs.



For example, insert the line:
DYNAMIC *[space or tab]* =VARA HF
to your *C:\Logger32\ADIFmodes.txt* file ▶

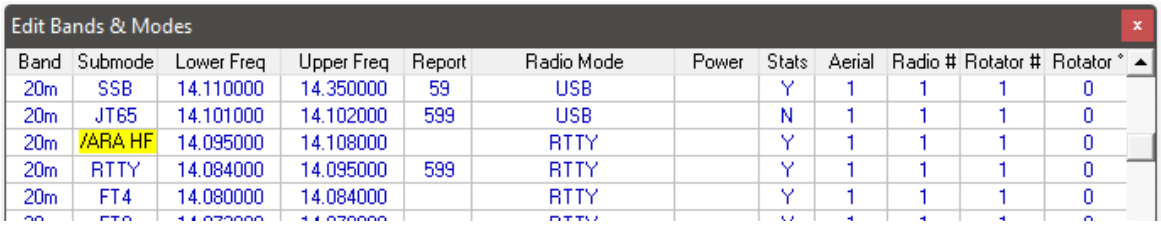
> **Hinson tip**: if, later on, you experience strange problems with bands, modes, award reports or statistics, it *may* be that *ADIFmodes.txt* has somehow been replaced with the unedited original/default version or corrupted. It turns out, missing equals signs in some of those mode groups on my system were the root cause of a broken QSO detail report within the DXCC_MIXED award (not DXCC_CW or others, oh no). This obscure problem *almost* out-foxed us, but for additional error checks Bob inserted in the code to diagnose it.

3. Since *ADIFmodes.txt* is only accessed when Logger32 starts up, close and restart Logger32 to pick up any newly-defined mode/submode(s) in your edited file.

[53] It inevitably takes a while for novel modes to stabilize and be incorporated into the ADIF standard. If you simply can't wait, log your novel mode QSOs with a similar mode or something distinctive (such as SSTV) temporarily, then update them when ADIF is updated. Keep a note of which QSOs they are though!

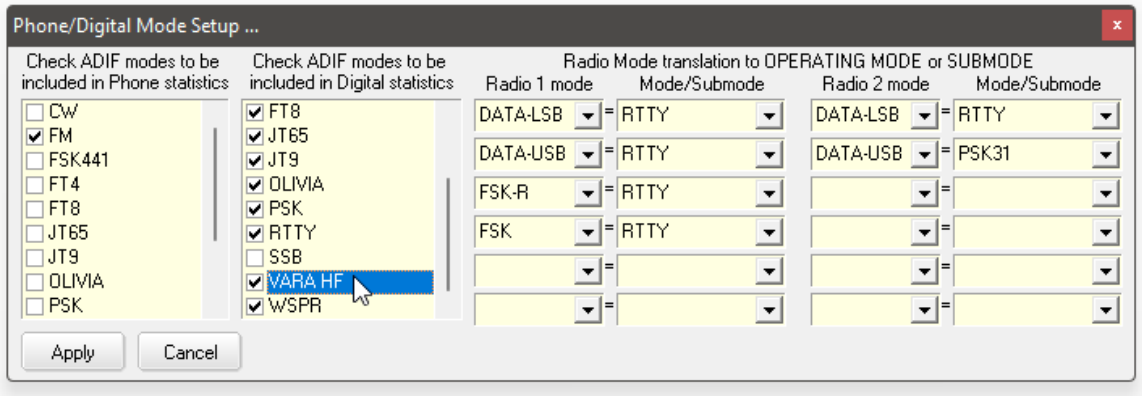
Copyright © 2026 Robert C. Furzer K4CY

4. **Tools ⇨ Setup Bands & Modes** and add the relevant 'watering hole' frequencies or ranges with the new mode/submode(s) specified in the <**Submode**> column. Optionally put Y in the stats column for at least one of the new watering holes to start recording statistics whenever you use the new mode/submode(s), and don't forget to click <**Apply**> to save the changes. For example ▼



Hinson tip: users of novel digital modes typically jostle for band space among users of the established modes, hence the corresponding watering holes/subbands can take a while to become established … or not, if the modes don't work out. So, be prepared to check/edit the Bands & Modes table accordingly, particularly if you are keen on experimenting with digimodes.

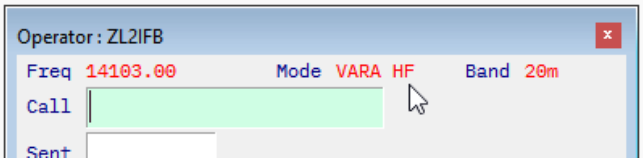
If you want QSOs using the new mode(s) to be included in your phone or digital statistics, open **Tools ⇨ Database Maintenance ⇨ Setup Phone Digital Modes** and tick the



appropriate mode/submode(s). Again, don't forget to click <**Apply**>. For example ▼

5. Recalculate your statistics to have Logger32 set up its internal statistics tables and pick up any QSOs already logged with the new mode/submode(s).

6. *[Optional]* QSY your radio to a watering hole to check that the new mode/submode is shown at the top of the log entry pane (assuming you are using <**Mode from BandPlan**>) *e.g.* ▶



Copyright © 2026 Robert C. Furzer K4CY

7. *[Optional]* Feel free to add other modes and/or submodes precisely as they are specified in the ADIF specification if you are already using them or intend to give them a go at some point. Look at the mode and submode enumeration tables in sections III.B.9 and III.B.10 of the ADIF standard, ignoring those marked "(import-only)". New modes/submodes are invented from time to time - ADIF version 3.1.4, for instance, added submodes FREEDV and M17 to mode DIGITALVOICE, while VARA HF is just one of four submodes of the mode DYNAMIC. Q15 is defined as a mode, whereas Q65 is an MFSK submode[54]. Logger32 reserves space for ***up to* 200 modes/submodes** and warns you if you define *more than* 200 in *C:\Logger32\ADIFmodes.txt* – in which case the 201st and any subsequent modes/submodes from the file are skipped and hence cannot be used in Logger32.

> **Hinson tip**: while it may be possible to import and log or edit QSOs to show any mode and submode, Logger32 doesn't 'know' about any modes and submodes that are not listed at all in your Bands & Modes table. Therefore, it cannot add the relevant QSOs to its statistics for applicable mode-specific awards. So, avoid deleting modes and submodes from your Bands & Modes table unless you are sure you have not logged any such QSOs, or don't care if they don't count towards your statistics.

Q. How do I add a new band to Logger32?

A. First edit *C:\Logger32\ADIFbands.txt* using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad, to include the new band. You can then add the new band and define mode sub-bands in the Bands & Modes table.

> **Hinson tip**: see the previous answer for more advice. Adding a new band is much the same as adding a new mode, requiring similar steps to avoid later issues.

Note: in order to retain any user edits, Logger32's auto-update function saves copies of your *ADIFbands.txt* and *ADIFModes.txt* files to the *C:\Logger32\\Rollback* folder and does not overwrite them with the default files. Both files are also included in user file backups.

> **Hinson tip**: if you are adding a 'spot frequency' rather than a conventional band, define the lower frequency as the spot frequency and add the permitted transmission bandwidth to get the upper frequency. The reason is that, when you are using a digimode such as FT#, Logger32 wants to log the VFO frequency *plus* the audio offset up to ~3 kHz higher. The calculated frequency needs to be within the range defined in your Bands & Modes table to avoid out-of-band warnings and interfere with parallel logging. You still need to be careful not to transmit out of band, though, so carefully check the wording of your license and stick to the defined limits – for instance, avoid setting your transmit frequency way out on the right hand (HF) side of the FT# waterfall. Your license may even specify permitted/prohibited modes.

Q. I messed up! My ADIF modes file is trashed! *Help!*

A. First, hunt for an un-trashed version of the file squirreled away somewhere. Look through your Logger32 'user files' backups, or whatever other backups you have made periodically. Copy any *ADIFmodes.txt* files you find to a temporary folder, sort them by date and check the most recent one to see if it is pristine (un-trashed) and usable. If it is, save it to *C:\Logger32*. If not,

[54] Don't shoot the messenger! When the ADIF committee changed its naming policy, it wisely chose not to re-name previously-defined modes and submodes to avoid breaking Logger32 and other, lesser, apps.

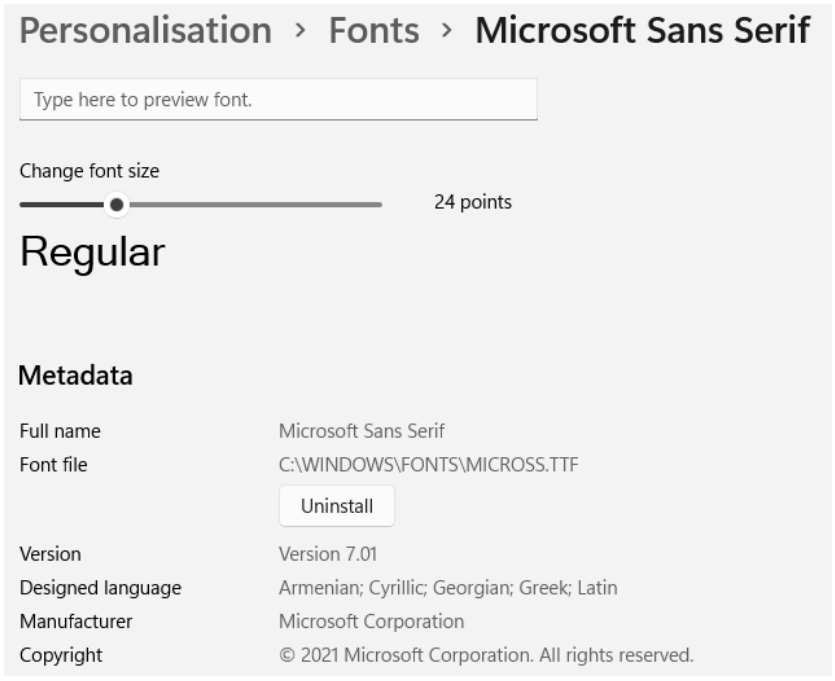
Copyright © 2026 Robert C. Furzer K4CY

keep checking ever older ones until, hopefully, you hit pay-dirt. If not, login to Google Groups then look in the files area of the Logger32 reflector for *ADIFmodes.txt* and *ADIFbands.txt* files. Although they are incomplete, they offer a fresh starting point, a basis for you to start customizing them again, this time hopefully being more diligent about those backups.

Q. Help! After installing Logger32 on a new Windows 11 PC, I get "Error 380 (invalid property value) in procedure setup picture border of module MISC" when it runs. I have no idea what that means … and Logger32 hangs …

A. That cryptic error results from the apparent lack of "Microsoft Sans Serif" font on the PC, a font that is required for captions (titles) to the windows that Logger32 displays. So, first check your installed Windows fonts to see if it is present:

1. Press <**Win+i**> to open **Win**dows Sett**i**ngs.
2. Click <**Personalisation**> on the left menu.
3. Click to open the <**Fonts**> option near the bottom of the list (that's where it is on my Windows 11 PC anyway).
4. Under **Available fonts**, type *microsoft sans serif* into the search box (case insensitive).
5. If the Microsoft Sans Serif font is found, click the font name to check that it opens OK and is usable. You should see something roughly like this, although the layout and details such as the font file location and version number will differ in other versions of Windows ▶



6. If the font is missing, you will need to obtain and install "Microsoft Sans Serif" font – that specific font with that exact name, not just some font with a similar name. According to Microsoft, "Microsoft Sans Serif" font is shipped with most versions of Windows back as far as Windows 2000, so it *should* be on almost every Windows PC unless someone has carelessly uninstalled and deleted it, or it has been corrupted and lost *e.g.* by a malware infection. It should be somewhere in the Windows installation files. You may be able to track down a downloadable version on the Web or on another Windows PC but beware dubious websites with malware-infested downloads. As a last resort, Microsoft will offer to sell it to you for about ^{US}40 (don't shoot the messenger: it is their intellectual property after all).

More info at https://learn.microsoft.com/en-us/typography/font-list/microsoft-sans-serif

Copyright © 2026 Robert C. Furzer K4CY

Q. Oh no! I get "Runtime error 5 - invalid procedure or argument" and Logger32 dies. What now?

A. That error *strongly* suggests a permissions problem: Windows or perhaps some other app (such as your antivirus or real-time cloud backup software) is blocking Logger32 from accessing and updating a database file. Try running Logger32 as an administrator to see if that solves it. If not, try suspending or stopping your antivirus and/or cloud backup software temporarily instead. If you are still unable to run Logger32, open Resource Monitor (one of the built-in Windows system tools) and keep a close eye on the disk activity tab while you attempt to run Logger32 again. Is some other program accessing one of the Logger32 database files at the critical moment when Logger32 tries to open it? Good luck!

Another simple option is to reboot your system before trying to run Logger32. Turnitoffandonagain. It *may* just work. Sometimes the system gets itself into a pickle. Rebooting will often un-pickle it.

Q. Loading and running Logger32 gets me "Runtime Error 13 - type mismatch". How do I fix this?

A. Error 13 indicates a problem within one of Logger32's internal databases.

First of all, close Logger32 (if it is still running) and in Windows make backup copies of your four logbook database files *C:\Logger32\mylog.isd, .isf, .isl* and *.ism* - where 'mylog' is the name of *your* logbook. Save them offline *e.g.* on a USB stick that you physically remove from the PC after saving the files. Even better, backup the entire *C:\Logger32* folder. That way, if something *else* goes wrong, at least you can recover to this point. You hopefully won't make it any worse. It's an insurance policy.

Now delete the four logbook database files mentioned above, restore the equivalent four logbook database files from your most recent Logger32 backup, then re-start Logger32. If that doesn't work (*i.e*. you get the Runtime Error 13 again), delete those four files and restore the four from the *previous* Logger32 backup … and so on, until hopefully you find a set that runs without the error.

If *none* of the backups work for you, delete the logbook database files and open Logger32: it will create a new, empty logbook. Now import your most recent ADIF log backup, and once Logger32 has populated its databases and recreated all its statistics, you're good to go.

Oh wait, hang on, you don't have a recent ADIF log backup? Woe is me. Do you have *any* ADIF log backup, maybe one made on your last birthday or new year's day? How about Club Log or LoTW or some other online log: have you recently uploaded your log? If so, can you retrieve an ADIF and restore that? If not, you have our sympathies. Your log is toast. **Do please tell all your ham friends about your dreadful loss because you didn't have good backups.** Hopefully, *your* sorry tale will persuade *them* to make backups, saving someone else from this terrible fate. Please don't blame Logger32. Log backups are *your* responsibility.

Q. What causes "Runtime Error 6 – overflow" so Logger32 hangs while starting?

A. Has someone been 'tidying up' the computer, by any chance, or moving Logger32 from one computer to another? The usual culprit is a missing font that Logger32 needs. Check your *C:\Logger32\Logger32.INI* file to make sure all the listed fonts exist in the *C:\Windows\Font* folder. Look out for any disk, folder or file names that need to be updated, too.

Copyright © 2026 Robert C. Furzer K4CY

Q. Oh oh! I get "db Open Error: VIS_DISK_ERROR". Sounds ominous! What now?

A. It seems you may have a PC hardware problem, perhaps a failing or failed hard drive. Here's what to do, step-by-step:

1. Don't panic: you have several recent backups stashed away in various safe places, right?
2. If your *only* log backups are on the failing disk drive, anxiously *try* to copy your log backups onto a working disk drive (*not* the failing one, obviously) such as a USB memory stick, an external disk drive, a CD/DVD disk or cloud storage (such as Google Drive, OneDrive or Dropbox). Then relax.
3. If you have no backups, go ahead and panic! Hopefully before your disk drive emits the magic smoke and finally expires, *try* running Logger32 then exporting your logbook as an ADIF file onto a working disk drive. Due to the disk error, Logger32 may not run at all. Any other valuable data on the disk may be history too. Tough luck. Lesson learnt.
4. Close Logger32 and *try* to copy your entire C:\Logger32 folder with all the Logger32 files[55] onto a working disk drive. Windows may well struggle to read the data. Wait patiently while it struggles. Try not to cry like a baby.
5. Tackle the hardware error. Is it real? Disk errors serious enough to cause problems for Windows and application programs are *generally* terminal, so you prefer to **replace the failing drive as soon as you can**. You *may* be able to squeeze a bit more life out of it using the disk management functions built-in to Windows and the disk drive itself, or separate data recovery software, but be prepared for it to fail completely at any moment. Newer disk drives – especially solid state drives – typically offer better performance at the same or better prices than the original magnetic disk drives. If your PC is so old that it has a floppy disk drive (remember them?), it is high time to replace the whole machine, disks and all.

   You *might* be lucky though. *Perhaps* the error was in fact caused by a cosmic ray, power glitch or something else that corrupted only the specific file noted in the error report – in which case, you might be able to recover an uncorrupted version of that file from the Logger32 installation pack or from your backups. GL …
6. Reinstall Logger32 on the replacement disk drive or computer. Either restore your log from a recent log backup or import the ADIF file into a new log, and check it for issues such as missing QSOs. If they have been uploaded, you *may* be able to recover lost QSOs from your online logs at LoTW, Club Log, QRZ *etc*.: good luck with that.
7. Having learnt your lesson, set up a suitable backup regime, making data backups routinely onto different storage devices, media and/or services.

Hinson tip: save the occasional 'last resort' archive copy of your log, preferably as an ADIF file onto, say, a CD or USB stick that you put somewhere safe – ideally a fire safe or bank vault if you feel that your log is truly valuable. If you archive your log securely just once a year (perhaps on New Year's Day or your birthday), the *most* you will lose is a year's worth of QSOs – still bad but not utterly disastrous.

[55] Conceivably, Logger32 may not be the only application on your PC, in which case you should probably make backup copies of all the other data on the failing drive as well, before it's too late. Good luck.

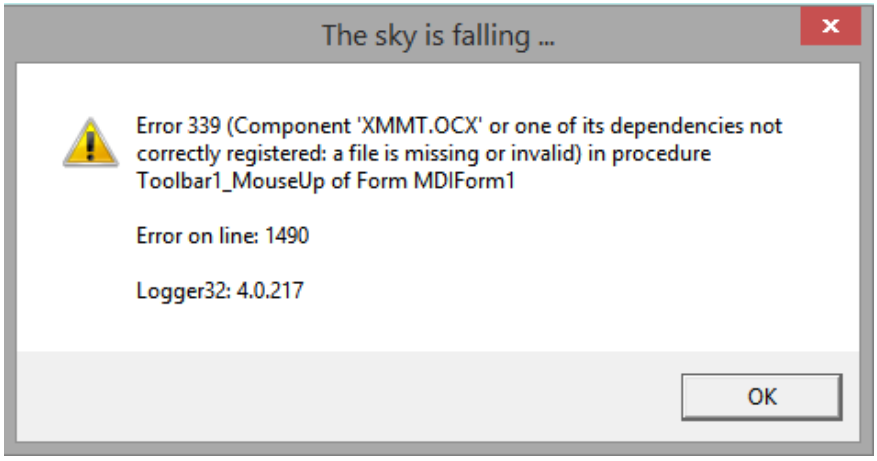
Copyright © 2026 Robert C. Furzer K4CY

Q. We recently installed Logger32 at our Scout camp station. It's working great. Can I copy the Band Mode table from my home instance of Logger32 to that one?

A. Yes. Simply copy the *C:\Logger32\myBandMode32.db* file from your home/shack PC to the Logger32 folder on the Scout camp's shack PC. If you are worried about messing something up, make a backup of the Scout PC first – its *C:\Logger32* folder at least.

> **Hinson tip**: it's easy to set up individual Logger32 configurations for various users and purposes (such as contesting), so if one of the Scouts *insists* on having vivid yellow text against a deep orange background, others need not suffer. Earn your Logger32 Scout badge by developing a simple batch file or script for users to launch Logger32 with the relevant parameter for their favorite configuration. For bonus marks and woggles, save backup copies of working configuration files somewhere safe for the time when *someone* inevitably decides that *everyone's* Logger32 config should have vivid yellow on orange …

Q. I just got an incomprehensible error message about a file or component not being correctly registered, or missing, or invalid, or something. Now what?



A. Logger32 is reporting the error 339 ▲ "The sky is falling …" is a K4CY classic.

The key clues in this example are '*not correctly registered*' (meaning there is a problem in the Windows registry) and '*.OCX*' (which is an ActiveX control - "an Object Linking and Embedding (OLE) custom control, a special-purpose program … providing such functions as handling scroll bar movement and window resizing" says TechTarget).

When Logger32 is installed or updated, the installer/auto-updater looks to see whether several necessary Windows functions and shared libraries are already installed and registered on your system. If so, it continues installing other files.

For some unknown reason, it appears the registry entry for XMMT.OCX in this case[56] or the file itself may have been corrupted or deleted. More likely there is an old version on your system that is incompatible with Logger32.

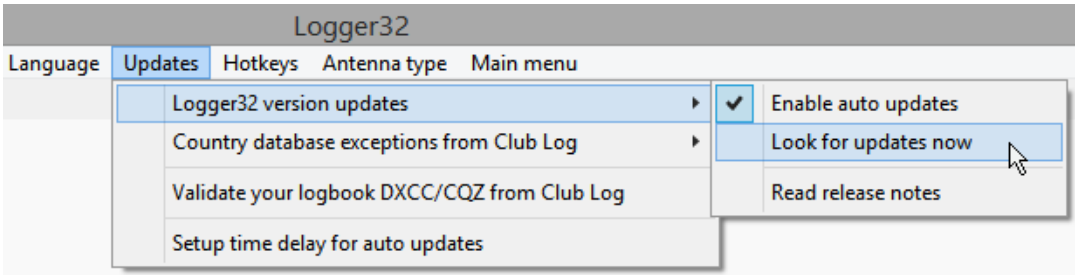
[56] Similar errors occur with some other shared libraries, such as *ComCtl32.ocx*. Presumably, they were installed with some other app ages ago, and remain on your disk today, unused, unloved and unupdated ('downdated'?). The resolution process is the same, so read on.

Copyright © 2026 Robert C. Furzer K4CY

To resolve this, choose the **easy**, **difficult** or **risky** route:

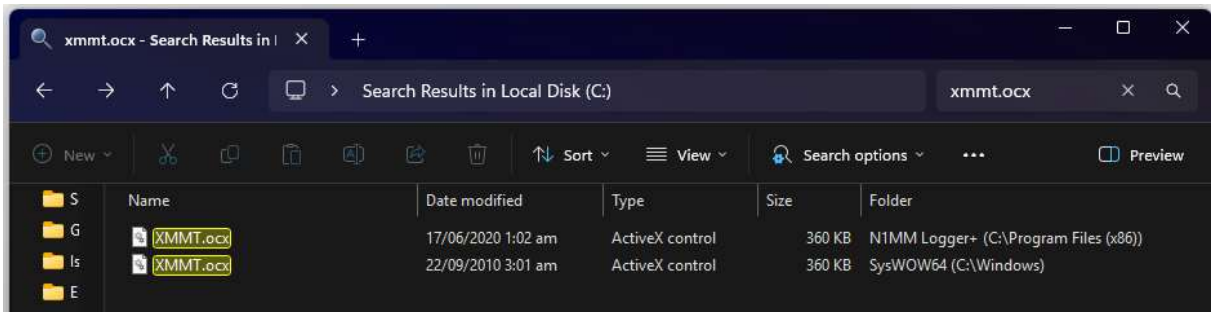
A) **Easy**:

- Export your entire log in ADIF format as a backup, just in case.
- Also backup your user files and logbook in Logger32's internal format using icons #1 and #2 on the far left of the toolbar.
- Do a full Logger32 v4.0 installation, using the download from www.logger32.net. The installer checks and if necessary puts in place and registers all the shared libraries *etc*. that Logger32 needs. Don't worry, it won't overwrite your log or configuration … and anyway even if it does, you have those backups, right?
- Start Logger32 and let it auto-update itself to the current version. You may need to use **Setup ⇨ Updates ⇨ Logger32 updates ⇨ Check for updates now** if you don't have it configured to check for auto-updates automatically as it starts up ▼

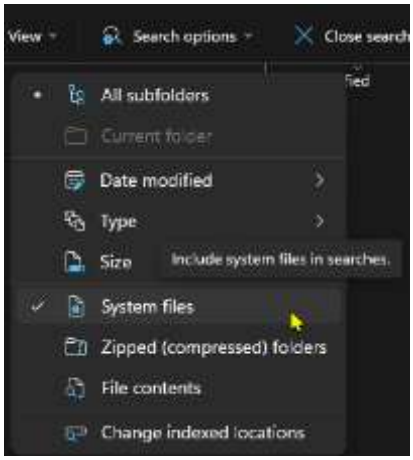


B) **Difficult**:

- Close Logger32 if it is still running.
- Press <**Win**+**E**> to open File **E**xplorer.
- Search for the problematic file by typing its name into the search box on the right (case insensitive), automatically opening the <**Search**> tab with its <**Search Tools**> ▼



- You may have to click the <**This PC**> button on the lower left to run the search, and wait for it to do its thing. The disk LED will probably glow red while it works, showing "Working on it…" on the screen.
- If nothing shows up in the results, ensure that <**All subfolders**> is selected and <**System files**> are ticked under <**Search options**> ▶



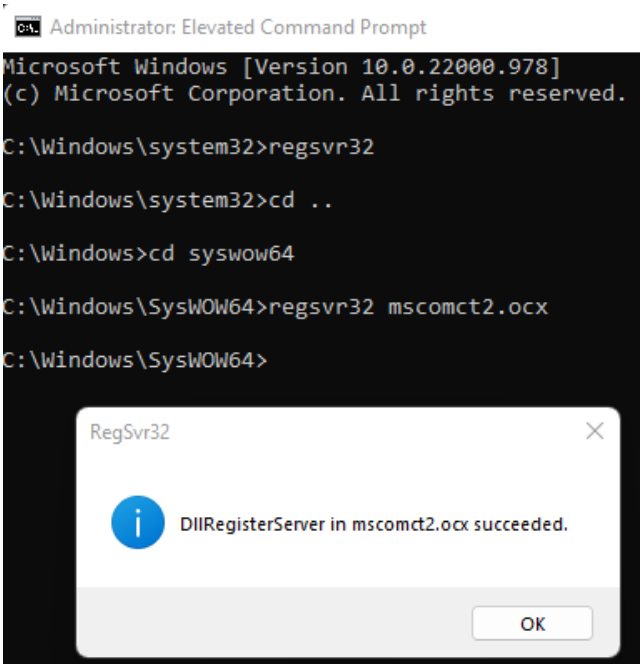
- If the problematic file is found, you should see it highlighted in yellow in the results area. In the example above, I found two copies of XMMT.ocx, one an ancient copy in the SysWOW64 folder and another from a more recent installation of N1MM+.

Copyright © 2026 Robert C. Furzer K4CY

- Delete the problematic file by clicking to select it, then clicking <**Delete**>. Repeat for any other copies of the same file[57].
- Now install Logger32 using the full installer[58]. This time around, it will find discover that the library file is missing, install it and register it with Windows.

C) **Risky**:

- If the following instructions make little sense to you, don't take the risk. Use another approach. You can easily screw up/brick your system this way.
- Make sure you have a recent backup of your C: system drive.
- Locate a current or reasonably recent copy of the file that Logger32 is moaning about – ideally on your PC somewhere, if not download it from a trustworthy source (such as Microsoft.com – OK, a *relatively* trustworthy source).
- Put the file in a suitable folder such as *C:\Windows\SysWOW64* on 64-bit Windows, if it is not already there.
- Manually register the file using regsvr32 from an elevated command prompt, having changed to the folder containing the file ▶
- If this didn't work, good luck restoring your C: drive from the backup, assuming it isn't totally bricked. You *were* warned.



Hi, I can confirm that running Logger32 with Windows 7 compatibility has solved my problem with sudden crashes. I haven't had a single crash since I set this compatibility.

Libor OK1ALX

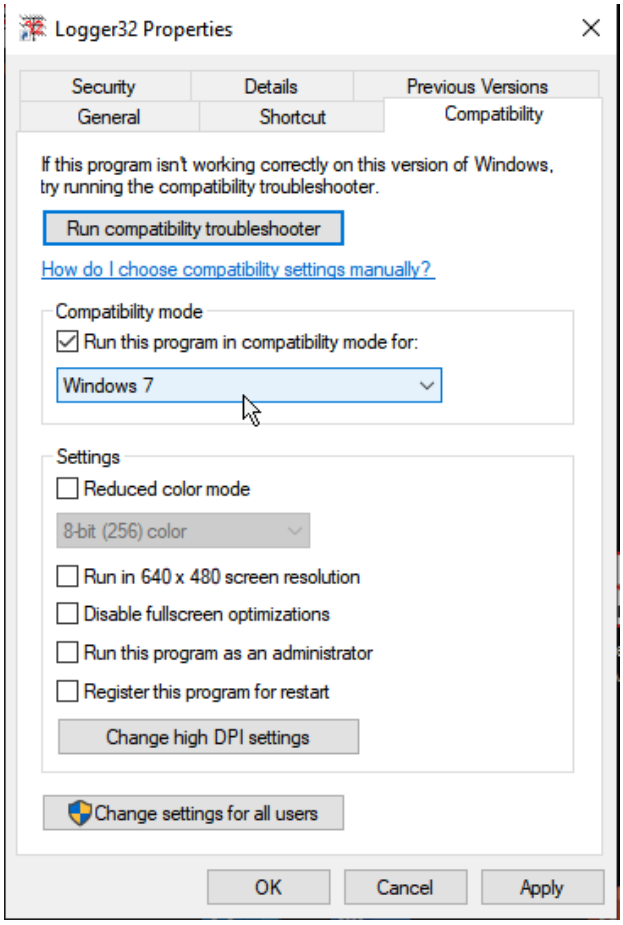
[57] Windows *might* complain that the file is in use and cannot be deleted, in which case you'll have to close down apps, or possibly reboot and try the search-and-delete again before running any apps.

[58] The installer replaces Logger32's files but leaves your log and *.INI* configuration files untouched. If you are wary of it accidentally damaging your valuable data for some reason, make a backup first, just in case.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why does Logger32 randomly crash?

A. I wish we knew the answer! Something is evidently broken in the way Logger32 interacts with the database middleware and Windows operating system in *some* Logger32 installations but, thus far, we have been unable to locate and eliminate the annoying, intermittent bug or bugs. However, aside from disabling DEP, we've found another workaround: running Logger32 in Windows 7 compatibility mode *seems* to prevent the crashes.



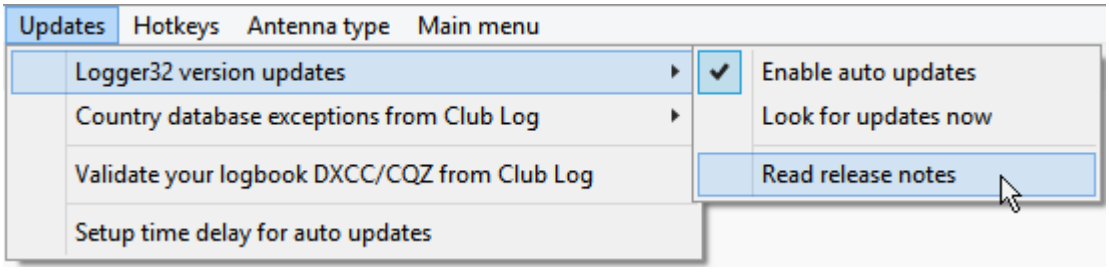
To do this:

1. Right-click the Logger32 desktop icon that you normally use to run Logger32.
2. Click **Properties** at the bottom of the menu.
3. Click to open the **Compatibility** tab.
4. Tick <**Run this program in compatibility mode for:**> and select Windows 7 from the options in the box ▶
5. Click <**OK**> to apply the change and close the properties form.

Hinson tip: paradoxically, 'random' or 'occasional' crashes cause far more grief than crashes that happen every single time certain conditions or sequences occur. You can help by taking notes about exactly what you had been doing when the crashes occur. Does Logger32 only crash when you are on 15m, or when you click a DX spot? Is it running one of the award reports or file export functions that makes it crash? What other applications are you running when the crashes occur? Once we know and can reproduce the trigger conditions on demand, we stand a *much* better chance of diagnosing and fixing the actual problem/s, then proving that it is truly fixed.

Q. Apparently, according to the release notes, I should have done something after an auto-update but I missed the instruction, so what should I do?

A. Read the release notes now using **Setup ⇨ Updates ⇨ Logger32 version updates ⇨ Read release notes** ▼ Do as instructed … and pay more attention in future!



Copyright © 2026 Robert C. Furzer K4CY

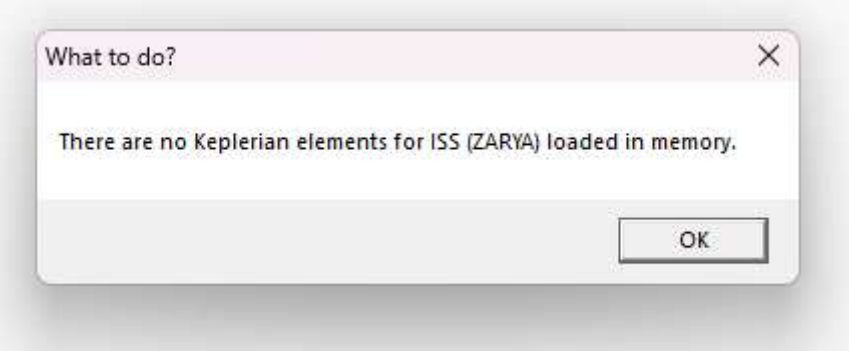
Q. Why is my bandplan wrong?

A. Logger32 provides a *generic* bandplan as an example, a starting point. **You need to edit the bandplan (*i.e.* the Bands & Modes table) to suit your individual needs and to reflect your region of the world.**

This is one of the practical issues when developing software for a global hobby with evolving standards. Logger32 cannot reasonably be expected to support all possible situations straight out of the box. You do, however, have *lots* of tools to customize the system.

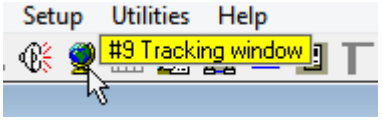
Hinson tip: see the earlier FAQs for advice on editing the Bands & Modes table. There's more to this than it appears.

Q. I have a message when opening the tracking window that says, "What to do? There are no Keplerian elements for [some obscure satellite, presumably] loaded in memory." I don't use satellites and I want to stop this annoying message from bugging me. Any suggestions?

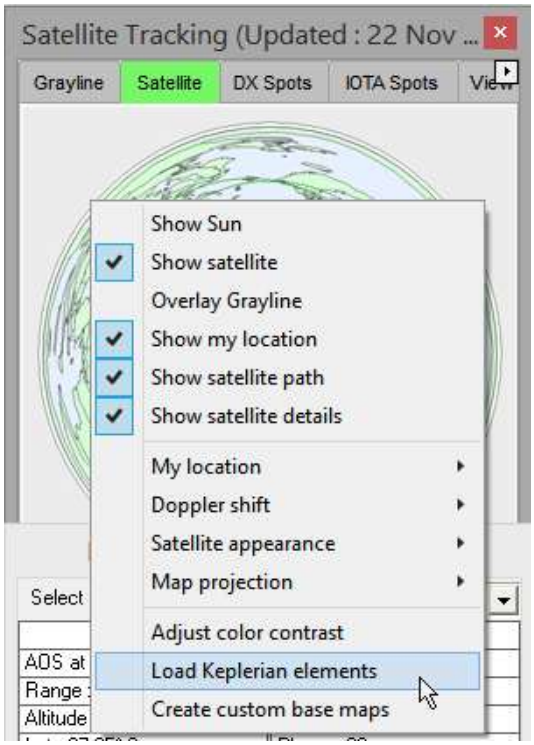


A. You can simply <**OK**> it and press on, but to get rid of the message, update your Keplerian elements:

1. Open the Tracking window using the globe icon #9, or **View ⇨ Show Tracking Window**.



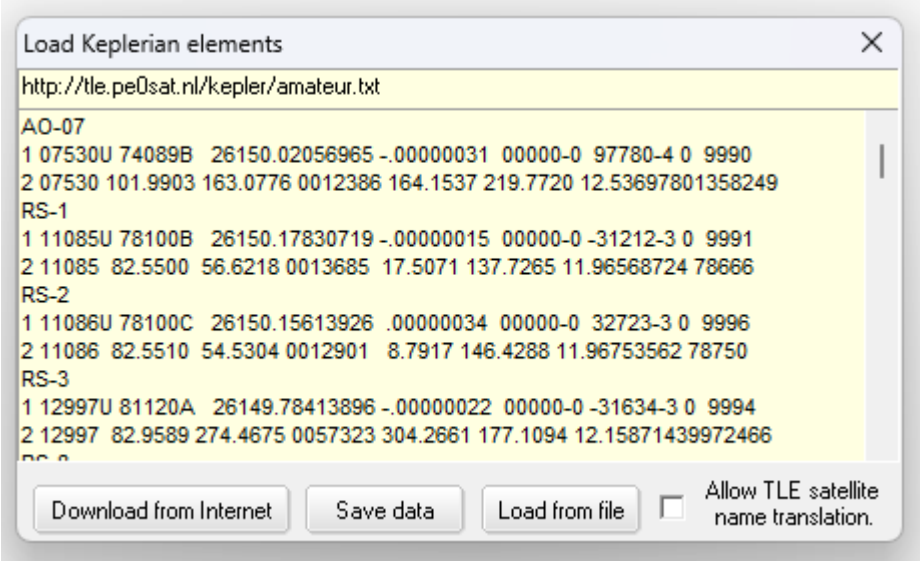
2. On the tracking window, click to open the <**Satellite**> tab. It will probably be blank at this stage with no map shown, since no orbital data have been loaded yet.
3. Right-click the pane then click <**Load Keplerian elements**> ▶



Copyright © 2026 Robert C. Furzer K4CY

4. At the top of the <**Load Keplerian elements**> form, check/adjust the Internet URL to download orbital data (Keplerian elements – also known as "**T**wo-**L**ine **E**lements"). Logger32 used to default to the Celestrak site but they changed TLE format, so find a different TLE source such as http://tle.pe0sat.nz/kepler/amateur.txt

5. Click <**Download from Internet**> ▼ to grab the current Keps



6. Click <**Save data**> to load the new Keps into Logger32.

Hopefully that's enough to stop Logger32 *whining* incessantly at every start up … and, as a bonus, you can now track the amateur satellites quite accurately. Hey, some day, you might even get interested in satcomms!

Hinson tip: Keplerian elements ("Keps") are not some obscure group of atoms clustered together on the periodic table, but a set of atomic (as in fundamental) geometric parameters defining the orbital paths of satellites, named after the famous astronomer and all round bright spark Johannes Kepler.

Q. Any time I load Logger 32, Microsoft Office 2007 wants to configure … why?

A. Office 2007?! By default, when Logger32 starts up, it looks for Excel on the PC. If Excel is installed, Logger32 enables various awards menu options to transfer award data from the screen to an Excel spreadsheet. This check is presumably waking up your installation of Office 2007 from its *long* hibernation.

Your options are to:

(a) Configure Office 2007 and get it running sweetly (good luck with that!);

(b) De-install Office 2007 completely (a good idea if you are not using it at all) or update it;

(c) Mess around with the registry settings so Windows no longer believes Office is installed on the PC (tricky, risky and definitely *not* recommended); or

(d) Simply tell Logger32 to skip its startup check for Excel:

1. Close Logger32.

Copyright © 2026 Robert C. Furzer K4CY

2. Press <**Win+E**> to open File **E**xplorer, then navigate to your Logger32 folder (usually *C:\Logger32*).
3. Open the file *Logger32.INI* in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad.
4. Browse down or search to find the section headed **[Globals]**.
5. Add a new row below the [**Globals**] heading with: **Check For Excel=No**
6. Save the edited file and exit the editor.
7. Open Logger32 to confirm that it no longer prods your ancient copy of Office into life.
8. Optionally, make the same change to any other Logger32 configurations that you use by editing their *.INI* files in the same way (*e.g.* you may be using something vaguely similar to *C:\Logger32\contests.INI* for casual contesting).

Hinson tip: you won't be able to analyse or print your DXCC and IOTA statistics in Excel from Logger32's award reports if you have told Logger32 not to check for Excel.

Q. What if I received some other error message during Logger32 installation?

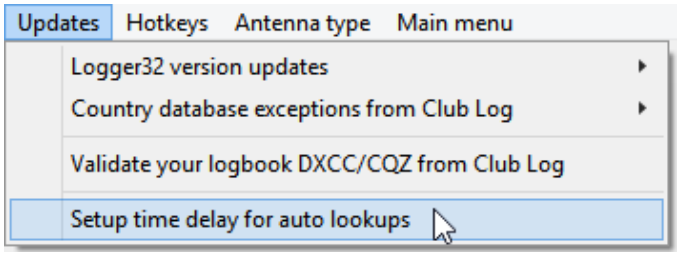
A. Reboot then try it again. This time, read and make a note of any error messages, or grab a screenshot of it if you can. Browse or search this manual for hints, including the troubleshooting section.

If you're stuck, describe your problem and seek help on the Logger32 reflector. Being crystal clear about the error message (ideally, include a screen-grab of the message, or a reasonably clear photo of the screen taken on your smartphone) and what you were doing at the time the error occurred is big help: we are not mystics. Our ESP powers are weak, our crystal balls cloudy and misshapen. After you OK to close the error message, try doing the same thing again to check if the error is repeatable: that will help us pin down and hopefully fix the issue. Poorly described, intermittent or irreproducible errors are the bane of our lives, worse even than those who don't follow the instructions and guidance in this manual …

Q. After starting my shack PC in the morning, I'd like to get busy checking DX spots … but the spots don't start flowing until Logger32 has finished checking for updates. What can I do to speed that up?

A. Well you *could* disable the update checks that happen automatically whenever Logger32 starts (by un-ticking **Setup ⇨ Updates ⇨ Logger32 version updates ⇨ Enable auto updates**), instead checking manually for updates if/when you remember … but in case you forget to check for update manually, a safer option is simply to *delay* the automatic update checks to give your system a chance to grab some DX spots first.

Look under **Setup ⇨ Updates** ▶



Waiting, say, 60 seconds should be more than sufficient for Logger32 to connect and log in to the DX cluster and send a SHOW/DX 100 (or whatever) command as part of your login script, populating its DX

Copyright © 2026 Robert C. Furzer K4CY

spots window and the BandMaps with plenty of juicy DX spots for you to peruse while it checks for updates.

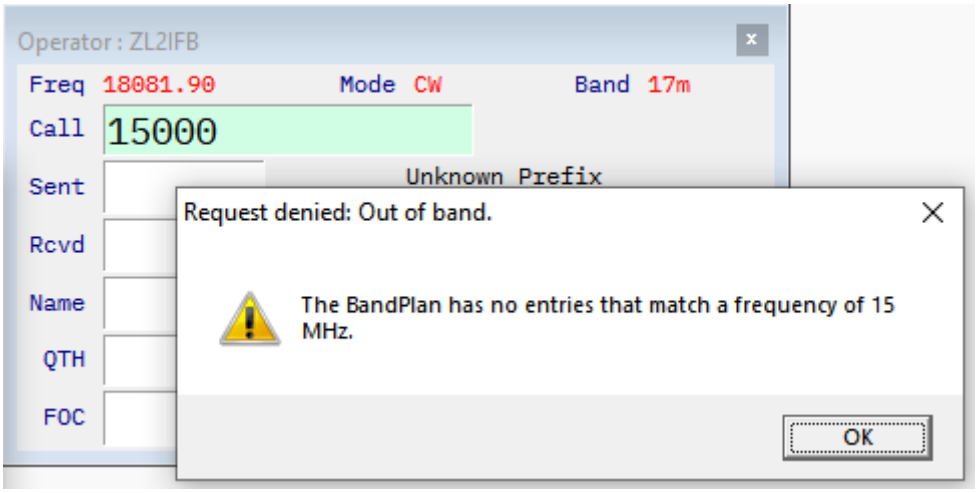
Q. What should I do here? The auto-updater failed with an error message about being unable to delete files from the *C:\Logger32\Rollback* folder.

A. The simplest solution is to delete the *C:\Logger32\Rollback* folder (delete all the files in it, and the folder itself) through File **E**xplorer (opened with <**Win**+**E**>). The auto-updater should now run, automatically recreating the rollback folder and moving into it the original files that it replaces during the update.

In the unlikely event that you need to rollback after this fix, having just deleted any earlier backups, you will only be able to go backwards just one release but that's probably sufficient. If not, try looking in your browser's default download folder for other Logger32 downloads.

> **Hinson tip**: if you are reluctant to delete the older rollback files and folder, you could rename the folder or move it to a backup location, such as a USB memory stick. Personally, I wouldn't bother … but then I do have numerous backups of my PC to recover stuff when needed.

Q. Logger32 won't let me QSY my radio to one of the WWV frequencies because it is 'out of band' ▼ So how do I calibrate my VFO or check the time?



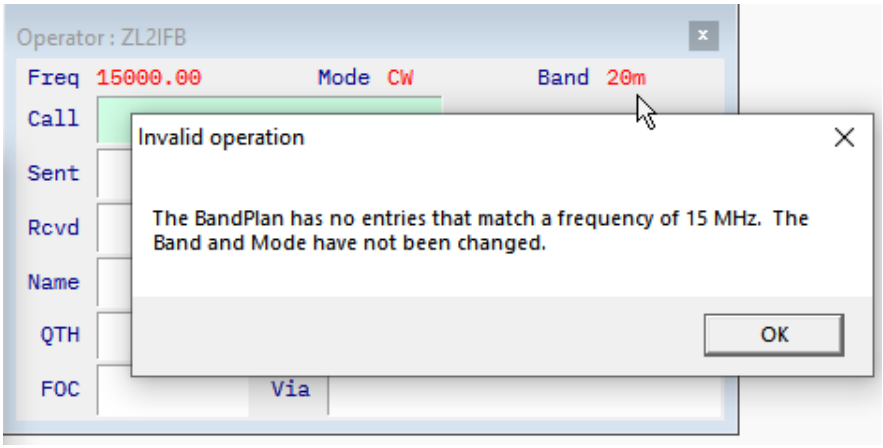
A. The crude but easy solution is simply to QSY directly to the required frequency using the knobs and buttons on the radio, rather than through Logger32. For example, on my K3, I press the <FREQ ENT> key, followed by 1 then 5 on the radio keypad, and finally press the <AFX ↵> key to go to 15.00000000 MHz.

That technique works just fine, even when Logger32 or the shack PC isn't running!

If Logger32 is running and is connected to the radio via CAT when you QSY the radio directly to a frequency not listed in its Bands & Modes table, it may grumble …

Copyright © 2026 Robert C. Furzer K4CY

If you have enabled out-of-band messages through the <**View**> menu, Logger32 notes the frequency change and duly warns you ▶



A more elegant solution is to add a band covering the entire frequency range of your radio (at least) using **Tools ⇨ Edit Bands & Modes**. That way, no valid VFO frequency is 'out of band', so you can enter any frequency into the Call field of the log entry pane and hit <**Enter**> to QSY the active radio.

Either call your new band "gen" (for **gen**eral coverage), or add an entry to *C:\Logger32\ADIFBands.txt*. Remember, the LF edges of every band or band segment *must* be listed in descending order, so add the band to the very bottom of the table ▼

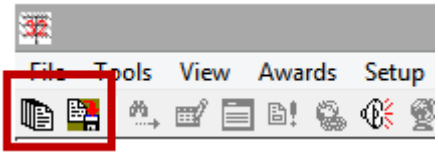


Hinson tip: having changed your Bands & Modes table, you may need to close and restart Logger32 to pick up the changes.

Hinson tip: to calibrate your VFO, let your radio warm up, then tune to a calibrated signal generator or standard frequency transmission at the highest frequency you can copy (*e.g.* WWV on *exactly* 25 MHz). Adjust your radio's reference oscillator so that the audio tone sounds the same as you switch back and forth between upper and lower sideband. For greater accuracy, use an audio program such as Audacity to confirm that. If you measure the audio frequency at the peak, it should match your radio's CW sidetone/pitch setting (500 Hz in my case). Alternatively, follow the radio manufacturer's calibration instructions.

Copyright © 2026 Robert C. Furzer K4CY

Q. I messed up and now Logger32 is *so* broken that I have lost all hope of ever fixing my configuration. I need to start over … but can I keep my preferences, macros and so forth without having to re-configure *everything*?



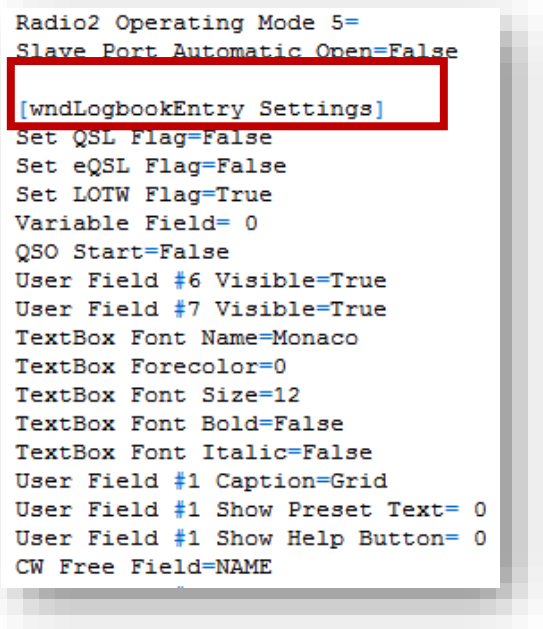
A. Provided Logger32 will still open and run, back up your user files and logbook using icons #1 and #2 on the far left of the toolbar ▶

Most of your settings and preferences[59] are kept in the configuration file *C:\Logger32\Logger32.INI*. Move this file to another folder or rename it (*e.g. MY-SETTINGS.INI*). Reinstall Logger32 by downloading and running the full v4 installer: in the absence of an existing *Logger32.INI*, Logger32 will create a new, basic *Logger32.INI* file with all the default settings.

Once you get it going you can then copy your old *Logger32.INI* back into the *C:\Logger32* folder to restore your original settings – either the whole thing (which is quite likely to land you back in the same sorry mess that started this quest) or - in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad - just copy across key, difficult to reconfigure parts one at a time from the old to the new *Logger32.INI* file until you're close enough to perfection to get on with using the program normally.

Hinson tip: each section in *Logger32.INI* starts with its name in square brackets, a name relating to its function. If you insert a blank line before each one, it is easier to navigate and edit the file ▶ Don't over-do it though: those are the only blank lines allowed.

Hinson tip: the parameter names in *Logger32.INI* relate to their functions, but some are a little cryptic and the parameter values are often incomprehensible to us mere humans: they are intended to be read and interpreted by the program. Avoid meddling with parameter values in the text editor: it is much safer and easier to change them through the configuration menus and functions within Logger32. If you feel *sure* there is 'something badly wrong' with a given parameter, it's OK to delete it so Logger32 will revert to its default setting, then you can reconfigure it through those setup screens.



[59] Macros are kept in their respective *SoundCardMacros.ini* files. Other *.INI* and *.ini* files contain other user configs. These will *probably* be OK … but back them up anyway if "*probably*" is unsettling.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: save a backup copy of your *C:\Logger32\Logger32.INI* file *before* you start editing it, just in case you make a hash of it. Work systematically: change one section, save the edited *.INI* file, then start Logger32 to check out those changes before closing Logger32 to work on the next section, and so on. Save additional backup copies during the process if you have a lot of changes to make, so you can easily revert to the last version that worked OK … and when you are all done (maybe in a week or three after you have had the opportunity to use Logger32 for real, as you normally do) save yet another backup of *Logger32.INI* with a name such as *Logger32 WORKS OK Q1 2026.INI* Trust me, as a Logger32 user for about two decades, having invested a lot of time and effort in customizing Logger32 to suit your particular habits and preferences, starting over with the default installation can be a cathartic experience.

Hinson tip: you can start Logger32 running with your choice of *.INI* files by specifying one on the command line. These are known as 'configurations'. Aside from your normal everyday operating configuration (normally defined in *Logger32.INI*) you might want additional configurations specifically for contesting, for special events, for testing and faultfinding on Logger32, for demonstrating the software to anyone interested, to try out new program functions safely *etc*.

Hinson tip: the same points apply to the other *.INI* files in *C:\Logger32*. They save the configuration options, macros *etc*. If something drastic goes wrong (such as a disk failure or major cock-up while you are fiddling with the program's settings), it's a relief to be able to retrieve a known good version from a fairly recent backup rather than have to start from scratch.

Hinson tip: that reminds me, be sure to save at least some backups 'offline', for example on a USB stick or CD-RW – particularly your log/s but also those *.INI* files. I've suffered more than enough disk failures in my life to appreciate the value of this, and ransomware incidents are a genuine threat today. Why take the risk when the solution is so simple? Love your backups!

Bonus tip for attentive readers: even if you foolishly neglect to make backups, Logger32's auto-update function saves a copy of your *.INI* files for you. Hunt through your *C:\Logger32\Updatefiles for Ver 4.0.nnn* folders for one that suits your purposes … assuming you have been running the auto-updates anyway. If not, well you've nobody but yourself to blame, having read these tips.

Q. My awards totals look wrong. I vaguely recall I should have recalculated my statistics after a recent update but I forgot. Now what?

A. It's never too late to recalculate statistics. Logger32 will erase its [dubious] statistical data, then work its way diligently through your log, laboriously rebuilding the stats from scratch, one QSO at a time. You'll then have to close and re-open the awards table to pick up any changes as a result.

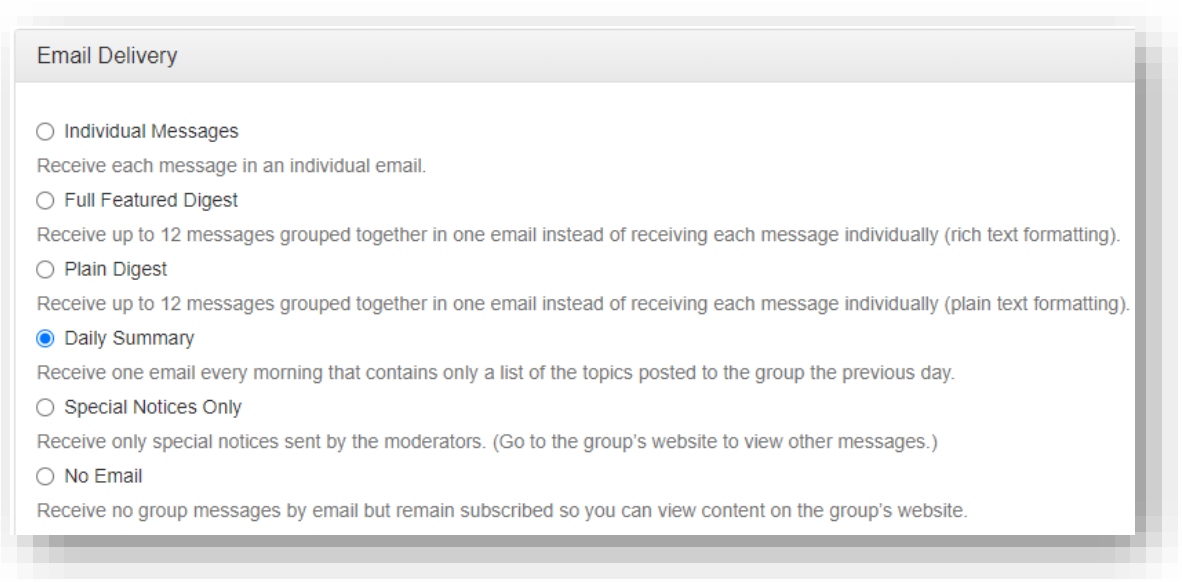
Copyright © 2026 Robert C. Furzer K4CY

Q. The Logger32 reflector is so busy that I can't keep up. Any tips?

A. Yes. The Logger32 reflector *can* be overwhelming, sometimes, with the occasional flurry of messages.

It may help to have reflector messages delivered in batches instead of individual emails:

- Browse to the Groups.IO form to set the Hamlogger[60] delivery options.
- Make your choice and select *e.g.* the daily summary or one of the digest formats ▼



Hinson tip: decent email software lets us set up rules to label, tag or file reflector messages into a "Logger32" folder, ready to open and browse when we have the time and inclination. Since all the reflector messages contain [hamlogger] in the subject line, a label/tag/filing rule based on that string *should* work nicely. Good luck.

[60] The reflector was conceived and named before Hamlogger morphed into Logger32. The name stuck.

Copyright © 2026 Robert C. Furzer K4CY

3 Importing and exporting logs and other data files

> "There are lies, damn lies, and then there are data files"
>
> *ChatGPT*

Logger32 accepts ADIF files[61] that comply with the ADIF standard, identifying any gross problems[62] at import time since, unfortunately, some other, inferior programs do not follow or enforce the ADIF standard as precisely. You *may* need to edit or pre-process noncompliant or corrupted ADIF files before importing them into Logger32 to get around the problems.

Hinson tip: Logger32's import routine can load contest logs, optionally tagging all the imported QSOs with the specified contest name. Using **File ⇨ Import logs ⇨ ADIF (.adi)** file, simply tick <**Import Contest Log**> and you will be prompted to enter the contest name. Easy.

3.1 Importing ADIF logs

When importing ADIF files into your precious log, take care, step-by-step! It is all too easy to fill your log with junk – duplicate QSOs, QSOs made by other people, QSOs with missing or erroneous values, ADIF-noncompliant data or whatever. **Back up your log *before* importing.** If it all goes horribly wrong, you can easily open a new log and import the backup to put things back to how they were before. Before importing a large ADIF file, separate out the header and the first few QSOs into a small ADIF file, import that, and check carefully that it went OK before proceeding with the remainder. Or not. *Maybe* the de-duper function will save your bacon if you imported the same file twice.

Import a log saved in ADIF format into the currently-open log in Logger32 using **File ⇨ Import logs ⇨ ADIF (.adi) file ▼**



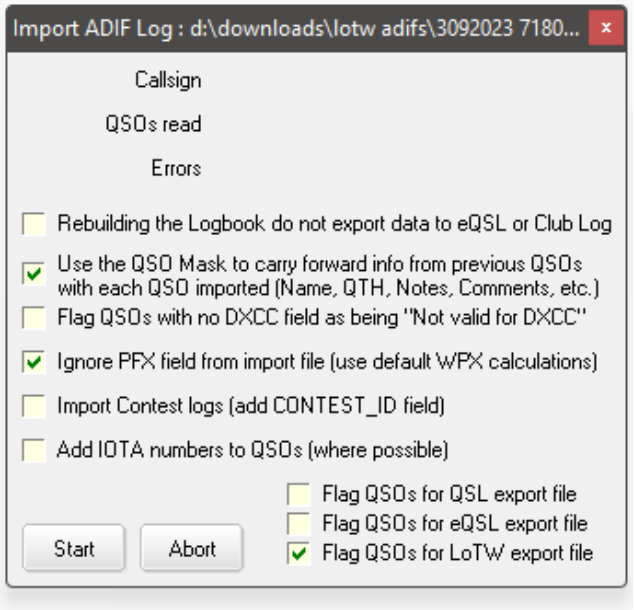
Logger32 asks which file you wish to import. Navigate to the relevant folder, click to select the ADIF file you wish to import and click <**Open**> to open a form with a few options …

[61] For instructions on importing logs from non-ADIF-compliant logging programs (such as Logger16), see Importing ancient logs from *prehistoric* systems.

[62] Logger32 cannot realistically be expected to identify and trap all possible input errors, particularly data errors such as little typos (*e.g.* a QTH of "Tokeo" instead of "Tokyo") and wrong zones where a station's DXCC entity spans multiple zones (*e.g.* Canada and USA).

Copyright © 2026 Robert C. Furzer K4CY

3.1.1 ADIF import options



- **Rebuilding the Logbook do not export data to eQSL or Club Log:** tick this if you are importing a log file containing QSOs that you know have already been uploaded to eQSL or Club Log. This will avoid uploading large numbers of duplicate QSOs. This option is independent of the <**Flag QSOs for QSL|eQSL|LoTW**> options. However, if you are rebuilding a logbook, the ADIF log file *presumably* has all the appropriate flags set already.
- **Use the QSO Mask to carry forward info from previous QSOs …** works in the same way as when you are logging QSOs manually: details such as the person's name and QTH can be carried forward from previously-logged to newly-imported QSOs.
- **Flag QSOs with no DXCC field as being "Not valid for DXCC": take care with this option!** Contest loggers typically omit the DXCC entity field so, if you tick this option, your imported QSOs will have the DXCC entity code 000 meaning <**Not accepted for DXCC**>. Logger32 also adds ** before each imported callsign in the log (for example **K4CY) as an indicator that the QSO is not counted in your DXCC award statistics.
- **Ignore PFX field from import file (use default WPX calculations):** Logger32 ignores any ADIF PFX data in the imported log, generating its own prefix values instead using the CQ WPX rules as specified in the ADIF standard. This option is intended to correct ADIF files that incorrectly populate the PFX field with, say, the prefix from the official DXCC list rather than the CQ WPX prefix. It is recommended to use this option always.
- **Import Contest Logs (add CONTEST_ID field):** if ticked, Logger32 prompts you for the name of the contest to use as the ADIF CONTEST_ID field for all the imported QSOs. Don't use this if you are importing a log with several contests and non-contest QSOs, as they will all be labelled with the same contest ID: either split the ADIF into separate logs or leave the CONTEST_ID field blank, unless you are happy to work your way through the log, editing numerous QSOs.
- **Add IOTA numbers to QSOs (where possible):** Logger32 can determine IOTA references by looking up the DXCC entity in the IOTA database (for situations where an entire entity has one IOTA reference) or from previous QSOs with the same station already in the logbook.
- **Flag QSOs for QSL export file**: sets the 'Send QSL' flag for *all* the imported QSOs. **Don't do this unless you truly want to prepare and send QSL cards for *every single* imported QSO**, despite a substantial proportion of hams no longer collecting QSL cards. Trust me, it is tedious to *un*set (reset) this flag QSO-by-QSO if you change your mind after importing and flagging a big log.
- **Flag QSOs for eQSL export file**: sets the 'Send eQSL' flag for all the imported QSOs. The next time you generate an eQSL export file, they will all be included, along with any others you have flagged.
- **Flag QSOs for LoTW export file**: sets the 'Send LoTW' flag for all the imported QSOs, marking them to be exported the next time you generate an LoTW export file, along with any others you have flagged. If you flag too many, no bother: LoTW ignores uploaded dupes.

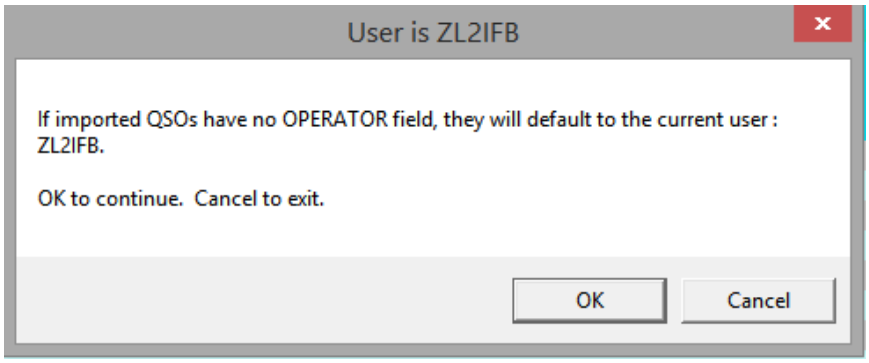
Copyright © 2026 Robert C. Furzer K4CY

Click <**Start**> to set the import going, and then <**Abort**> if for some reason you need to abandon the import (*e.g.* second thoughts about 100% QSLing …). Any QSOs that were imported *before* you hit <**Abort**> will remain in the logbook, as flagged … so it is better to review the import options *before* you start the process!

3.1.2 Import warnings and error messages

Prior to importing an ADIF file, Logger32 makes a just-in-case automatic backup first. If you are at all concerned about damaging/messing up your log (as you should be), take this opportunity to export your log as an ADIF file and stash it somewhere safe.

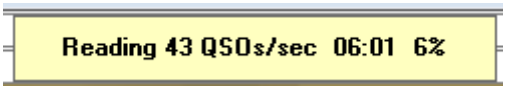
While importing your logs, Logger32 saves the content from the OPERATOR field in the *.ADI* file (if any) in the operator column of the logbook for the respective QSOs. If the OPERATOR field is missing from the ADIF file, Logger32 uses the current station operator instead. Before the import starts, Logger32 reminds you of this and shows the operator callsign that will be used if there is none in the import file ▶



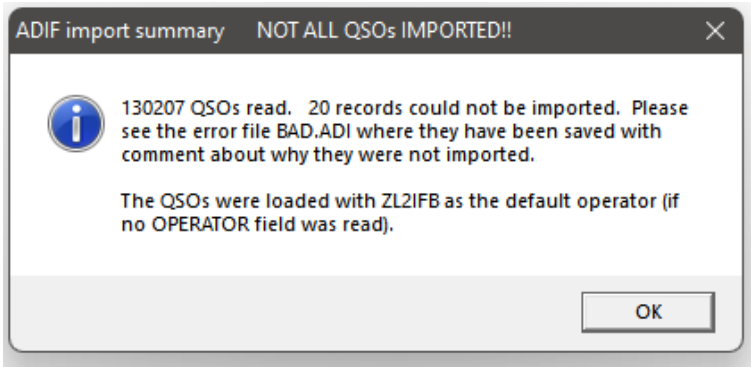
***Before* importing your ADIF log, be sure the log entry pane is displaying the correct operator callsign.**

If it isn't, use **File ⇨ Change operator** to correct it.

A message box in the bottom right corner of the screen shows the import rate[63], estimated time remaining and percentage complete ▶



After the file import completes, a message tells you ▶



- How many records (QSOs) were found in the imported ADIF file.
- How many data errors resulted in records not being loaded (imported) … and what to do about them *i.e.* open *C:\Logger32\BAD.ADI* with MS Notepad or WordPad, TED Notepad or ADIFmaster to check and maybe correct the errors.
- The QSOs were loaded into the log with the station operator (most likely your callsign) unless the OPERATOR field was populated int the ADIF file. More below.

This example shows how errors are typically reported in *BAD.ADI*:

[63] Importing is briefly interrupted whenever new DX spots arrive, so you can speed up the process a little by disconnecting from DX cluster while importing a large file. The QSO processing rate depends on your PC's raw CPU power and disk read/write speed – more info here.

Copyright © 2026 Robert C. Furzer K4CY

Error in the <QSO_DATE:x> field: <CALL:5>CA1LL <QSO_DATE:8:D>04052002 <TIME_ON:6>230814 <MODE:3>SSB <EOR>

The ADIF standard specifies that dates *must* be in the format YYYYMMDD but, as you can see, this record does not comply. There was no radio in the year 0405, amateur or professional!

3.1.3 Operators

OPERATOR is an ADIF field for the **personal callsign of the operator** - the person who made each QSO … but, unfortunately, the field is optional. Inferior logging apps may not use or export the OPERATOR field, or the data may be noncompliant (*e.g.* containing the operators' *names* rather than their *callsigns*).

You can use this ADIF field in your logbook to keep track of the different callsigns you hold now (*e.g.* regular and contest calls, portable and mobile calls …) or may once have held (*e.g.* a novice call), even if your logbook contains a mixture of QSOs with different operators. Logger32 lets you display QSO data and statistics either for individual operators, or for all the operators in the logbook.

In a radio club, for instance, the club station and callsign may be aired by several members: if they each 'sign on' to the club's Logger32 system using their personal callsigns as "operator", their QSOs and statistics can be distinguished. They can even export *just* the QSOs they have made from the club's log, filtered by operator, in order to update their home logs with those QSOs.

Hinson tip: if you use the L32 LogSync utility to upload your QSOs automatically to Club Log, it will dutifully upload the imported QSOs to your personal account and log at Club Log – which may be fine *if* you made those QSOs at the club, but not otherwise. Before importing the club's multi-operator logs into Logger32, remember to tick the first option on the ADIF import form <**Rebuilding the Logbook do not export data to eQSL or Club Log**>.

Hinson tip: as noted above, if you import an ADIF log that lacks the optional OPERATOR field, Logger32 records the current operator in the OPERATOR field of the imported QSOs. So, if you made the imported QSOs using a different callsign (such as your club, contest or special event callsign), change the operator to that callsign using **File ⇨ Change operator** *before* importing the log, then afterwards change it to whatever callsign you are using over the air. If it is too late (you rashly ignored the warning and messed up your log), be grateful for your pre-import log backup.

3.1.4 Importing logs from *prehistoric* systems

Logs can generally be imported into Logger32 quite easily *if* they are available in ADIF format, even an old ADIF version. The ADIF standard has been updated a few times, usually by adding new ADIF fields or adjusting the lists of recognized field values, hence it is generally backward-compatible. For instance, logs generated a few years back pre-date FT8. It didn't matter then that there was no standard way to represent the mode in ADIF … since the mode hadn't even been invented yet.

Your main challenge is likely to be how to generate ADIF log files from old IT systems, particularly *ancient* logging software that pre-dates ADIF, or more recent software that didn't (and maybe still doesn't) properly support ADIF.

You may have available, or be able to generate, some other file format that can be translated into ADIF by pre-processing: for example, a log in **C**omma **S**eparated **V**ariable *.csv* format can probably be imported into Excel, generating an *.xls* Excel data file which can then be converted to ADIF using one of the handy Excel-to-ADIF utilities out there (Google them!).

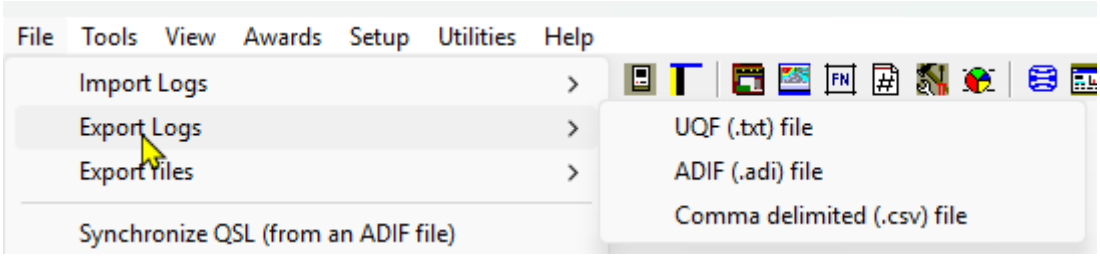
Copyright © 2026 Robert C. Furzer K4CY

Another complication is that misinterpretation of the ADIF standard has led to minor non-compliance in some logging software and/or in how it is used *e.g.:*

- Some ADIF files erroneously contain DXCC information in the PFX field (*e.g.* "K" for a QSO with WD4ABC). According to the ADIF standard, the PFX field should in fact contain the WPX prefix, not the country prefix taken from the current ARRL DXCC list. Thankfully, the ADIF import routine in Logger32 can ignore the PFX field in the input file, automatically determining and saving the *correct* ADIF-compliant PFX data from the callsign in each QSO record.
- The introduction of 'submodes' into ADIF when the standard was updated in 2019 caused confusion, partly due to inconsistencies in how the modes and submodes were defined. ADIFs created around that time may contain the pre-, para- or post-update mode definitions that some of us set up as workarounds until this was resolved (*e.g.* we defined the mode "DATA").

3.2 Exporting logs

Logger32 can export your logs in three different formats using **File ⇨ Export logs ▼**



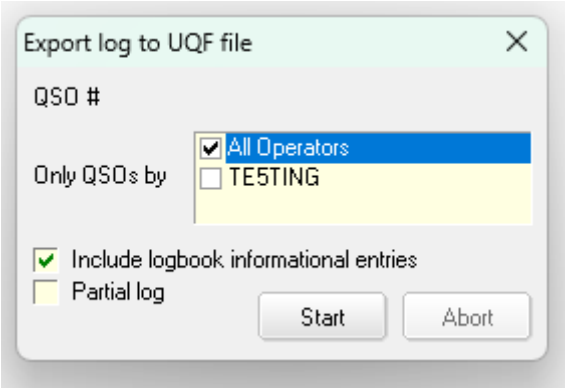
Choose one of the three output formats on offer. If in doubt, choose ADIF. The next *four* sections cover those *three* formats …

3.2.1 UQF (*.txt*) log exports

```
K5DX 16 Jun 2021 18:38 17m CW 599
EZ1DX 16 Jun 2021 18:38 17m CW 599
P5DX 16 Jun 2021 18:38 160m SSTV 599
K4CY 21 Jan 2026 23:00 15m FT4 229
JA1NLX 21 Jan 2026 23:01 15m CW 599
SV1VS 21 Jan 2026 23:02 12m SSB 59
N8DC 21 Jan 2026 23:02 6m CW 447A
```

◀ **U**niversal **Q**SO **F**ormat is a curious plain ASCII space-delineated text file format from the early 80's with only a few QSO data fields, and in non-standard formats at that.

There are just 3 UQF export parameters ▶



1. **Only QSOs by:** from the list presented, choose the operator(s) whose QSO records should be exported, or tick "All Operators".
2. **Include logbook informational entries:** as well as ordinary QSOs, the UQF export may include or exclude logbook entries containing propagation information, SWL reports *etc*.
3. **Partial log**: see below.

When you are ready, click <**Start**>. Logger32 asks you for the folder and name of the export file: supply the information then click <**Open**> … and respond to the next inane prompt about whether to create the file or, if a file with the name already exists, whether to replace or append to it, or

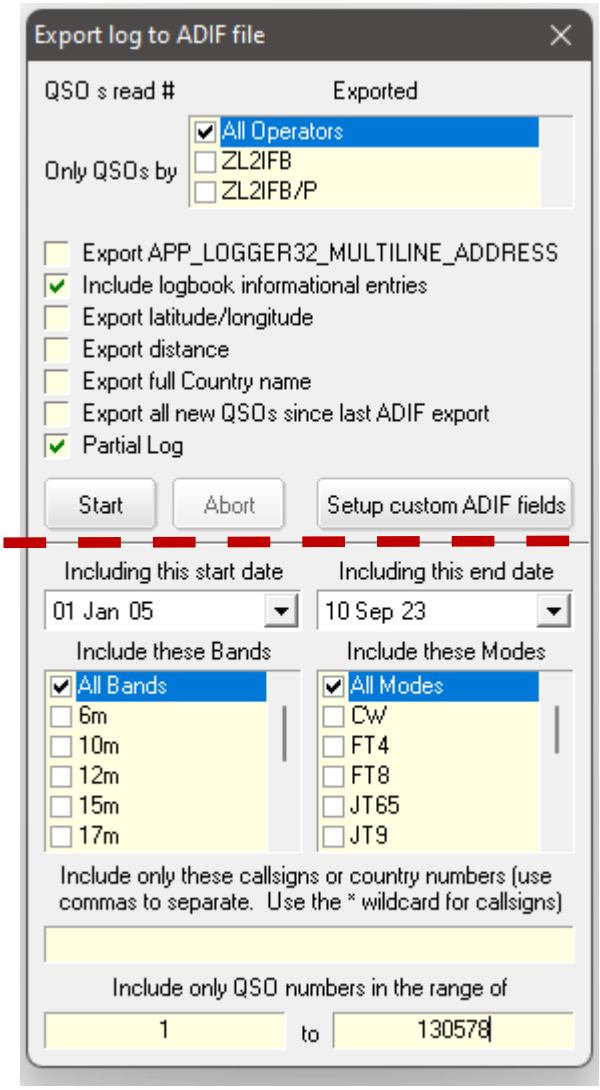
Copyright © 2026 Robert C. Furzer K4CY

cancel the export and become a monk. If you have second thoughts while it is exporting, click the <**Abort**> button and start again.

3.2.2 Partial log exports

The <**Partial log**> export capability applies in the same way to all three types of log export, so we'll explain it just once. If you don't want to export your *entire* log, <**Partial log**> let you specify QSOs ▶

1. Within a range of UTC dates (*including* both the start and end days).
2. Logged on specific bands and/or modes.
3. With specific callsigns or DXCC entities[64].
4. Starting and ending QSO numbers (inclusive of both, again)[65].



The lower half of the form with the selection criteria magically appears *after* you tick <**Partial Log**> above ▶

If you haven't ticked <**Partial Log**> yet, you see a short form - just the part above the red dashes.

Depending which file format you are exporting, you may only have some options here for a partial, partial extract.

Hinson tip: unless you are exporting your log as an ADIF backup that you may need to import into Logger32 later, *don't* tick <**Include logbook informational entries**>. Informational entries or pseudo-QSOs are unlikely to be interpreted correctly if imported by other logging software: they may be skipped, flagged as errors, perhaps even logged as genuine QSOs. Logger32 imports and understands them correctly, of course. It's clever like that.

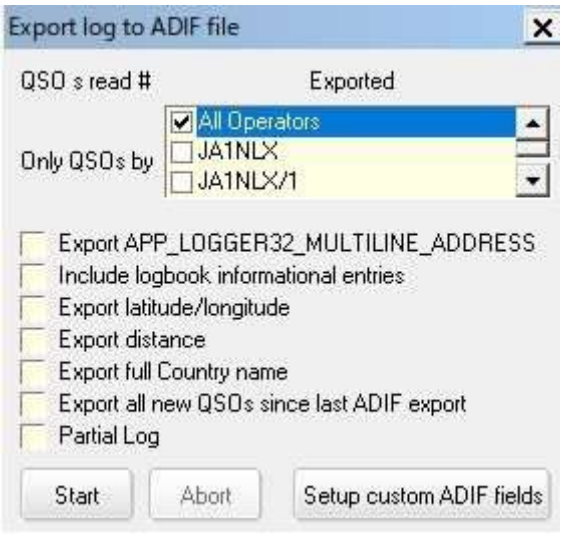
[64] You need to list their DXCC entity numbers, separating each number with a comma … which means looking up the Official Numbered List of DXCC Entities, also known as the 'ARRL DXCC LIST'.

[65] Run <**Reformat QSO numbers**> before using this option if you think QSOs might have been edited or added (*e.g.* imported) to your log out of sequence, and if you want to be sure of exporting all QSOs in a specific range. Or don't. Live a little. Take a chance. See what happens.

Copyright © 2026 Robert C. Furzer K4CY

3.2.3 ADIF log exports

Logger32 follows the current ADIF standard when exporting ADIF files. After selecting the ADIF format you have several options ▶

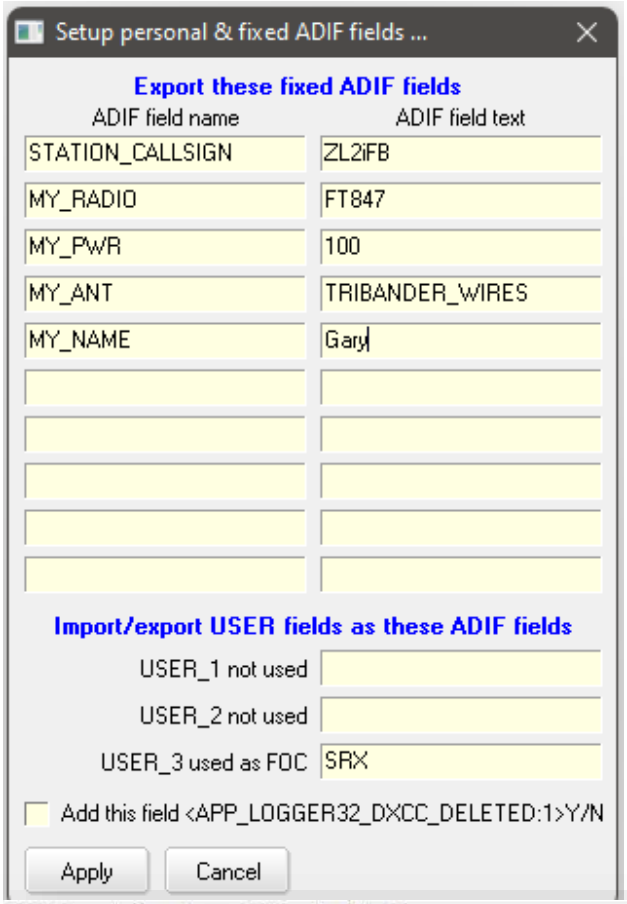


1. **Only QSOs by**: from the list, choose the operator(s) whose QSO records should be exported, or "All Operators" for anyone's.
2. **Export APP_LOGGER32_MULTILINE_ADDRESS:** adds a custom version of the ADIF ADDRESS field, replacing commas in the address with newlines.
3. **Include logbook informational entries:** as well as ordinary QSOs, the ADIF export can include logbook entries containing propagation information, dubious/incomplete QSOs, SWL reports *etc*.
4. **Export latitude/longitude:** the calculated latitude and longitude for each DX station logged are exported into the <APP_LOGGER32_LAT:x> and <APP_LOGGER32_LNG:x> fields. The values are degrees and decimal degrees, with **negative** values for **South** of the equator and **East** of the Greenwich meridian. For example, a JA station positioned at 35.10 degrees latitude and -136.86 longitude is near Nagoya City, according to the search function in Google Maps.
5. **Export distance:** the file includes the calculated great circle short path distances from our station location to DX stations whose locations are known, at least roughly anyway. Distances to maritime or aeronautical mobiles are incalculable without their grid squares.
6. **Export full Country name:** the DX station's DXCC entity name (from the current ARRL DXCC list) will be included in each ADIF record exported *e.g.* "Wales" for QSOs with Welsh stations.
7. **Export all new QSOs since last ADIF export**: Logger32 remembers when you last ran an ADIF export. It can export only more recent QSOs with this setting, perhaps to update your QSL manager with recent QSOs or upload them to an online log such as LoTW.
8. **Partial Log** is similar, but you get to choose the date range for the export, or other selection criteria. See the previous section for more on this, if you need it.

Hinson tip: a log 'archive' is basically a backup that is designed to last indefinitely – for years or decades. It enables you to recover your log despite serious incidents such as shack fires or floods, computer thefts, lightning strikes, hard disk failures, hacks, malware infections, 'operator error' *etc*. ADIF is a good format for archival, since it is portable and reasonably future-proof. When exporting, for a complete record, be sure to archive *all* your QSOs (all callsigns, all time, all modes, all bands, all logbooks) and probably any informational entries too. Save or copy the exported ADIF file to a medium suitable for long-term storage such as a USB memory stick (good for about a decade) or CD-ROM (good for about 25 years to maybe 5 centuries for "archive quality" disks stored under perfect conditions) and store the archives somewhere safe, away from the shack.

Hinson tip: when you are ready, click the <**Start**> button to run the export function and generate the file. Hitting <**Return**> does nothing since there is no default button on this form.

Copyright © 2026 Robert C. Furzer K4CY



◀ **Setup custom ADIF fields**[66] lets us add fixed and variable information to the exported QSOs in the ADIF file.

The field *names* must be in the specified ADIF_FORMAT – see the ADIF specification for details. The specified *content* can be in any text format and length. The fields and content specified are then appended to every QSO record exported.

Click <**Apply**> to save the custom setup and return to the log export form.

When we click the <**Start**> button, Logger32 starts exporting QSOs in ADIF format. Click <**Abort**> in case of panic.

Any USER fields are included and any custom fields are appended to the ADIF QSO records ▼

```
ADIF Export from Logger32 Version [4.0]
Copyright 2001-2024  Robert C. Furzer.
Logs generated on 06 Feb 24 at 16:21 UTC by : ZL2IFB.
File output restricted to QSOs by : All Operators - All Bands - All Modes
<PROGRAMID:8>LOGGER32
<PROGRAMVERSION:7>4.0.317
<EOH>

<BAND:3>20m <CALL:6>IK0IXI <CONT:2>EU <CQZ:2>15 <DXCC:3>248 <FREQ:9>14.023740 <ITUZ:2>28 <SRX:4>1999 <MODE:2>CW <NAME:5>Fabio
<NOTES:17>FOC Marathon 2024 <OPERATOR:6>ZL2IFB <PFX:3>IK0 <LOTW_QSL_SENT:1>Y <QSO_DATE:8:D>20240203 <TIME_ON:6>175623
<RST_RCVD:3>599 <RST_SENT:3>599 <K_INDEX:1>0 <TIME_OFF:6>175623 <SFI:3>143 <A_INDEX:1>3 <APP_LOGGER32_QSO_NUMBER:6>130860
<STATION_CALLSIGN:6>ZL2iFB <MY_RADIO:5>FT847 <MY_PWR:3>100 <MY_ANT:15>TRIBANDER_WIRES <MY_NAME:4>Gary <EOR>
```

Hinson tip: generating and saving log exports to disk require a fair amount of CPU effort. If your DX cluster connection is busy handling lots of spots, that takes priority and slows down the export process. If you have a large log to export and/or a slow PC and are as impatient as me for the log export to complete, right-click the DX Spots window and tick <**Hold off DX spots**> before or during the export, not forgetting to un-tick it again once the export is swiftly completed.

Hinson tip: the custom ADIF fields are 'sticky'. Having defined custom fields, all future log exports will contain the custom fields … until you either redefine or delete them. If, like me, you later forget having set them up, it may be confusing to discover the custom info in your ADIFs … so if they were only needed for a specific purpose (such as a contest entry), while you remember, after the custom export go back into the log export function, click the <**Setup custom ADIF fields**> button, erase your data and click <**Apply**> to save the blank form ready for your *next* log export.

[66] Internally, Logger32 only supports the ADIF fields listed in the Logbook setup chart. Non-standard ADIF fields can be *exported*, but will be ignored upon log *import*.

Copyright © 2026 Robert C. Furzer K4CY

3.2.4 Comma delimited (.csv) log export



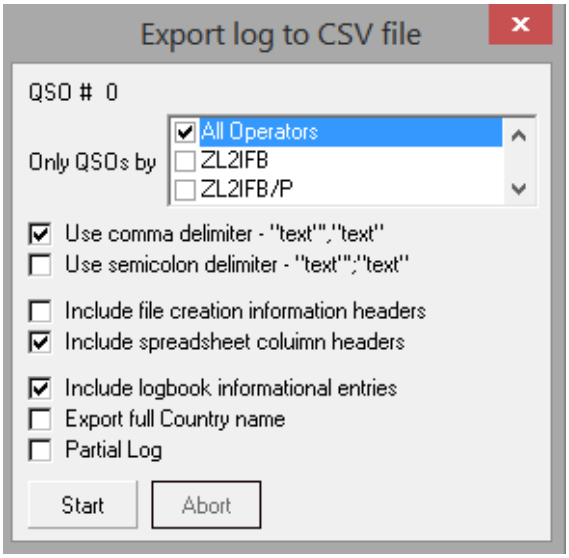
```
CSV Export from Logger32 Version [4.0]
Copyright 2001-2020  Robert C. Furzer.
Logs generated on 14 Nov 20 at 00:02 UTC by : ZL2IFB.
File output restricted to QSOs by : All Operators

"ADDRESS","DISTANCE","ARRL_SECT","BAND","CALL","CNTY","COMMENT","CONT","CONTEST_ID","CQZ","APP_LOGGER32_USER_1","DXCC","FREQ","GRIDSQUARE
","IOTA","ITUZ","APP_LOGGER32_USER_2","APP_LOGGER32_USER_3","MODE","SUBMODE","NAME","NOTES","OPERATOR","PFX","PROP_MODE","QSL_RCVD","QSL_
SENT","QSL_VIA","QSLMSG","QSLRDATE","QSLSDATE","QSO_DATE","APP_LOGGER32_QSO_DATE","QTH","RST_RCVD","RST_SENT","RX_PWR","SAT_MODE","SAT_NA
ME","SRX","SRX_STRING","STATE","STX","STX_STRING","K_INDEX","TEN_TEN","SOTA_REF","TIME_ON","TIME_OFF","TX_PWR","SFI","A_INDEX","eQSL_QSL_
SENT","eQSL_QSL_RCVD","LOTW_QSL_SENT","LOTW_QSL_RCVD","FREQ_RX","BAND_RX","CREDIT_SUBMITTED","CREDIT_GRANTED","APP_LOGGER32_QSO_NUMBER","
APP_LOGGER32_LOTW","APP_LOGGER32_QSL","APP_LOGGER32_eQSL","APP_LOGGER32_SUBMITFORDXCC"
"","19651","","20m","ZB2ER","","","EU","","14","","233","14.076323","IM76","","37","","","FT8","FT8","","","ZL2IFB","ZB2","","N","N","","
","","","20201112","12 Nov
20","","-15","-13","","","","","","","","","","","04:33","04:33","100","","","N","N","Y","N","14.075646","","","","00121284","","N","N
","N"
"","12604","","15m","K7ACT","AL,Baldwin","","NA","","03","","291","21.075329","EM60dk","","06","","","FT8","FT8","","","ZL2IFB","K7","","
N","N","","","","","20201112","12 Nov
20","","-14","-09","","","","","","AL","","","","","","20:59","20:59","100","","","N","N","Y","Y","21.075627","","","","00121285","","N",
"N","N"
"","13529","","15m","KE7NJ","NC,Wake","","NA","","03","","291","21.075329","FM05oo12","","06","","","FT8","FT8","","","ZL2IFB","KE7","","
N","N","","","","","20201112","12 Nov
20","","-08","-11","","","","","","NC","","","","","","21:00","21:00","100","","","N","N","Y","Y","21.076215","","","","00121286","","N",
"N","N"
```

▲ **C**omma **S**eparated **V**ariable files are text files with the data values for each record separated by delimiter characters - typically commas or semicolons. CSV files may also contain header lines before the data records, containing metadata about the file (*e.g.* its source and date), and titles or names for the data fields.

CSV files can be imported into many other programs, including Microsoft Excel, although you *may* need to delete or edit the header lines if they mess up the formatting.

CSV log exports have these options ►



1. **Only QSOs by**: select the appropriate operator(s) whose QSOs you wish to export, or choose "All Operators".
2. Use comma|semicolon delimiter – you choose.
3. Include file creation information|spreadsheet column headers: your choice again.
4. **Include logbook informational entries**: export informational entries from the logbook rather than skipping them.
5. **Export full Country name**: if ticked, an additional field containing the country name for each QSO will be included in the export file.
6. **Partial Log**: see above.

3.2.5 Exporting ADIF modes and submodes

When Logger32 exports the mode:

1. If the logged QSO mode is an ADIF mode, that is what gets exported *e.g.*

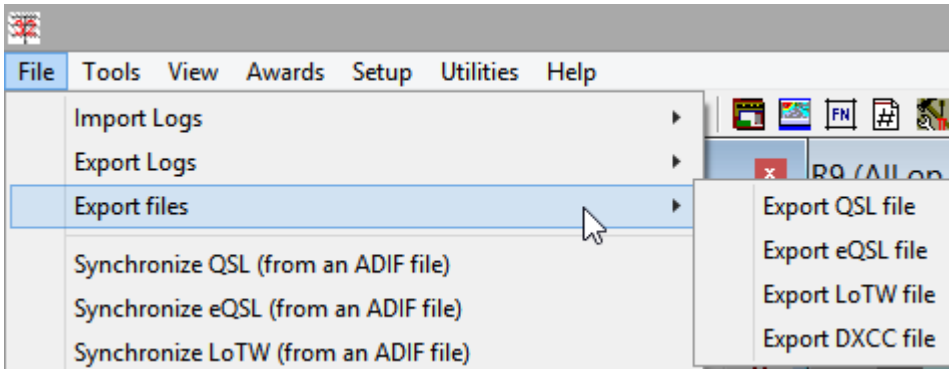
<BAND:3>40m <CALL:6>YC9IPJ <CONT:2>OC <CQZ:2>28 <DXCC:3>327 <FREQ:8>7.076299
<GRIDSQUARE:4>OI81 <ITUZ:2>54 **<MODE:3>FT8** <OPERATOR:6>ZL2IFB <PFX:3>YC9
<APP_LOGGER32_LoTW:1>Y <QSO_DATE:8:D>20201102 <TIME_ON:6>172300
<RST_RCVD:3>-08 <RST_SENT:3>-01 <TIME_OFF:6>172359 <TX_PWR:3>100
<APP_LOGGER32_QSO_NUMBER:6>120904 <FREQ_RX:8>7.075801 <COUNTRY:9>Indonesia <EOR>

Copyright © 2026 Robert C. Furzer K4CY

2. If the logged QSO mode is an ADIF submode, the corresponding ADIF mode is output as <MODE> with the submode as a <SUBMODE> *e.g.*

<BAND:3>30m <CALL:6>SM7HZK <CONT:2>EU <CQZ:2>14 <DXCC:3>284 <FREQ:9>10.142497 <GRIDSQUARE:6>JO76hx <ITUZ:2>18 **<MODE:3>PSK <SUBMODE:5>PSK31** <NAME:2>Bo <OPERATOR:6>ZL2IFB <PFX:3>SM7 <QSL_RCVD:1>Y <QSL_SENT:1>Y <QSLRDATE:8>20131010 <QSLSDATE:8>20131010 <QSO_DATE:8:D>20130826 <TIME_ON:6>051156 <RST_RCVD:3>599 <RST_SENT:3>559 <STX:3>991 <K_INDEX:1>2 <TIME_OFF:6>051156 <SFI:3>113 <A_INDEX:1>9 <LOTW_QSL_SENT:1>Y <LOTW_QSL_RCVD:1>Y <APP_LOGGER32_QSO_NUMBER:5>46569 <COUNTRY:6>Sweden <EOR>

3.3 Exporting other data files



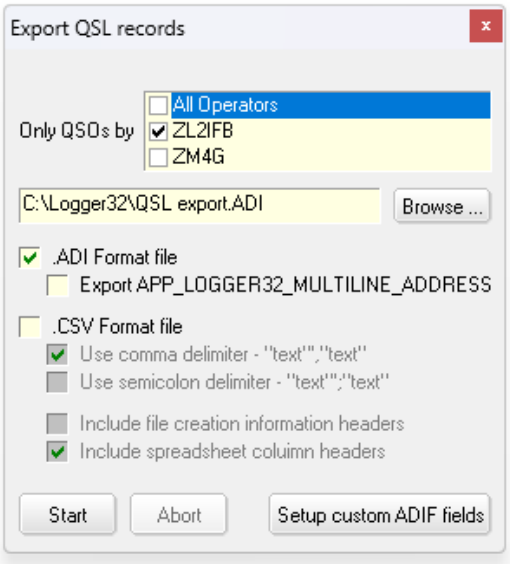
Aside from exporting your entire or partial log, Logger32 can export flagged QSOs for QSLing purposes.

◀ Click **File** ⇨ **Export files** then click the type of file you want to generate.

3.3.1 Export QSL file

Commercial QSL card services such as URE's QDure can prepare, print and dispatch QSL cards for us, taking and printing the QSO details from a data file we supply onto postcards of our design. We reduce the time, effort and cost of printing or writing the QSO details on each card and posting them to the recipients or QSL bureaux.

Logger32 can generate an ADIF or CSV file of QSO records from QSOs in our log that have been flagged (marked) to send QSL cards ▶



Click to tick the operator/s (making sure that we are exporting QSOs made with the callsign that is to be printed on the QSL cards) and the type of file, and name the output file, then click the <**Start**> button to set the process running.

We have the option to include a custom ADIF field for multi-line addresses, assuming we have recorded the recipients' multi-line postal addresses in the log.

For **C**omma **S**eparated **V**ariable exports (typically for use in spreadsheets or databases), we have a choice of record delimiters and column headings for the data fields.

Copyright © 2026 Robert C. Furzer K4CY



◀ Click the <**Setup custom ADIF fields**> button if you wish to add ADIF fields with fixed (predetermined) data values to every QSO in the output file …

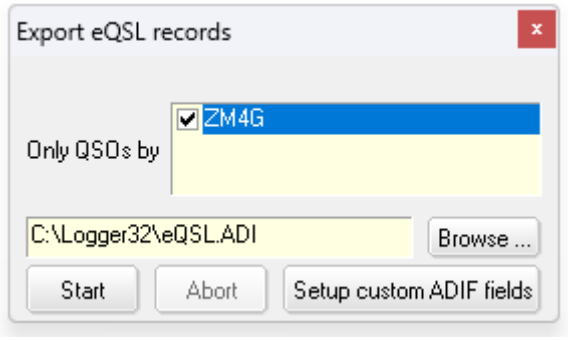
◀ … or to convert the data from USER fields in your log into standard ADIF fields in the output file …

◀ … or to identify QSOs with now deleted DXCC entities.

Hinson tip: these custom ADIF field settings are sticky *i.e*. they are saved in an .INI file and will be used on subsequent file exports of the same type … so if you only intended to customize a single export for some specific reason, right after the export, go back to the same export function, open <**Setup custom ADIF fields**> and clear the customizations. Otherwise you may at some future point be surprised to see unusual data in your exported files, having long since forgotten about this.

3.3.2 Export eQSL file

'Electronic QSLing' involves matching QSO details from both parties to a QSO, supplied in the form of data files, normally in the ADIF format. Online electronic QSL services are provided by eQSL.cc, QRZ.com, ARRL (LoTW) and others.



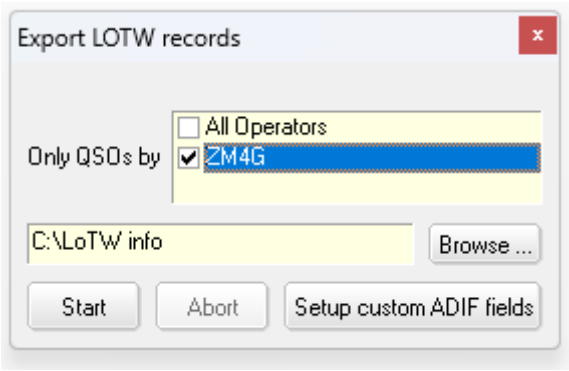
◀ The eQSL export function is easy to use. It simply generates an ADIF file of QSOs made by the ticked operator/s that have been flagged with<**Send eQSL**> in the open log, then clears the flag and sets <**eQSL sent**> instead. Either edit the file name directly or click <**Browse**> to find the relevant folder.

Copyright © 2026 Robert C. Furzer K4CY

See above for info on the <**Setup custom ADIF fields**> button.

Having generated the ADIF data file, login to your chosen electronic QSL service then upload the file to update your online records – or do whatever you want with the information.

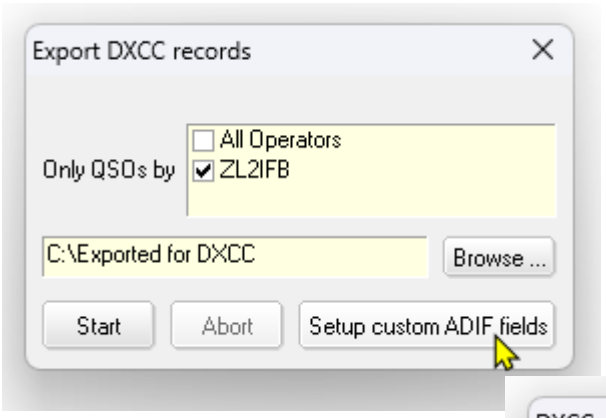
3.3.3 Export LOTW file



This function generates an ADIF file of QSOs that have been flagged with <**Send LoTW**> ready to be signed and uploaded to LoTW using the TQSL program ▶

Notice the <**Setup custom ADIF fields**> button here too.

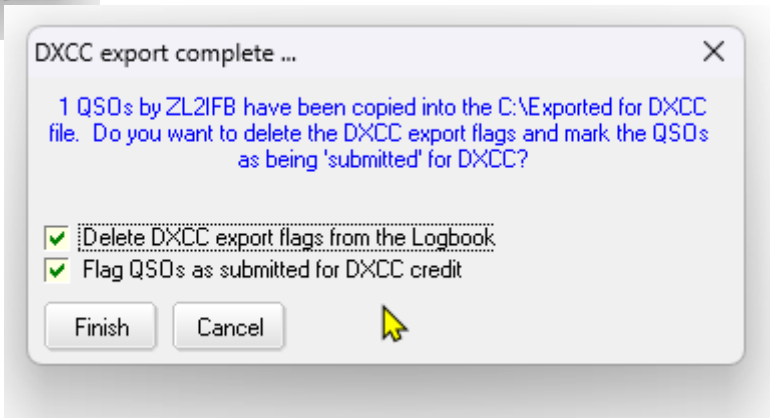
3.3.4 Export DXCC file



The <**DXCC file export**> function is clever: it generates an ADIF file of **QSOs for which we have received QSL cards but *not* LoTW confirmations.** We would normally submit these QSL cards for DXCC card-checking either to ARRL HQ or through the nearest DXCC card checker, as part of an Online DXCC application.

◀ There's another <**Setup custom ADIF fields**> button here.

Having generated the data file, Logger32 offers to delete the DXCC export flags and flag the exported QSOs as "Submitted for DXCC credit" if we wish – simply tick the boxes then click <**Finish**>, or click <**Cancel**> to close the function without updating any flags ▶



Hinson tip: personally, I prefer *not* to update the flags until I have found the relevant QSL cards in my shoeboxes, submitted them for checking and waited patiently for the results, in case I lose the plot during that process. That way, I can easily regenerate the 'DXCC file' again later on, when I remember what I was doing. Once the process is complete, I run the export once more, this time clearing the flags.

Copyright © 2026 Robert C. Furzer K4CY

3.3.5 *Send* ⇒ *Sent* flags

After the export files have been generated, we are invited to update the flags for the associated QSOs ▶



Click <**Yes**> to clear the <**Send QSL|eQSL|LoTW**> flags from the QSOs *and* set the <**QSL|eQSL|LoTW sent**> flags *and* record the date they were sent, or click <**No**> to leave the flags and dates unchanged.

Updating the flags is the obvious choice since it means those QSOs will not be exported again if we re-run the export function … but leaving them unchanged gives us the chance to [sign and] submit them and get some sort of confirmation of receipt back from the recipient systems. If that process goes to plan, we can run the file export function again, re-generating the export file but this time clicking <**Yes**> at this final stage to clear the flags. However if, for some reason, the submission process fails (*e.g.* if the recipient system is offline for maintenance or due to ransomware), we can try [signing and] submitting the same exported file once again, or start over by repeating the export from scratch.

Hinson tip: it is not a bad idea to export your log from time to time – either the entire log or just the QSOs made within, say, the past year – and [re]submit them to the QSL services, if there is any possibility that some QSOs might have been lost in the export-[sign-]upload-import process. By default, TQSL will automatically skip any QSOs that it thinks have already been signed and uploaded to LoTW, but you can even override that block if you wish: LoTW checks and skips duplicate QSOs on receipt anyway. As a free bonus, you now have a fresh ADIF backup of your log to save somewhere safe, just in case.

Just wanted to say thanks for the issue of V4. My L32 log goes back 20 years now and I can't envisage playing radio without it. With some trepidation (because I know nothing about PCs and was sure something would go wrong!) I updated last night and everything was fine. No problems. I shall now spend time trying to get to know all the enhancements in V4. So Bob and all the developers, that's a huge amount of work you have put in, my thanks.

Ken, G3XPO

Copyright © 2026 Robert C. Furzer K4CY

3.4 Import-export FAQs

Q. I loaded my ADIF log into Logger32 and it reported dumping reject QSOs in the *BAD.ADI* file, apparently. How do I find this file? What do I do with it?

A. Look in your Logger32 folder *i.e*. *C:\Logger32* by default. You can open and edit the *BAD.ADI* file using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad. The file includes comments explaining the issues that Logger32 found.

After you have reviewed, corrected and saved it with a new name[67] (such as *GOOD.ADI*), you can import the corrected ADIF file into Logger32, hopefully without any more errors.

Many of the errors found in *BAD.ADI* are problems with the state and county fields *e.g.*

> Error in the STATE field: <CALL:5>KL7QW <CNTY:9>MO,Camden <CONT:2>NA <CQZ:1>4 **<DXCC:1>6** <ITUZ:1>1 <MODE:3>SSB <PFX:3>KL7 <QSL_SENT:1>Y <QSO_DATE:8:D>20001118 <RST_RCVD:2>57 <RST_SENT:2>56 <STATE:2>MO <EOR>

Despite what it says, the error here is *not* in fact the STATE field. This record is showing a contact with KL7QW. The prefix originally indicated Alaska (DXCC entity code 6). MO is not a valid Alaskan state! However, KL7QW lives and operates from Camden, MO, hence his state is correctly identified in the ADIF record. His DXCC entity code is wrong and needs to be changed to 291 (mainland USA). Changing <DXCC:1>6 to <DXCC:3>291 will allow the corrected QSO record to be imported.

If you have a *lot* of QSO records in *BAD.ADI* but lack the time and intestinal fortitude to check and correct them all manually, you have two choices:

1. If you were synchronizing your log with confirmations downloaded from LoTW, re-run it but this time de-select the county field. Logger32 will completely ignore the county but will check/update the other selected fields.
2. Edit the *BAD.ADI* file, searching and replacing **<CNTY:** with **<xCNTY:** then save the file with a new name (*e.g. GOOD.ADI*) and import that into Logger32. Since there is no ADIF-defined xCNTY field, Logger32 will *automatically* ignore that invalid field but will check/update the other selected fields.

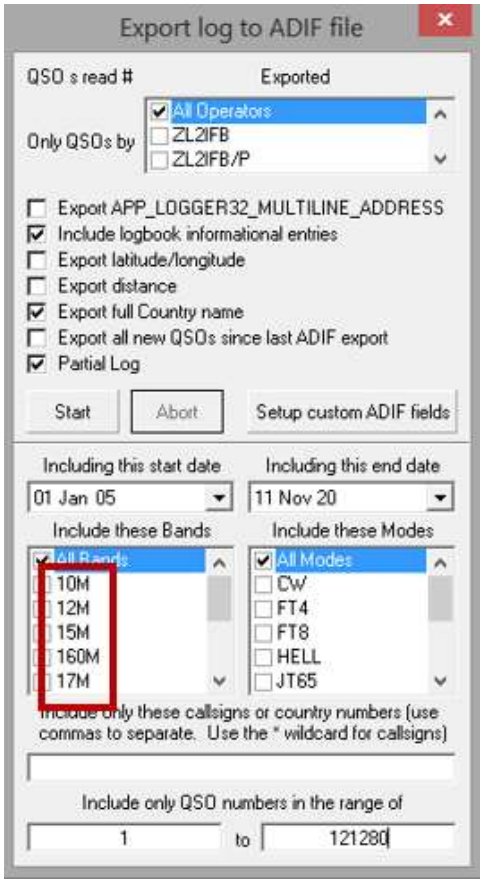
By the way, if you do this often, it's handy to add an entry to your <**Utilities**> menu which invokes an editor such as MS Notepad or WordPad, TED Notepad, or ADIFmaster, with *C:\Logger32\BAD.ADI* as a command line parameter … and perhaps something similar for checking/editing other temporary files.

[67] Logger32's file import function refuses to accept a file called BAD.ADI … which is good.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why don't partial log exports by band or mode work for me?

A. Does the export log form on your system show bands with capital M's, like this? ▶



Oh oh.

The root cause of the export problem is a change in the way Logger32 records bands. Previously under version 3, capital M's were used - incorrectly - to denote metres *e.g.* **160M**. The new, sexier and improved Logger32 version 4 uses SI-compliant lowercase m's *e.g.* **160m**.

The bands listed on the export log form are drawn from a quick scan of your log, so it appears you still have QSOs with capital M bands recorded in your log.

You *could* work systematically through your log to find and update the erroneous QSOs manually … but it is *much* easier and quicker to recalculate your statistics which painlessly corrects the bands to lowercase m's as part of the process.

After that one-time recalc (well, OK, once per v3 log, to be precise), partial exports by band should work correctly in future, SI-compliantly.

Q. Why is my ADIF import taking so long? After every few QSOs it pauses.

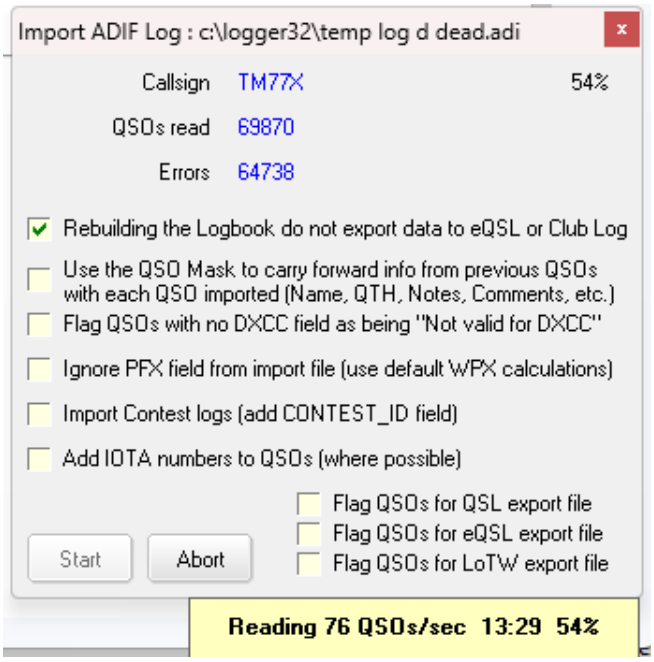
A. There is some heavy lifting going on behind your screen, interpreting and checking the ADIF data, creating new QSO records, updating statistics, updating a bunch of database and stats files on disk. The database needs time to extend its allocated storage space, repeatedly. It is not instantaneous. However, there are some things you can try to speed it up a little.

> **Hinson tip**: processing incoming DX spots is quite CPU-intensive and visibly interrupts the importation, so temporarily disconnect from the DX cluster/s *before* importing large ADIF logs, to smooth the process and speed things up.

Copyright © 2026 Robert C. Furzer K4CY

Likewise, options on the import form such as <**Add IOTA numbers …**> and <**Use the QSO mask …**> also slow down the importation process fractionally.

If you are loading a large log (say 50,000+ QSOs) or have a slow PC, un-tick everything except <**Rebuilding the Logbook …**> ▶



Possibly, temporarily disable your antivirus scanner or add and exception for *C:\Logger32\Logger32.3exe* just in case it is noticeably slowing the disk accesses.

If, like me, you foolishly forgot to do any of this before launching the import and can't face the prospect of waiting *hours* for it to complete, click <**Abort**>, change your settings and start it again with the same ADIF file and log. The import routine will skim very rapidly past all the QSOs you already imported (don't worry: they are not duplicated but are simply ignored as 'errors'), then settle down to complete the import job at its normal measured pace.

Hinson tip: the advice on improving PC performance applies here. Replacing a magnetic disk with or adding a PCIe NVMe solid state disk on which to store your log is recommended. Doing so may require a replacement motherboard … so this could be an opportunity to replace the shack PC with something more modern and capable – think Flex rather than AR-88.

Q. Can I import a log containing my QSOs made using various callsigns?

A. When importing an ADIF file, Logger32 checks the QSO records for the optional ADIF OPERATOR field. According to the ADIF specification, the OPERATOR is "the logging operator's callsign [and] if STATION_CALLSIGN is absent, OPERATOR shall be treated as both the logging station's callsign and the logging operator's callsign."

If the OPERATOR field is missing from any QSO record in the imported ADIF, Logger32 records the *current operator's callsign* as the OPERATOR for the imported QSO. You may not want that. So, check the ADIF file for the OPERATOR field first, using a plain text editor such as Notepad. Provided the correct OPERATOR is recorded for all the QSOs, go ahead and import it. If not, you can manually edit (*e.g.* using ADIFmaster) or regenerate the ADIF to include the correct OPERATOR for every QSO record, or break the log into sections by station callsign, setting the current operator's callsign accordingly before importing each section.

Copyright © 2026 Robert C. Furzer K4CY

Q. I have imported logs containing Canadian QSOs identified as Labrador province, although I understand that "LB" is *not* a valid Canadian province. I didn't see any warning or error message. Isn't this a bug? Have I earned a prize?

A. No, Logger32 silently translates "LB" to "NL" when importing Canadian QSOs. The ADIF standard specifies that the Canadian primary subdivision code "NL" stands for "Newfoundland *and* Labrador". There is no separate "LB" code for Labrador alone in ADIF. The problem lies in whichever logging software accepts and outputs "LB" as a Canadian province – not Logger32. Logger32 is saintly.

Q. How do I start uploading to LoTW?

A. This process sets up Logger32 for easy uploading in the future, assuming that you haven't been flagging all your QSOs for LoTW already.

1. Open Logger32 and export your entire logbook as an ADIF file using **File ⇨ Export logs ⇨ ADIF (.adi) log**. Scribble down a note of the folder and filename you've chosen if you might not remember it in a moment …
2. Close Logger32 and open File **E**xplorer using <**Win**+**E**>. Navigate to the *C:\Logger32* folder and move the four logbook database files (*[logbook name].ISD, .ISF, .ISL* and *.ISM*) to another folder for safekeeping.
3. Open Logger32. Your logbook will be completely blank at this stage but don't panic – it's intentional.
4. Right-click the log entry pane and click **Setup ⇨ QSLing & QSL Export**. Click to tick <**Flag QSOs for LoTW**>. Having done this, any QSOs you log in future will automatically be flagged for LoTW export.
5. Click **File ⇨ Import Logs ⇨ ADIF (.adi) file** and enter the filename of the ADIF file you exported in step 1. Select the <**Rebuilding the logbook …**> option and click <**Open**>.
6. On the import form, tick <**Flag QSO for LoTW export file**> and any other appropriate options, then click <**Start**>. This step flags all your imported QSOs for LoTW export as they fill the empty logbook.
7. After the import, have a quick look at your logbook to make sure it's intact. Breathe a sigh of relief.
8. Click **File ⇨ Export files ⇨ Export LoTW file** to generate the file for LoTW.
9. When prompted, click <**Yes**> to mark the QSOs as having been sent to LoTW in your log.
10. Launch TQSL by double-clicking the TQSL key icon that magically appeared on your desktop when you installed the TQSL program ▶ 
11. In TQSL, follow the prompts to sign and upload the LoTW export file to LoTW.

Following the initial process, in future all you need to do is periodically export the LoTW file, then sign and upload it using TQSL.

By the way, uploading log updates to online log systems such as Club Log and eQSL is even easier after an LoTW update since you already have the ADIF files and you don't need to sign/process them through TQSL. Simply login to the relevant website, find the log update/upload option and follow the prompts.

Copyright © 2026 Robert C. Furzer K4CY

Q. I boobed. Having imported my QRZ.com log as an ADIF file into my Logger32 log, I now have *loads* of duplicated QSOs with only minor differences. How do I get rid of them without having to find and delete each one individually?

A. Simply **restore the log backup that you made just before importing the ADIF**, as instructed. This reverts all the changes, taking you back to how things were, pre-boob.

If you ignored the dire warnings and don't have that backup, your task is more arduous. Due to the risk, Logger32 does not allow us to select and delete or amend multiple QSOs – a deliberate design decision by Bob K4CY to reduce the boob-risk. So:

- Since the QSOs that you imported are numbered sequentially as a block after the last QSO from your original pre-import log:
  - Find the first imported QSO at the start of that block by sorting your logbook by QSO number. Check the imported ADIF file for the callsign or other information identifying the first QSO record in the file, or simply scroll back from the end of your log until you notice a distinct discontinuity in, say, the QSO dates. Take note of the number of the last QSO in your original pre-boob log.
  - Generate a *partial* ADIF export from your log by QSO number from 1 to the last original QSO number: that is your original log. Now create a new, empty logbook and import the original log. Job done; or …
  - Provided there are not too many, manually delete the imported QSOs individually, now conveniently listed as a block at the end of your log.
- Alternatively, export your complete but messed-up log as an ADIF file and clean it up in an ADIF-compliant program such as ADIFmaster using its ability to sort, select and delete multiple QSOs, then import the cleaned ADIF into a new Logger32 logbook.
- Learn the harsh lesson. Backups are important! Love your backups!

Q. Can I synchronize Logger32 across multiple computers?

A. Yes, it is possible to synchronize your logbook across two or more computers depending on your situation, and your understanding of 'synchronize'. In sequence from relatively safe down to risky, here are *six* possible methods:

1. **ADIF transfer**: output the log in ADIF format from one computer, transfer it to the second computer by some means (*e.g.* on a USB memory stick), the import it into the open logbook on the second computer using **File ⇨ Import log**. Logger32 will ignore any exact duplicates so only the new, non-matching records will be accepted onto the second machine. It is safer to open a new, empty logbook, avoiding any possibility of messing up an existing log on the second machine.

2. **Transfer a Logger32 logbook-only backup**: create and then copy the logbook backup *.zip* file from the main shack computer to the second computer by some means, unzip it there and open it in Logger32 on the second computer. The two installations of Logger32 should be at the same or very similar versions (database formats change occasionally but nearby minor versions are *usually* compatible). The screen resolution and layouts, ports *etc*. may differ between machines since they each have their own Logger32 configurations.

Copyright © 2026 Robert C. Furzer K4CY

3. **Transfer a Logger32 full (logbook plus user files) backup**: if you want the same screen layouts, ports, macros *etc*. on both machines you can *try* transferring the user files as well … but this can be very tricky, especially if the two computers are physically diverse with different screens, Windows versions, ports, peripherals, fonts *etc*. You may be better off transferring an ADIF log backup plus all the *.INI* files from *C:\Logger32*, then manually editing the *.INI'S* on the second machine, adjusting them as necessary for the system – or simply configuring the second machine to your liking from scratch. Think of it as a golden opportunity to reconsider all those configuration choices. Relish the freedom.
4. **Transfer the logbook files**: copy the logbook database files from the shack computer to the other computer memory stick. Logger32 can open and update the logbook on the memory stick from any computer onto which it has been installed … but be sure to keep an up-to-date backup somewhere safe in case the stick gets damaged or lost. This is a simple but risky way to run Logger32 on a laptop for a temporary excursion, using a copy of the shack logbook. Don't forget to update the shack/main station logbook, carefully, when you return (preferably by means of option #1 above, an ADIF export from the laptop, imported into the shack PC … a much safer approach).
5. **Transfer the database files**: this is possible with Logger32 but due to indexing restrictions, it is absolutely imperative that all the database files be kept and shared together, including the Country, Offset and Alias databases (4 files each, 12 files in total). They *must* remain together and distributed as a complete set of 12 files. If you use a group of database files from another computer, recalculate the statistics on your logbook to update the information tucked away in the Countries database.
6. **Cloud 'sharing'**: copy the database files (*all of them*) between computers via a cloud service such as Google Drive or Dropbox. This can be automated by creating a *.BAT* batch file on each computer to copy the files into *C:\Logger32* after backing up any existing files. Note that **only one installation of Logger32 can operate at a time**: while it is running, it holds the database files open so they cannot be accessed simultaneously by other computers. ***Logger32 is a single-user system.*** That's just how it is, sorry.

Copyright © 2026 Robert C. Furzer K4CY

4 Getting started with Logger32

> "The expert in anything
> was once a beginner"
>
> *Helen Hayes*

4.1 Screen layout

It may be tempting to play with *all* the new toys in Logger32 when you open the box … but there are so many of them, and so much to configure, that the choice can be overwhelming. Instead, we suggest **starting out with just the few windows and configuration details you actually *need***, gradually adding more as you become familiar with the program and how it works.

4.1.1 Choosing the information you need under <View>

Most of us *need*:

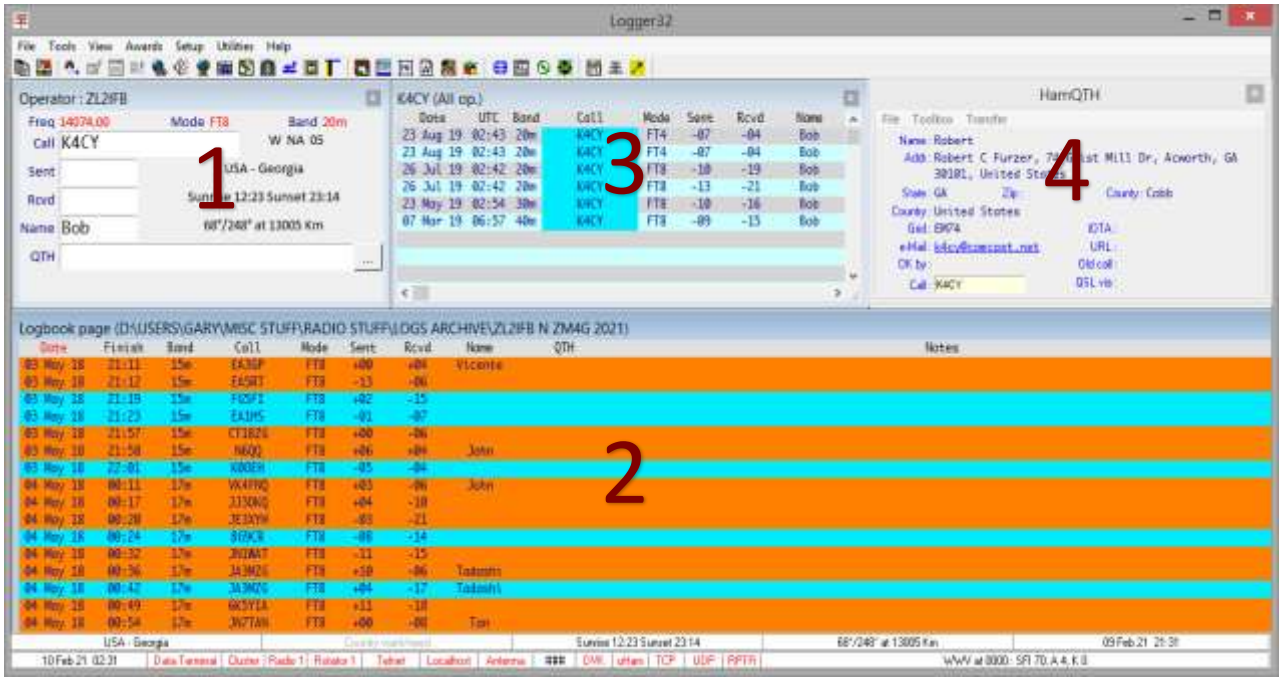
1. The log entry pane in which to log new QSOs.
2. The logbook showing QSOs already logged.

Two other very useful windows are:

3. Previous QSOs showing QSOs already logged with the station you are working.
4. Callsign lookups showing operator names, addresses *etc*. from HamQTH or other online databases.

The toolbar, DX information bar and status bar take up just one line each but give you ready access to additional information and functions that you will probably find useful in due course.

So, with just those few items selected, Logger32 might look *something* like this ▼



Copyright © 2026 Robert C. Furzer K4CY

The information shown within most windows can be configured, both in terms of what items are shown and how they are presented (*e.g.* the sequence of columns, the colors of text and backgrounds). On the minimalist example screen above:

- The log entry pane can have additional data entry fields *e.g.* to record QSL managers for the DX stations you work.
- The logbook and previous QSOs pane both offer an *extensive* choice of columns, plus configurable highlighting for QSOs that have been confirmed … and more, much more.

4.1.2 Adding more functions and windows

As you get comfortable with the minimalist Logger32 screen and functions and continue studying this User Manual, you will probably want additional functions and windows *e.g.*

- A DX cluster connection to receive and show DX spots.
- Maps showing the location of stations spotted and worked.
- A BandMap showing DX spots across the band you are currently using … and perhaps others.

So, you may well need to move and resize the windows to fit your display[68]. In the logbook and previous QSOs pane, it helps to display the columns that you refer to most often, together on the left of the screen, with less-frequently accessed columns on the right and perhaps out of sight (in which case you can scroll horizontally to see them).

If your radio is CAT-capable, you will almost certainly choose to connect it to Logger32 in order to capture frequency and mode information from the radio, and to QSY instantly when you click spots on the DX cluster pane or BandMaps. Other station hardware such as automatic antenna switches and rotators are worth connecting too, *when* you are ready: if you rush into it, you risk getting into a muddle and not properly using the powerful functions already available.

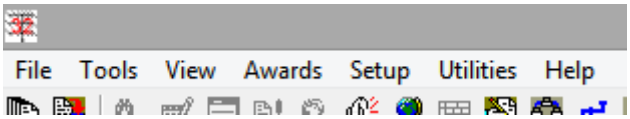
Eventually, as you run out of space, you may yearn for a larger screen or supplementary displays to show more BandMaps, and to operate digital modes such as FT# and JT65 – at least in the shack. Conversely, if you operate from mobile or portable locations, you may go the other way, trimming everything down to the bare minimum again to fit a small laptop/tablet PC.

4.1.3 Different configurations

If you enjoy different types of operating (*e.g.* DXing, rag-chewing and contesting, or LF, HF and VHF/UHF), or if the shack PC is used by more than one ham (*e.g.* you and your spouse), you can set up and save the screen layouts and other configuration items separately as Logger32 configurations.

4.2 Main menu

The text menu across the top left of Logger32's main window is known as the main menu ▶

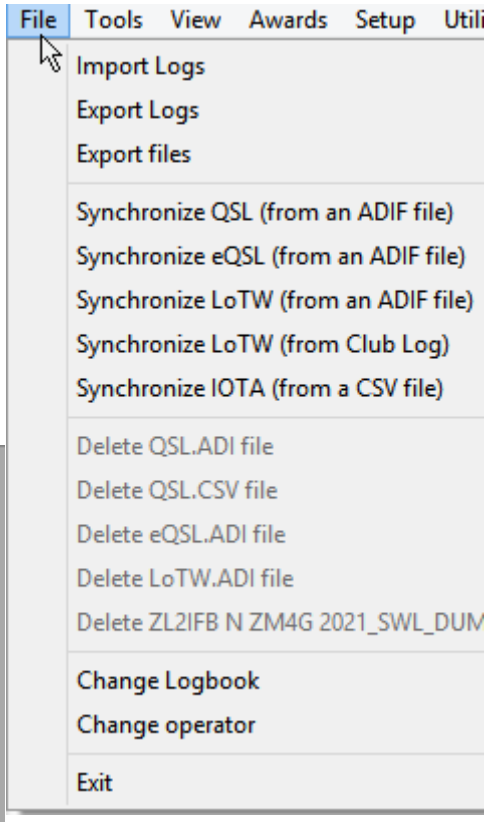


We'll take a *quick* look at all seven menu items next. You will find lots more information on all the functions and most of the settings elsewhere in this User Manual: click the underlined hyperlinks to jump directly to additional information on anything that catches your eye.

[68] Aki JA1NLX publishes a collection of example Logger32 screen layouts if you seek inspiration … and by all means send Aki a screenshot of yours once you settle on a layout that works nicely for you.

Copyright © 2026 Robert C. Furzer K4CY

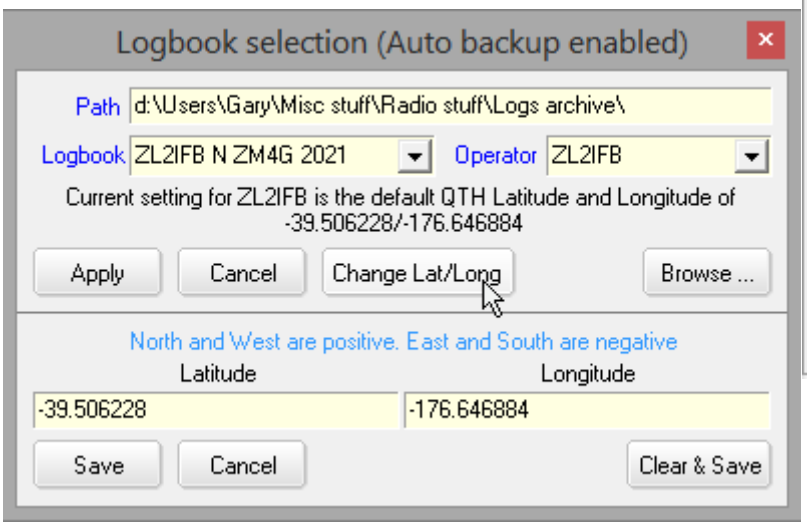
4.2.1 File



The <**File**> menu mostly concerns importing and exporting files of various kinds – logs and lists of QSOs for QSLing ▶

Use the <**Synchronize …**> options to update the QSL status of QSOs in the logbook to reflect confirmations received by various means on various dates.

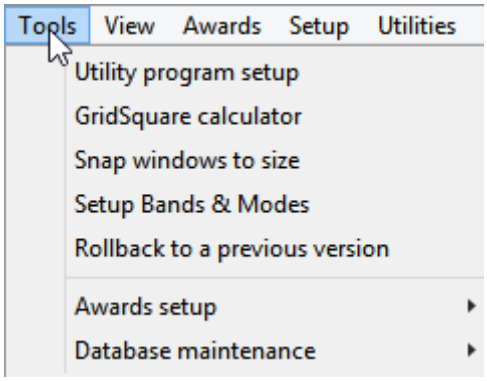
You may find the options to delete temporary files convenient: personally, I prefer to review and manage my folders and files through File **E**xplorer.



◀ <**Change Logbook**> is necessary if you use more than one logbook in Logger32 because only one logbook can be open at a time, although each logbook can be used for one or more of your callsigns (see just below). Use <**Change Logbook**> to create new logbooks as well. <**Change Logbook**> lets you define your QTH (station location). The latitude and longitude are used to calculate directions (bearings, azimuths) and distances and bearings to the stations you contact.

<**Change operator**> is where you specify your own callsign – specifically, the callsign you are currently using on-air to make QSOs. This callsign is recorded against future QSOs that you log.

4.2.2 Tools



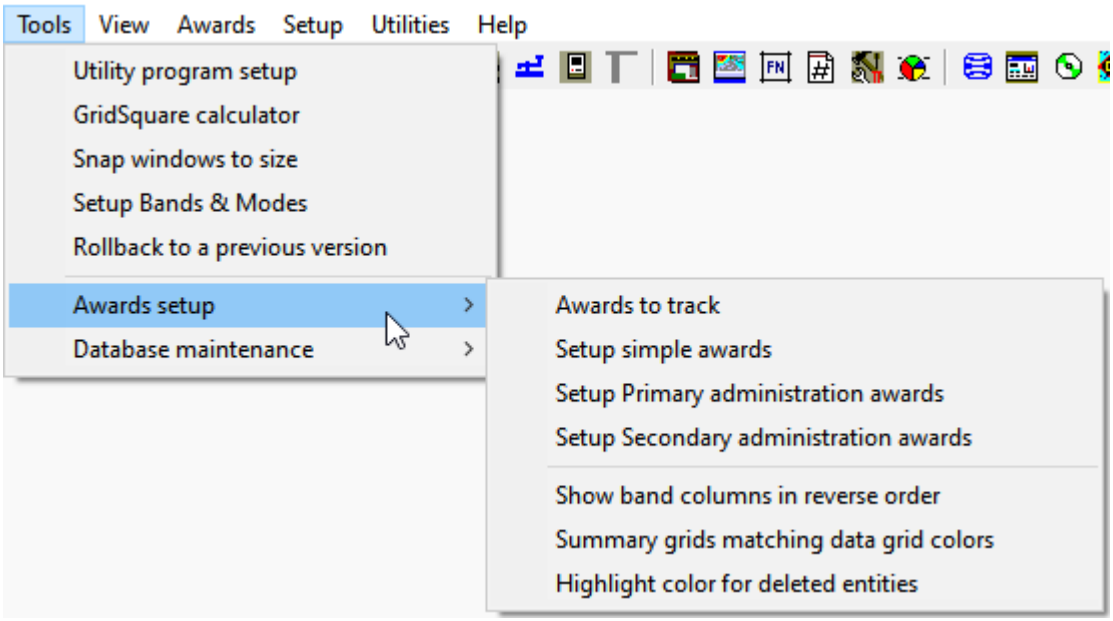
The <**Tools**> menu is a bit of a mixture … ▶

- **Utility** program **setup**: determines which programs are available on the <**Utilities**> menu.
- **GridSquare calculator**: calculate the distances and bearings to DX grid squares from your station location as defined using **Setup ⇨ My QTH (Lat/long)** from the log entry pane right-click menu.
- **Snap windows to size**: child windows within the main Logger32 window can be made to butt up against adjacent windows[69].

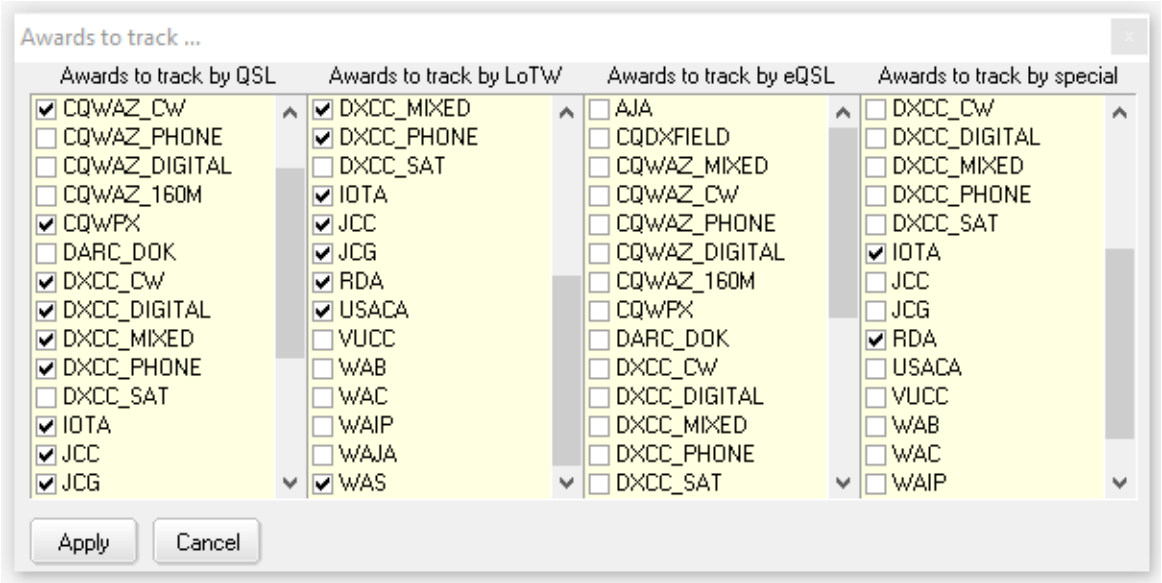
[69] You may ask why this option is here under <**Tools**> rather than under <**View**>. The menu structure has grown organically over the years. It is part of Logger32's charm.

Copyright © 2026 Robert C. Furzer K4CY

- **Setup Bands & Modes**: a critical part of using Logger32 is to define which bands and modes you intend to use and whether they are to be included in various statistics (such as the DXCC award reports) and the worked/confirmed table.
- **Rollback to a previous version**: if Logger32 isn't working properly, you can revert to an earlier version to find out whether the problem was inadvertently introduced by an update.
- **Awards setup**: Logger32 can track your progress towards awards, highlighting new ones as they are spotted on DX cluster or appear in your FT# decodes, and recording receipt of QSLs. Use this menu item to configure the awards you are chasing ▼

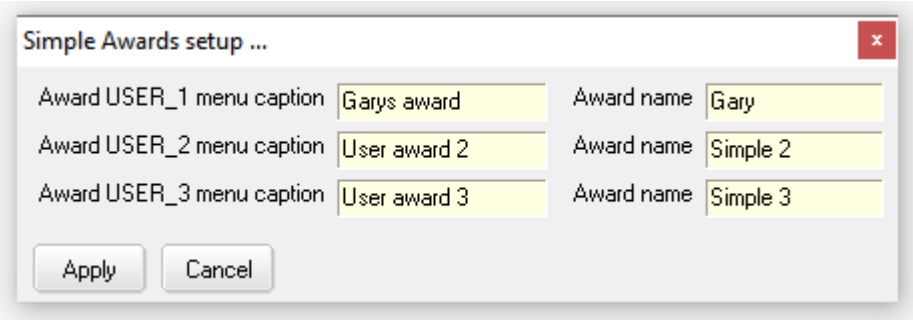


  - **Awards to track**: on this form, choose which awards you are interested in chasing, and tick the relevant columns if those awards accept QSL cards, LoTW confirmations, eQSL confirmations, or 'special' (something else *e.g.* entrants in the IOTA contest submit their logs, allowing the award administrators to check if you were worked and logged by a given IOTA station as you claim) ▼

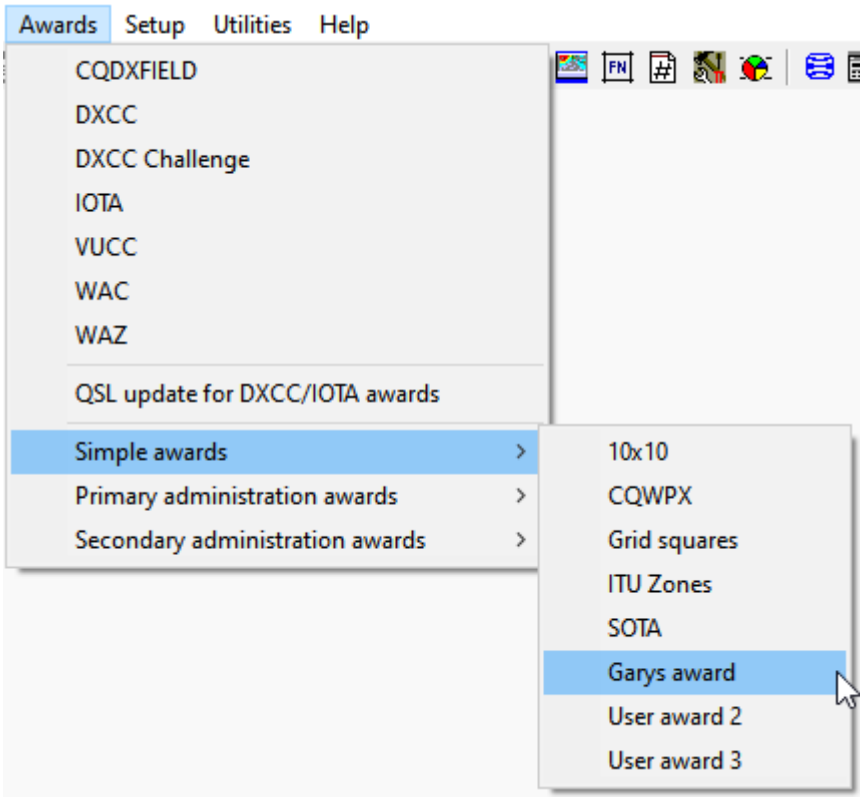


Copyright © 2026 Robert C. Furzer K4CY

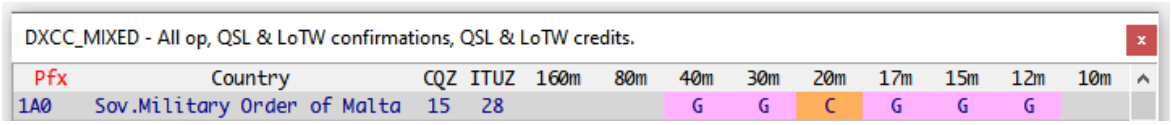
- **Setup simple awards**: aside from the simple awards already supported by default, you can track up to 3 additional awards based on working stations that qualify for the award (*e.g*. they are operating from one of the summits specified in the SOTA award rules) ▼



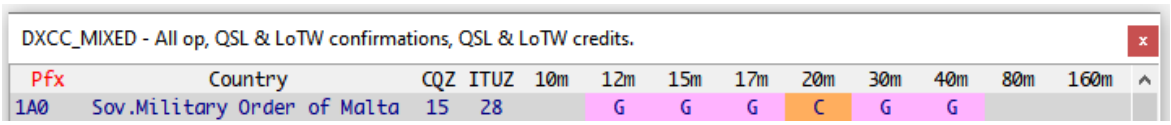
Their captions then appear on the awards menu ▼



- **Setup Primary administration awards**: such as WAS.
- **Setup Secondary administration awards**: such as USACA.
- **Show band columns in reverse order**: if you don't like the default order of the band columns on the award reports ▼ ...

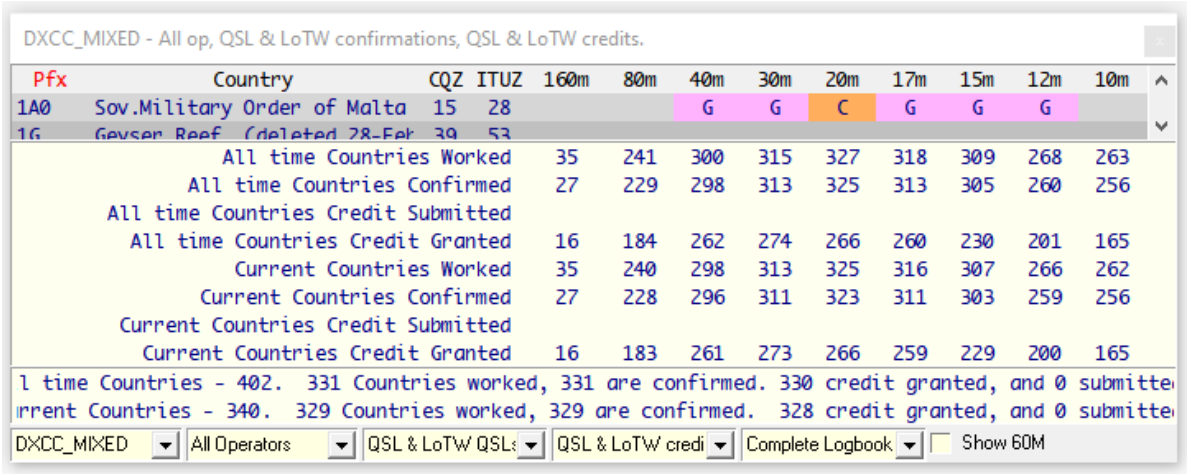


... tick this option to reverse the order ▼

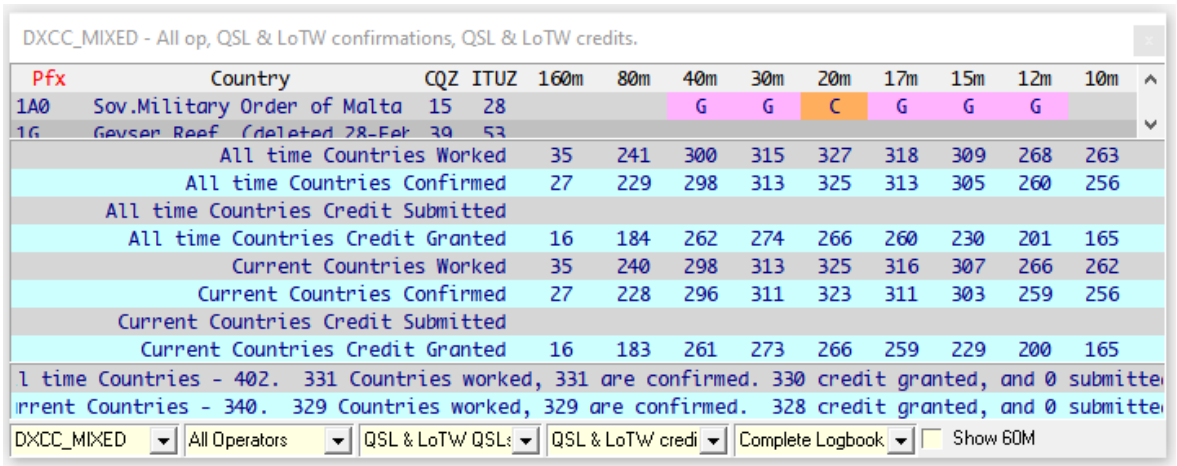


Copyright © 2026 Robert C. Furzer K4CY

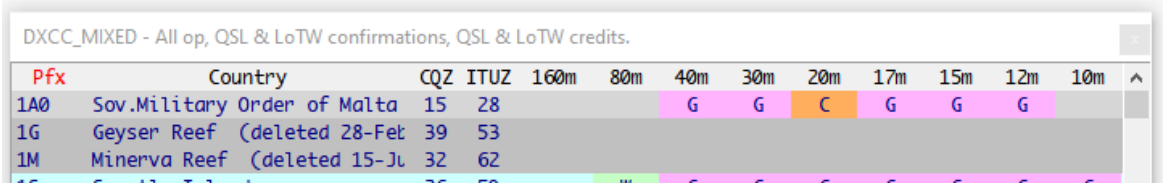
- **Summary grids matching data grid colors**: if you don't like the uniform coloring of the summary section at the bottom of the awards reports ▼ ...



... tick this option to show banded colors like the main body of the reports ▼



- **Highlight color for deleted entities**: pick the background color used on the awards reports for deleted entities *e.g.* a mid-gray▼



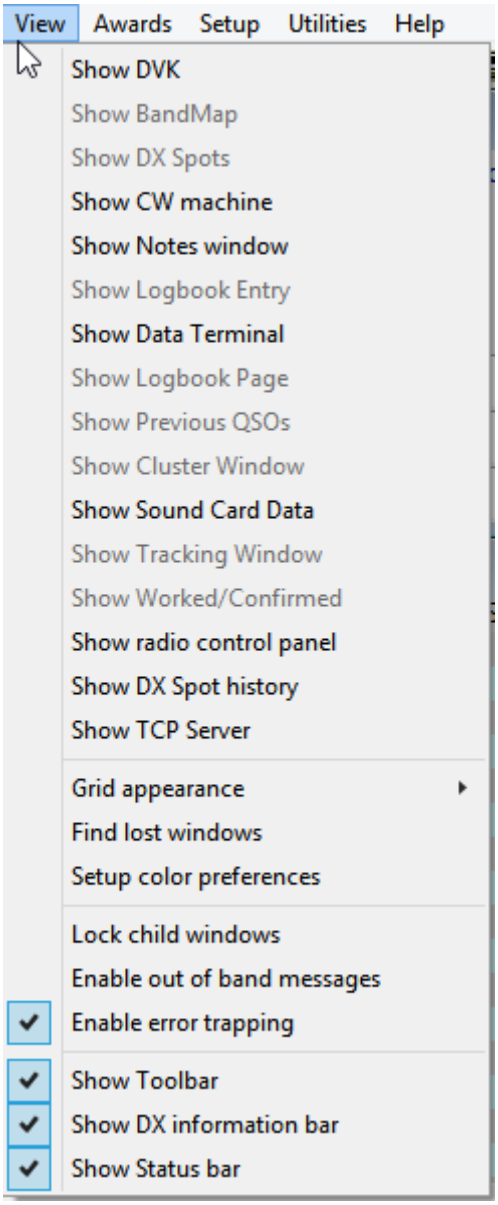
- **Database maintenance**: identifying which DXCC entities, states, continents, IOTAs *etc*. people are operating from requires looking things up in databases containing the relevant information. You can maintain the data manually *e.g.* adding prefixes or changing country names, but usually the automated updates are sufficient, easier and safer (less error-prone).

Copyright © 2026 Robert C. Furzer K4CY

4.2.3 View

- The top section of the <**View**> menu lets you display *most* of Logger32's windows.

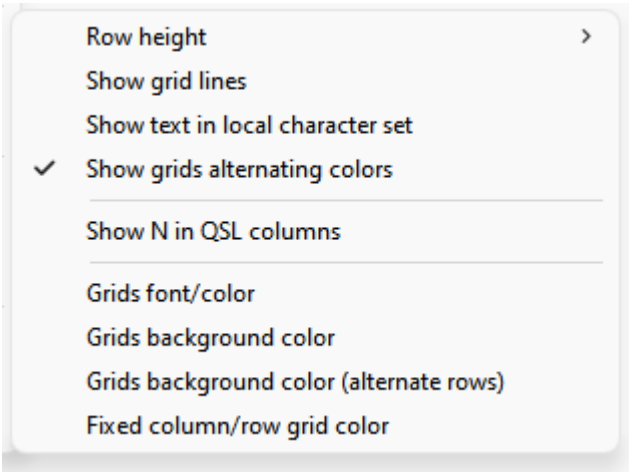
Windows that are already shown are grayed out ▶



- A few other windows are not listed here, such as the UDP BandMap, JTDX Control Panel, scratchpad and DVK Control Panel: those are opened from other menus and right-click combinations.

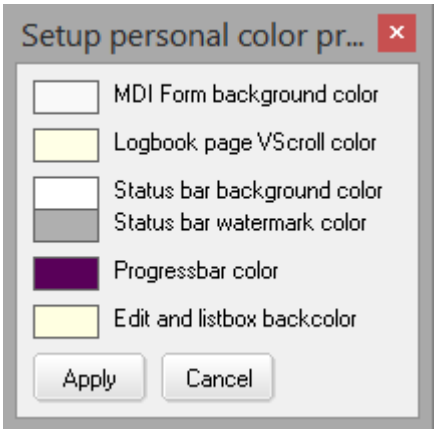
  Some windows can be opened automatically according to the mode you are using.

- **Grid appearance**: opens a submenu to setup and color the columns displayed in your logbook and previous QSOs "grids" (tables) ▼



- **Find lost windows**: is a workaround for annoying issues with Windows' handling of program windows. Sometimes, for various obscure reasons, Logger32's windows get hidden behind others or are moved off-screen and out of sight. **View ⇨ Find lost windows** puts them top-left of the screen, from where you can use, close, move or resize them, as you wish.

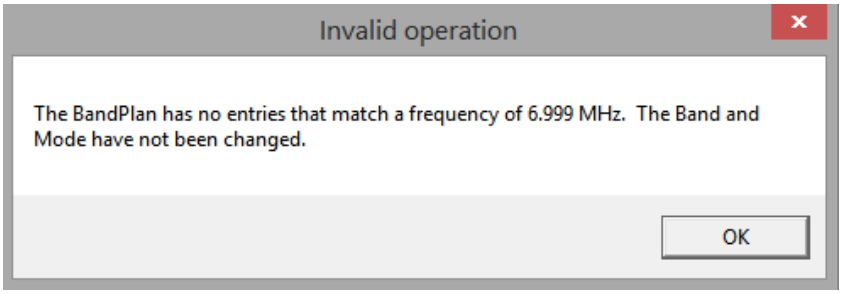
- **Setup color preferences**: choose some of the many colors in Logger32 *e.g.* for the background of Logger32's main window (MDI form) ▶



- **Lock child windows**: this is like a child lock for your car windows. Once locked, the child windows (or panes) within Logger32's main window can no longer be resized or moved … so set them up to your liking before locking them. If you change your mind, you can always unlock them, shift them around, and re-lock them. Just hope that your children don't discover how easy it is to get around the lock!

Copyright © 2026 Robert C. Furzer K4CY

- **Enable out of band messages**: if you tune your radio's VFO beyond the limits of the bands listed in your Bands & Modes table, with this option set Logger32 will a pop-up a warning message ▶



To reduce the chances of you transmitting out of band, the message is *intentionally* intrusive, requiring you to (a) move your VFO back into an amateur band, and (b) click the <**OK**> button to dismiss the warning. If you enjoy short wave listening, monitoring the broadcast and utility stations between the amateur bands (*e.g.* WWV), you probably want to disable the warning messages … and be careful where you transmit[70].

- **Enable error trapping**: Logger32 incorporates numerous internal error checks designed to prevent situations that might threaten the integrity of your logbook *etc*. This option enables additional checks to identify and report other adverse conditions or warning signs in Logger32, in Windows, in device drivers or in Other Places. Especially if you are experiencing problems with Logger32, it is a good idea to enable error trapping.



- **Show Toolbar**: display the row of 28 colorful icons ▲ near the top of Logger32's main screen, just below the text menu, giving one-click access to 28 popular program functions:

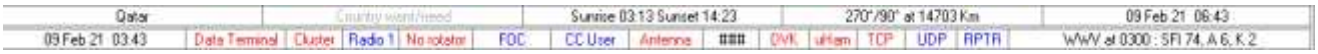
1. Zip databases and Logbook (backup your logbook data).
2. Zip user files (backup Logger32's configuration).
3. DX Spot pane.
4. Logbook entry window (a.k.a. QSO info or log entry pane).
5. Logbook page (your logbook!).
6. Previous QSOs pane.
7. DX cluster pane.
8. Sound card data (for legacy digital modes such as RTTY & PSK).
9. Tracking pane (maps).
10. Worked/confirmed table (breakdown of band+mode slots).
11. Notes pane.
12. Data terminal.
13. CW (CW machine).
14. DVK (**D**igital **V**oice **K**eyer).
15. BandMaps.
16. Snap windows to size.
17. Ham CAP propagation prediction.
18. GridSquare calculator.
19. Contest serial numbers.
20. Conversion between units.
21. NCDXF Beacons.
22. Internet callsign lookup.
23. HamQTH (online callsign lookup).
24. CD ROM lookup (*e.g.* HamCall CD).
25. DXpedition info update.
26. Alpha 87A (amplifier controls).
27. VSC (**V**irtual **S**teppIR **C**ontroller).
28. Preset rotator headings.

[70] Your *radio* may have an out-of-band transmit lockout too, for a belt-n-braces control.

Copyright © 2026 Robert C. Furzer K4CY

The toolbar icons are grayed-out (they lose their color) when the respective functions/panes are active, and regain their color when they are closed.

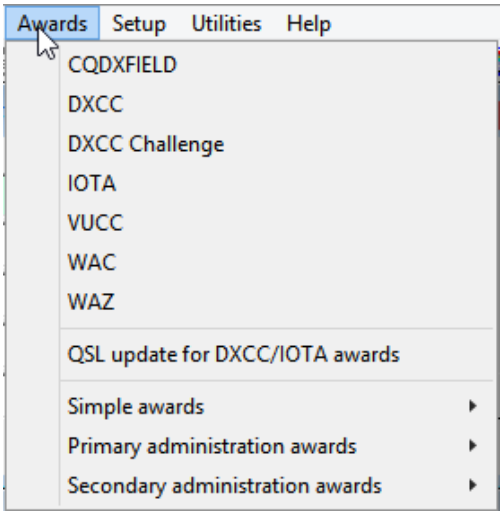
- **Show DX information bar**: five panels near the bottom of Logger32's main screen, just above the status bar, give information about the station you are currently logging ▼



- **Show Status bar**: the very bottom of Logger32's main screen has a row ▲ of fifteen black, red and blue text panels. These are primarily informational (*e.g.* the UTC date and time on the left, and the most recent solar data received from WWV/WCY via the DX cluster network on the right), but most panels can be right-clicked to open menus with which to enable/disable and configure things (such as contest numbering and the UDP BandMap).

Hinson tip: hover the mouse cursor over any of the icons and panels for a moment to display a yellow "tooltip" pop-up message with hints about whatever you have moused-over.

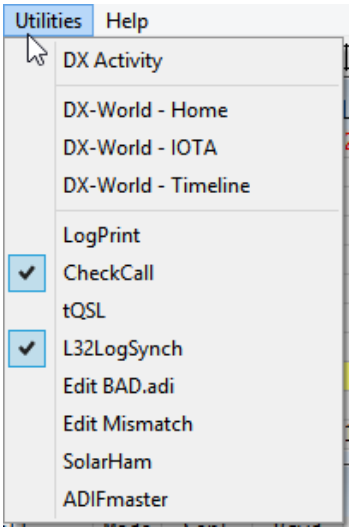
4.2.4 Awards



See the Awards chapter for the low-down on configuring and using Logger32's *extensive* awards tracking and entry submission functions ▶

4.2.5 Setup

See below. This option swaps to another text menu with additional Logger32 configuration options.



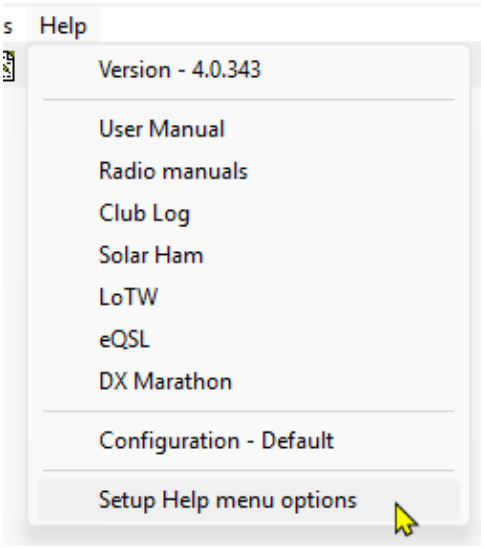
4.2.6 Utilities

Utilities are additional programs you might want to run, either every time you use Logger32 (tick the ones you want to start at the same time as Logger32) or occasionally (unticked: click to run them as and when required, on-demand).

◀ Customize the list of **up to twenty programs or URLs** offered on the bottom section of your menu using **Tools ⇨ Utility program setup**. These are the ones I have chosen: yours will differ.

Copyright © 2026 Robert C. Furzer K4CY

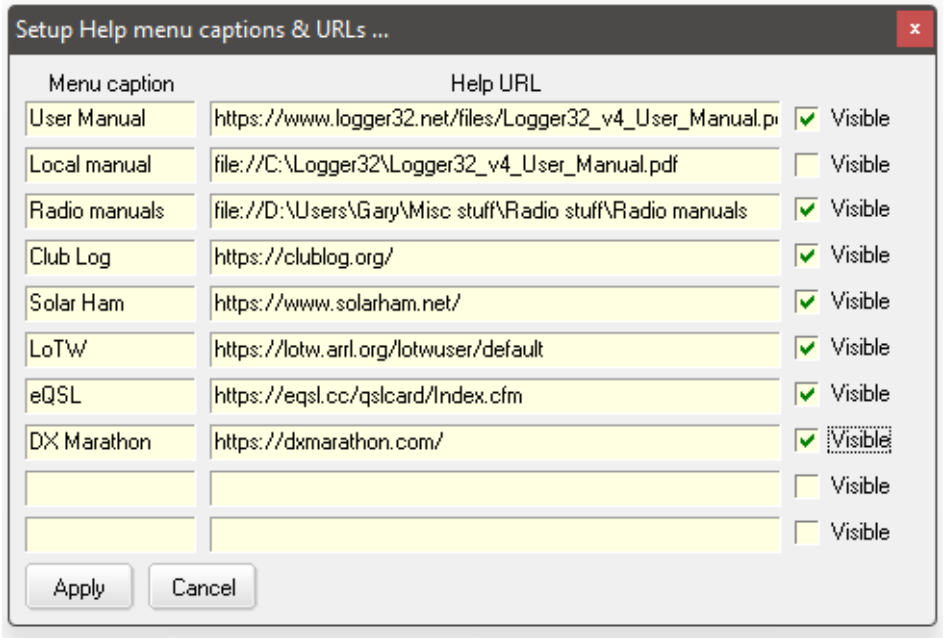
4.2.7 Help



- **Version**: Logger32's version number[71] increments with each program update ▶
- **User Manual**: is a hyperlink (URL) to the latest User Manual – an online *.PDF* Adobe Acrobat file.

> **Hinson tip**: if the version number shown on the front page of the User Manual you are reading is some way behind Logger32's version, click <**User Manual**> on the menu and, if necessary, refresh the cache in your browser using <**Ctrl**+**F5**> to download and open the very latest User Manual. Both Logger32 and the manual are normally updated within a few hours of each other, near the start of most months.

- **Other entries**: I have added stuff to *my* Logger32 Help menu using the menu options at the bottom – more below.
- **Configuration – Default**: by default, Logger32 saves and recalls its configuration settings in the file *C:\Logger32*\Logger32.*INI*. Alternatively, you can launch Logger32 with a command line parameter specifying a different *.INI* file *e.g.* for casual contest logging or testing. If so, the <**Configuration – [name]** > line under <**Help**> names the *.INI* you are currently using[72].
- **Setup Help menu options**: using this function, we can rename the Help menu line that normally shows "User Manual" and/or configure a different URL, and optionally add more lines and associated URLs to the Help menu – up to ten entries in total.



◀ Under <**Menu caption**>, name each entry and tick <**Visible**> to show that line on the menu. Or not.

The Help URLs may point to web pages and other online documents using typical URLs starting with **HTTP://** or **HTTPS://** Alternatively, open files or folders[73] on your local disk using URLs starting with **file://** followed by the disk, folder

[71] The version number is also shown, briefly, on the initial splash screen while Logger32 is launched.

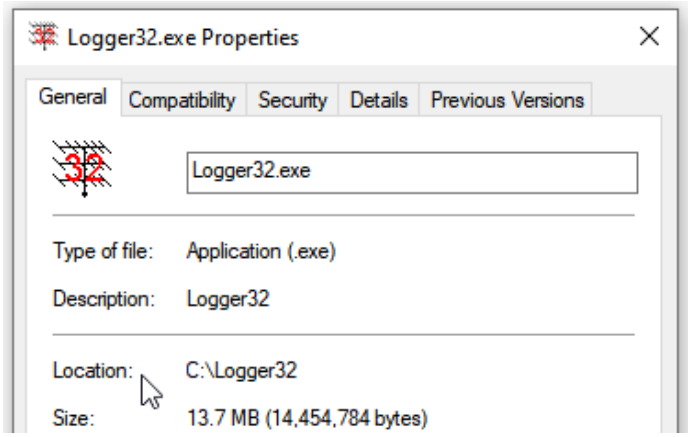
[72] Changing configurations involves closing and then re-starting Logger32 to read and load the selected *.INI* file. If you change various options and settings while Logger32 is running, those changes are saved in the current .INI file/s for the current configuration only, leaving others as they were.

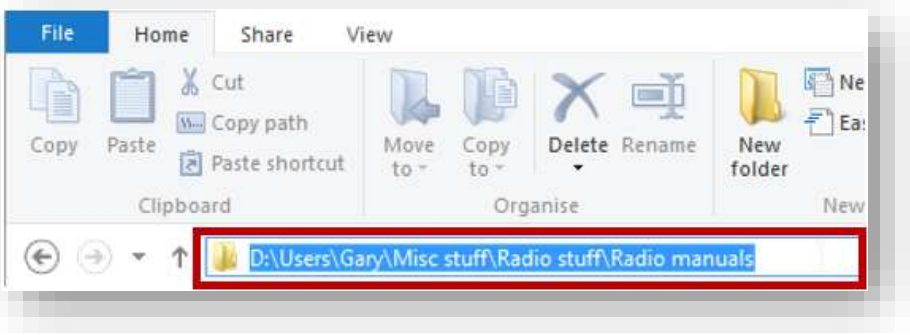
[73] You might like to store local copies of all your radio manuals in one folder for ready reference.

Copyright © 2026 Robert C. Furzer K4CY

and filename in the conventional Windows backslash format including any embedded spaces – for example, on my system, the entry *file://C:\Logger32\Logger32_v4_User_Manual.pdf* opens a local copy of this manual that I have previously downloaded to the *C:\Logger32* folder on my PC.

Hinson tip: in File **E**xplorer (opened using <**Win+E**>, right-click any folder or file then click <**Properties**>. The Location*:* field shows you the disk and folder name in the Windows backslash style ▶





◀ You can also click in the path area near the top of the File **E**xplorer screen to display the current folder name in the backslash style. <**Ctrl**+**C**> copies it to the Windows clipboard, ready to <**Ctrl**+**V**> paste it into your Help menu.

4.3 Setup menu

Clicking <**Setup**> on Logger32's main menu swaps the entire menu for a different one giving you *fifteen* further options ▼



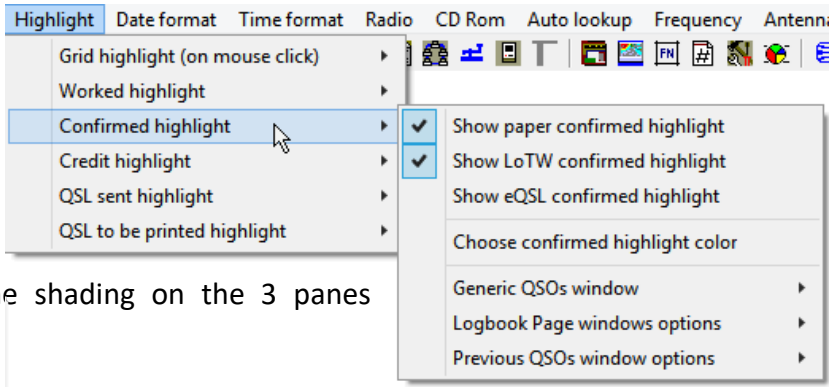
4.3.1 Highlight

'Highlighting' refers to the colored background shading of specific QSO rows in your logbook and text in other 'grids' (tabular reports):

- **Grid highlight (on mouse click)**: when you click a QSO line in your logbook or the previous QSOs pane to check or edit the information, it is handy if the clicked line is clearly identified. This option lets you color both the foreground (*i.e*. the text) and background (the shading).
- **Worked highlight**: this shades the **W** (**W**orked … but not yet confirmed) blobs on your award reports. Choose a fairly light pastel shade for contrast against the dark text.

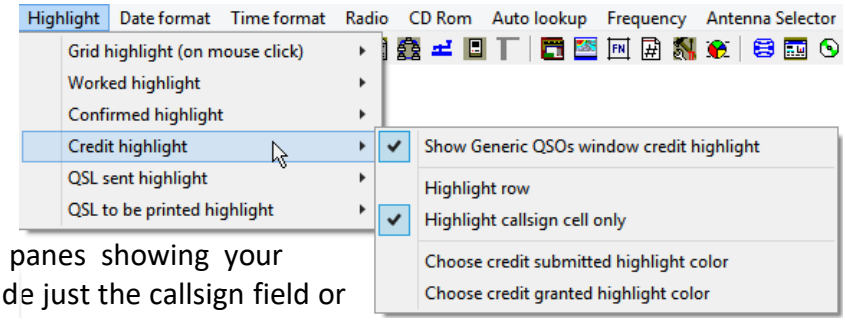
Copyright © 2026 Robert C. Furzer K4CY

- **Confirmed highlight**: there are four options and three submenus here ▶



  The shading applies to QSOs that have been confirmed on paper QSL cards, on LoTW and/or on eQSL, and you can choose whether to show the shading on the 3 panes specified.

- **Credit highlight**: this shades the Submitted and Granted blobs littering your award reports ▶



  Again, choose distinctive but light pastel shades to contrast the dark text. For panes showing your QSOs, decide whether to shade just the callsign field or the entire QSO row.

> **Hinson tip**: having changed between callsign field or whole row colors, the current display is not re-colored until it is re-drawn. One way to trigger a re-draw and re-color is by sliding the slider on the right of the display … or simply wait until you log another QSO.
>
> **Hinson tip**: coloring only the callsign field leaves the remaining fields colored with the zebra stripes, making it easier to read along each QSO line without straying into the adjacent line.

- **QSL sent highlight**: use this to show which QSOs you have QSLd.

> **Hinson tip**: shading QSOs in my log that I have signed-and-uploaded to LoTW is a useful reminder to do so, since for various reasons I choose to sign-and-upload QSOs in batches about once per day rather than individually as they are logged.

- **QSL to be printed highlight**: shade QSOs that you intend to confirm on QSL cards.

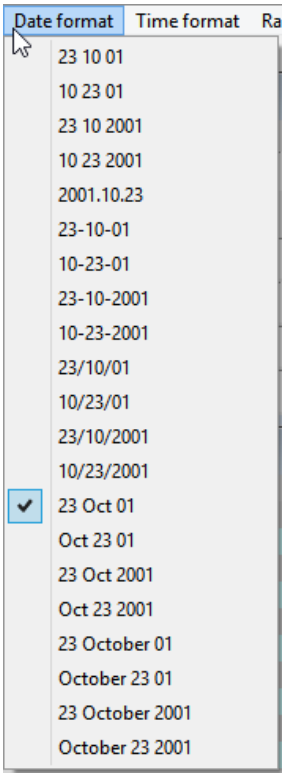
> **Hinson tip**: with so many choices, focus on your priorities, your main interests and objectives. What matters most to you? Get that highlighting setup first with distinctive colors, then work your way systematically down the priority list with subtler shades. Take your time to get familiar with and fine-tune the color coding, then live with it!

Il vaut mieux un qui sait, que 10 qui cherchent !!

F6ITD

Copyright © 2026 Robert C. Furzer K4CY

4.3.2 Date|Time format

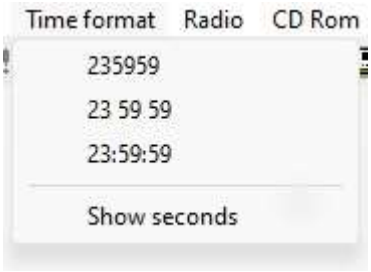


Logger32 records and handles dates and times internally in UTC. It automatically determines the current date and time in UTC from your system clock, taking account of your time zone including daylight savings time where appropriate.

It always outputs them (*e.g.* if you export your log or generate a file of QSOs to send to LoTW) in UTC as specified in ADIF.

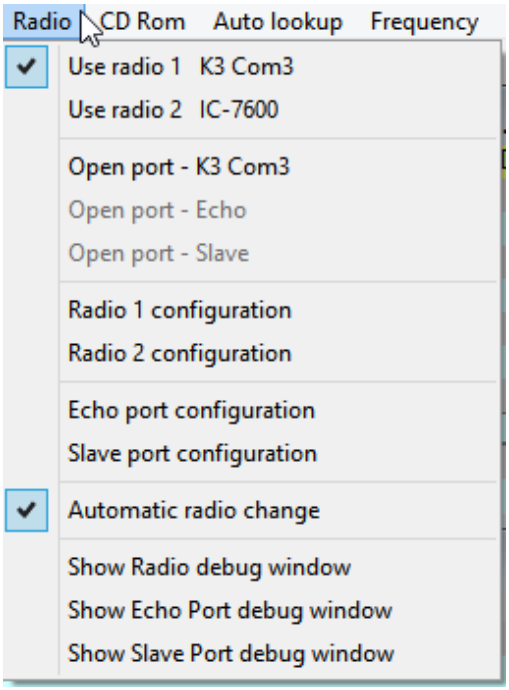
However, just for your convenience, it can *display* them in various formats as you wish.

Take your pick from the date and time formats on offer ▼ ▶



Choose whether or not to display[74] the seconds part in the logbook and previous QSOs ▶

4.3.3 Radio



◀ Connecting the radio to the computer paves the way for Logger32 to monitor and control the radio, supplementing our manual operating skills.

The **Setup** ⇨ **Radio** menu is where we tell Logger what type of radio we have (which affects the format of the commands it needs to poll and change radio settings) and which communications port the radio is connected to, at what data speed *etc*.

Logger32 can monitor and control two radios simultaneously, for example an HF and VHF/UHF radio, or two HF radios for SO2R.

4.3.4 CD Rom

You can set up callsign lookups from callsign databases supplied on CD-ROM (*e.g.* HamCall). Most of us are using online callsign lookups but in case your Internet connection is nonexistent, dead, unreliable, slow or expensive, Logger32 still supports CD-ROM callbooks.

[74] Seconds are always recorded but need not be shown if you find the fine detail distracting.

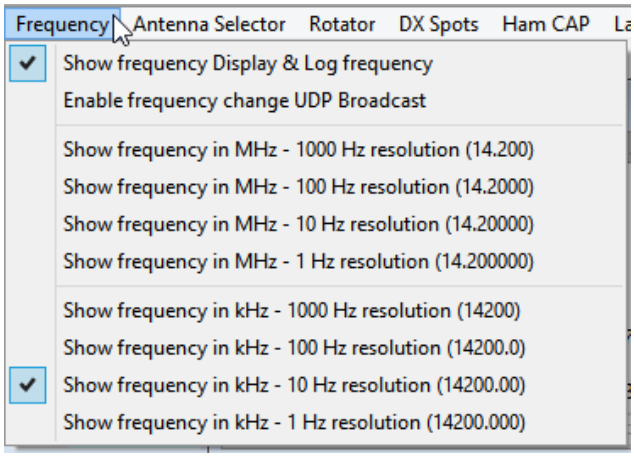
Copyright © 2026 Robert C. Furzer K4CY

4.3.5 Auto lookup

As you are logging a QSO, Logger32 can automatically lookup the callsign in a callsign database containing information such as the operator's name and address, QSL route, grid square *etc*. This menu is where you tell Logger32 which database to use.

4.3.6 Frequency

In decades past, hams generally logged QSOs by band, not by frequency, because it was quicker and easier that way. It was good enough.



◀ These days, our CAT-connected radios report their VFO frequencies automatically, so logging them is simply a matter of ticking the first option on the menu.

<**Enable frequency change UDP Broadcast**> lets Logger32 inform *other* software that your radio has QSY'd via UDP messaging on the shack LAN.

Choose between the remaining options to log and display VFO frequencies in kilo or megahertz, with 1, 10, 100 or 1,000 Hz resolution.

Hinson tip: if your activity is primarily on the low bands, you probably want one of the lower options on this list. Go for the upper options if you are into VHF/UHF/microwaves.

4.3.7 Antenna Selector

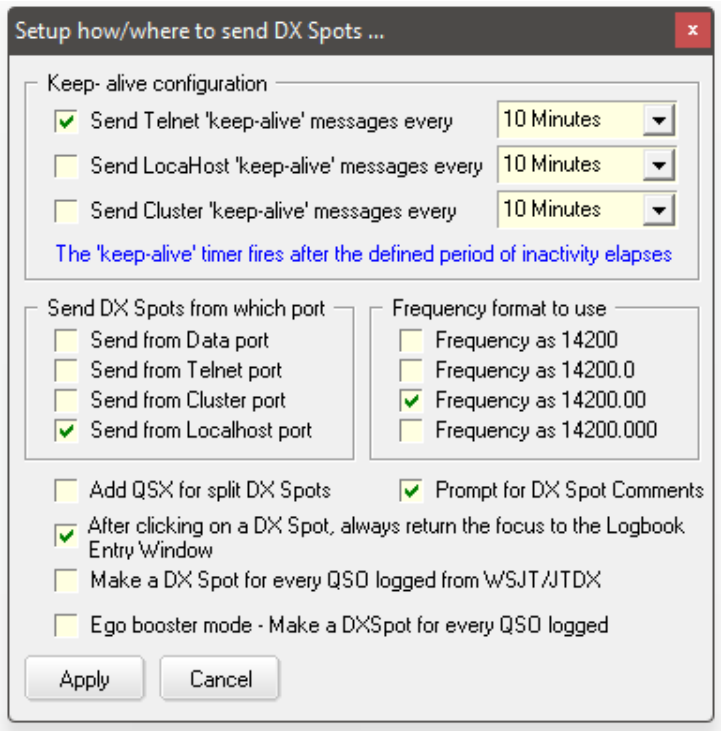
Configure an automatic antenna switch using ports on your PC.

4.3.8 Rotator

Configure up to nine digitally-controlled antenna rotators here.

4.3.9 DX spots

Send regular keepalive messages to your DX cluster connections, and determine which connection (port) is used to submit DX spots ▶

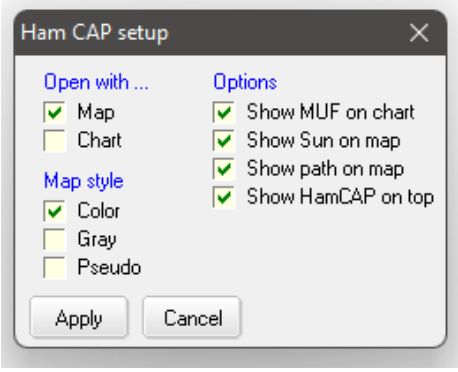


This is where you can opt to type and send comments with your DX spots, and (if you insist) spot *all* your QSOs as you log them ('ego booster mode' ☺).

Hinson tip: once tabs in the DX Cluster pane are connected, they take on the names of the connected nodes instead of showing their default names. If, like me, you forget which one is which, look at the boilerplate text in the command fields on each tab for clues. Or disconnect.

Copyright © 2026 Robert C. Furzer K4CY

4.3.10 Ham CAP

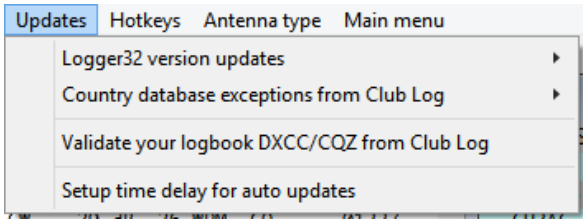


Configure the Ham CAP propagation prediction software ▶

4.3.11 Language

Shows a Windows message about selecting translated DLLs in order to display non-English menu options and messages in Logger32.

4.3.12 Updates

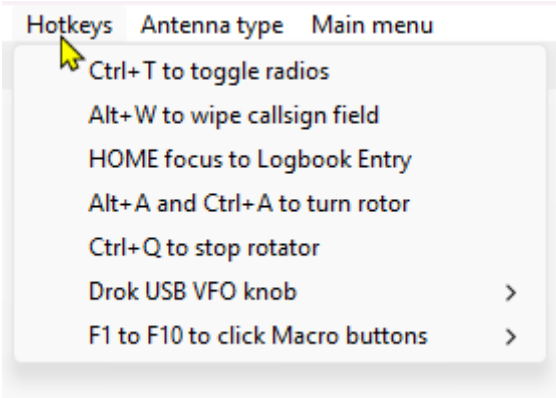


◀ Check for updates to Logger32 and its country database (which affects the determination of DXCC entities when you log future QSOs).

Optionally, validate country and zone allocations for QSOs already in your log, and setup automatic Logger32 version checks and updates shortly after Logger32 starts up.

4.3.13 [Global] Hotkeys

[See chapter 24 for more on hotkeys.]



Enable/disable *global* hotkeys (keyboard shortcuts) here ▶

Any hotkeys you tick on this menu are assigned to Logger32 'globally', meaning they trigger things in Logger32 even while other programs are in the foreground, in focus, being actively used.

Hinson tip: if you use any of these hotkeys routinely in your email, browser, office or other software (such as <**Alt**+**W**> to **W**ipe the QSO currently in the N1MM+ log entry pane), *don't* tick to assign them globally to Logger32! Whether ticked or not, you can still use the defined keys *while* Logger32 is in the foreground and has the focus. Any ticks here mean those hotkeys *also* work even when Logger32 is in the background.

- **Ctrl+T to toggle radios**: makes it easy to swap to the other radio when using SO2R.
- **Alt+W to wipe callsign field**: is one of several ways to clear all data (not just the callsign field) from the log entry pane.
- **HOME focus to Logbook Entry**: with the focus on the logbook (*e.g.* after checking or editing a QSO entry), the pressing <**Home**> key normally takes the cursor and view to the first QSO in your log, in the same manner as the <**End**> key takes you to the last (most recent) QSO. If you tick this hotkey menu option, however, <**Home**> jumps the cursor directly to the log entry pane, ready to enter the next QSO.
- **Alt+A and Ctrl+A to turn the rotor and Ctrl+Q to stop rotator**: if Logger32 has digital control of your rotator/s, you can turn the **A**ntenna defined for the current band clockwise using <**Alt**+**A**> or anticlockwise with <**Ctrl**+**A**>, or **Q**uit (<**Ctrl**+**Q**>) to stop it turning either way.

Copyright © 2026 Robert C. Furzer K4CY

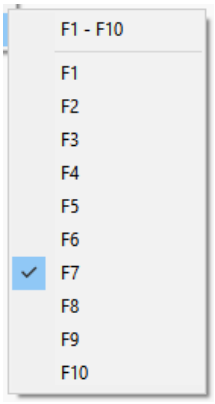
Hinson tip: the hotkeys are hard-coded in Logger32. If for some reason you *need* to change them, try a keyboard redirector utility. Or disable them. Or put up with them.

- <**Drok USB VFO knob**> opens a submenu ▶



  - **Main Radio**: the Drok knob controls the VFO of the main radio – if configured correctly.
  - **SO2R/SO2V**: the Drok knob controls whichever radio and VFO is currently active.
  - **Lock Drok**: stops the Drok device controlling anything.

- <**F1 to F10 to click Macro buttons**> also opens a submenu ▶

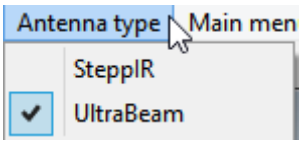


  - **F1 – F10**: tick this to give Logger32 full global control of *all ten* PC function keys, running the corresponding macros regardless of whichever programs are in the foreground at the time (provided Logger32 is actually running on the PC, that is: if not, curiously enough, it doesn't react if you click its function keys or hotkeys …).
  - **F1** to **F10:** tick any individual PC function keys that should trigger Logger32 macros even when other programs are active. Function keys that are *not* ticked here will work as normal in other programs.

Hinson tip: if you only use a few specific function keys in other programs, you *can* assign unused keys to trigger Logger32's macros but it's a good idea to label or mark the keys on your keyboard as a reminder. Be sure to configure the corresponding macros, otherwise the global function keys will appear to be broken, lifeless, ex-function keys.

4.3.14 Antenna type

Tell Logger32 whether you have an UltraBeam or a SteppIR which affects the commands used to control it ▶



4.3.15 Main menu

Revert to the main menu from the setup menu[75].

4.4 The bottom panels

Along the lower edge of Logger32's main screen are two horizontal rows of information/control panels. The upper one is the DX information bar ▼



▲ The lower one is the Status bar.

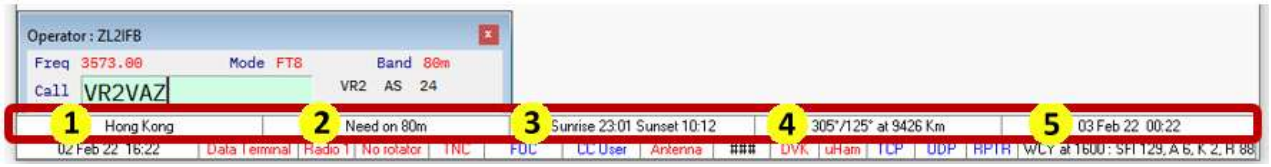
If either or both are not shown, tick to enable them under the <**View**> menu.

[75] Logger32's **Main ⇄ Setup** menu toggle is a novel workaround since the window is too narrow to show all the configuration options on one enormously wide menu!

Copyright © 2026 Robert C. Furzer K4CY

4.4.1 The DX information bar

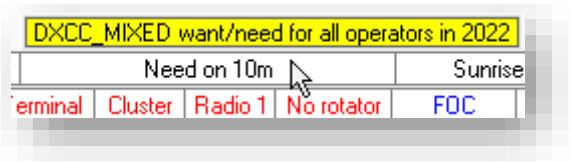
While the Call field of the log entry pane is empty, the DX info bar's five panels tell you, in light gray 'boilerplate' text as shown above, what information they will show when there is a callsign present to look up ▼



These are the five DX info bar panels:

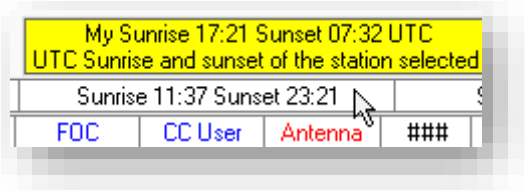
1. **DXCC entity name**: having looked-up the prefix or locational modifier for the DX callsign, Logger32 determines the DXCC entity from which the station is *believed* to be operating. It takes account of known exceptions based on information from Club Log (if you use the facility), such as those rare E5 stations known to be based in the North Cook Islands rather than the more common South Cook Islands.
2. **DXCC wanted/needed status**: Logger32 tells you if you *need* a QSO with the DX station's DXCC because you have *never* worked it, or you have worked it but not on the present band and/or mode, at any time or (*if* the top left corner of the worked/confirmed table shows the year) not yet this UTC year.

   Mouseover panel two for a yellow popup with details ▶



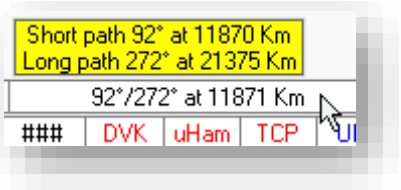
3. **DX sunrise and sunset times**: Logger32 *estimates*[76] and displays the dawn and dusk times in UTC for the DX station.

   Mouseover panel three to see *your own* dawn and dusk times (also in UTC) ▶



   If either of your dawn and dusk periods overlap with either of the DX station's, you may be able to take advantage of enhanced low-band propagation along the gray line.
4. **Bearings and distance to the DX**: to help you turn your beams, Logger32 calculates and displays the azimuth directions (with **true North** at 0˚) for the great circle short and long paths from your station QTH to the DX station, and the short path distance.

   Mouseover panel four to see the long path distance as well ▶



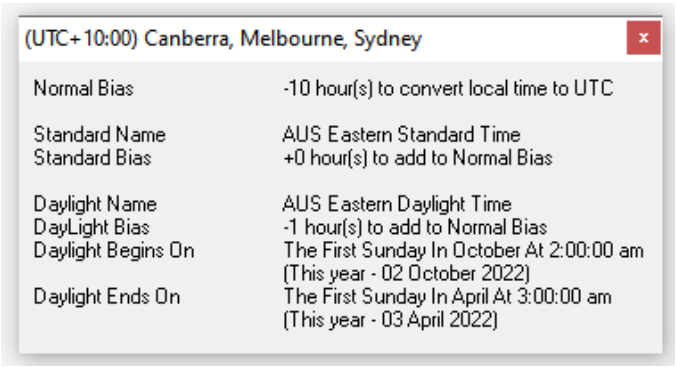
[76] To calculate the *precise* sunrise and sunset times, Logger32 would need to know the *precise* location of the DX station. If the best locational information it has is the station's DXCC entity, that may span several time zones for a large country, the approximate center of the entity is used for estimation purposes.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: distances are shown in miles or kilometers, whichever you have configured. The values are calculated geometrically using your station location and the DX station's probable location – both of which are subject to error, hence they may not be accurate. The actual path taken by radio signals between you may meander and bounce between the Earth's surface and the ionosphere or other reflective surfaces such as meteor trails and aircraft. The DX station may not be in the precise center of the DXCC entity or grid square (if known). In short, YMMV. Literally.

5. **Local time for the DX station**: this is *probably* his 'clock time', taking account of daylight saving/summer time adjustments.

Right-click panel five to read all about the DX station's time zone ▶



If you are contacting him in his evening, anticipate a fairly relaxed easy-going QSO. During his morning, he may be in a hurry to get to work.



4.4.2 The Status bar

◀ The "Status bar" at the very bottom of Logger32's main screen is *not* in fact a cool place for fashion-conscious egotists to hang out in their fancy finery, sipping expensive champagne cocktails and speaking *far* too loudly.

The Status bar shows the status of numerous program functions, and (for most) gives you the opportunity to right-click and configure them.



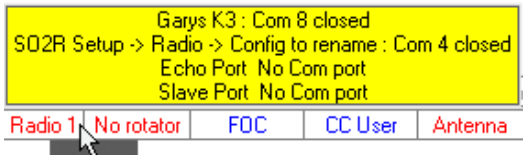
▲ From left to right, these are the Status bar's fifteen color-coded panels:

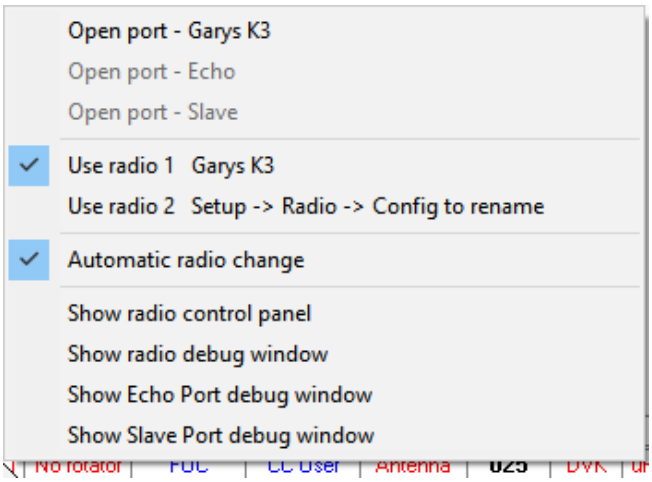
1. **UTC date and time**: unlike local/clock times and dates, UTC is a precise, globally consistent time and date reference based on atomic clocks. Hams who log their QSO dates and times in UTC are *far* more likely to have them confirmed by the counterparties than those who use some other reference, particularly if they neglect to state the time system or zone used, or write dates ambiguously without clearly denoting which are the day and month values.
2. **Data Terminal** status: like most of the Status bar panels, the panel text is in red if the Data terminal function is not presently running. If you have a TNC connected to a serial port and the Data terminal is running, the text goes blue. Right-clicking the panel opens a little menu of options for configuring the TNC – presumably (I no longer have a TNC so I can't check this, having rashly sold mine … way back in the 1980s).

Copyright © 2026 Robert C. Furzer K4CY

3. **Radio #** (CAT connection) status: the name of the currently selected radio is shown in this panel, in blue text if the CAT port is open and the radio is connected, otherwise in red.

   The mouseover yellow popup also shows the name of the *other* radio if you have an SO2R setup, and the status of the echo and slave ports ▶

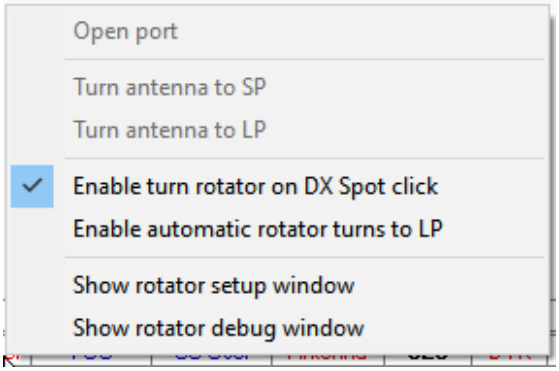




◀ Right-click the Radio panel for a menu to open|close the CAT connection/s, choose which SO2R radio is active (or enable automatic change-overs), and display the **R**adio **C**ontrol **P**anel and debug windows.

4. **Rotator** status: you know the score. Blue text tells you your digitally-controlled rotator's COM port is configured and open, otherwise the panel reads No rotator in red.

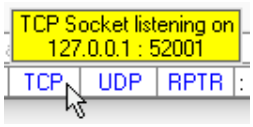
   Right-click the rotator panel to configure, open and control the rotator nominated for the current band in your Bands & Modes table ▶

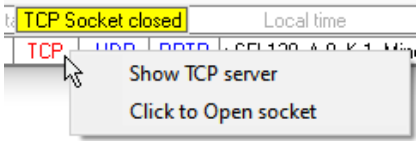


5. DX Cluster **TNC** connection status: if you connect to the DX Cluster network using an old-school TNC and VHF/UHF "packet radio", by the color of the text, this panel shows you whether the TNC tab in the DX Cluster pane and its port are open for business, or not.
6. DX Cluster **Telnet** connection status: if you connect to the DX Cluster network through the Internet using Telnet (as most DXers do), this panel shows you what's going on. Configure the DX Cluster's IP address and Telnet port through the DX Cluster pane. The text in this panel is the name of the corresponding tab in the DX Cluster pane. A yellow mouseover popup tells you if the auto-reconnect and keepalive functions are active on that tab.
7. DX Cluster **localhost** connection status: likewise, this panel has the name of the 'localhost' tab on the DX Cluster pane in red if it is presently disconnected or blue if connected. Right-click to connect or disconnect it. Point the mouse at the panel to check the autoconnect and keepalive status.
8. **Antenna** switch status: find out what antenna is currently selected, according to the current band. Mouseover the panel for more information. Right-click it then click to show the antenna selector form through which you can check and reconfigure your antenna switch.
9. **Contest serial number**: *if* contest serial numbering is presently enabled, this panel shows the serial number for the present QSO if you are currently logging someone, or the *next* QSO if not. When enabled, the number is automatically logged with each QSO in the ADIF STX field. While serial numbering is disabled, the panel shows **###**, meaning "Num num num".

Copyright © 2026 Robert C. Furzer K4CY

10. **D**igital **V**oice **K**eyer status: is your DVK connected and running or not? Find out here.
11. **uHAM** status: is your *micro*HAM unit connected and running or not?
12. **TCP** server status: the blue text and mouseover popup yellow tooltip (with the IP address and TCP port number) show that the TCP port is configured and open, listening obediently for any ADIF QSO records sent by other logging software via TCP to this PC and port ▶

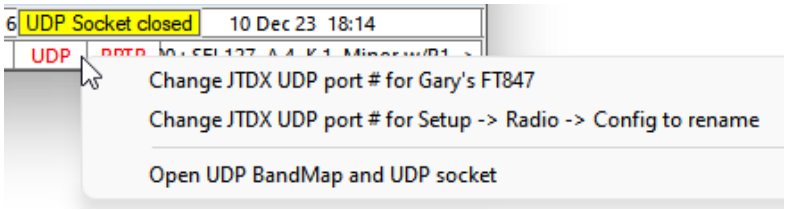




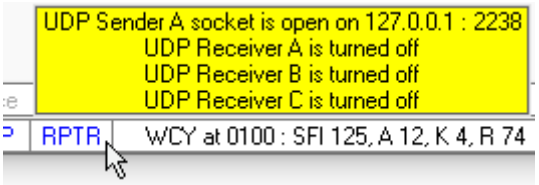
◀ Right-click the panel then click to show the TCP server configuration form or close the TCP socket, turning "TCP" red.

13. **UDP** port/connection and BandMap status: UDP means Logger32's UDP port is closed, whereas UDP means the port is open, waiting for any ADIF QSO records sent by other logging software (such as JTDX) via UDP to this PC's IP address and UDP port number.

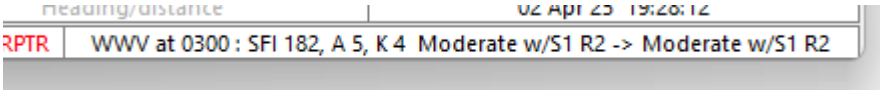
    Right-click the panel to change port numbers or open/close the UDP BandMap and UDP socket (port) ▶



14. **RPTR** port status: check whether Logger32's UDP **R**e**P**ea**T**e**R** ports are ready to send and receive QSO data or not. Mouseover the panel for more info ▶



15. **Solar & geomagnetic data**: the larger rightmost panel displays the solar and geomagnetic data most recently received from WWV/WCY via the DX cluster network (if any) ▼



4.5 Getting started FAQs

Q. Do I *really* need to read the whole of this manual?

A. No, not really. Logger32's basic functions are reasonably intuitive and self-explanatory. Have fun exploring the menus and right-click options.

Before you go though, you might like to bookmark this page to make it easier to come back here if (or rather when!) you get confused or stuck, and to continue reading about the program's myriad cool features.

Don't forget that you can click on the contents listing or other underlined links to go directly to sections of interest, while <**Ctrl**+**F**> makes it dead easy to **F**ind things in this document.

Hinson tip: despite having used Logger32 heavily for well over a decade, and having compiled and maintained this User Manual, I *still* refer to and search it when I forget stuff.

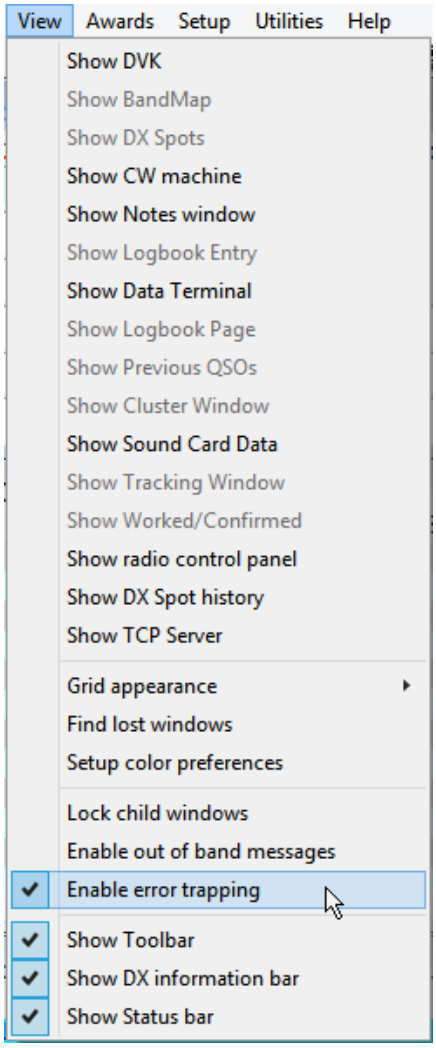
Copyright © 2026 Robert C. Furzer K4CY

Q. The error messages from Logger32 are annoying. I'm not interested in errors and bugs. I simply want to log my QSOs. What can I do?

A. We want you to log QSOs too but it's not simple to provide a logging program with so many fantastic features requested by users over many years, using such a diverse range of computers, radios and modes. There's a reason this User Manual is so huge!

Designing, developing, testing and maintaining the program is an ongoing challenge, making sure it works correctly, does what it's supposed to do for everyone while protecting our valuable logs against damage or loss in the process.

You can cut down Logger32's informational error messages by *un*ticking <**Enable error trapping**> on the <**View**> menu ▶



That won't stop Logger32 detecting and showing error messages for more serious conditions – show-stoppers that prevent the program operating correctly – but hopefully they are so rare that you won't see any of them, thanks to the error trapping, reporting and fixing that is still being done routinely by other hams on your behalf …

Q. Why oh why does Logger32 take s o l o n g to start up?

A. Logger32 is busy doing all this:

- It opens and reads the *.INI* file (using *Logger32.INI* by default, unless Logger32 is launched with a command line parameter specifying a different configuration).
- It sets itself up by applying various configuration settings specified in the *.INI* file *e.g.* which windows and functions to open where and at what size, what columns to show in your logbook and other panes, what highlighting and alerting to enable …
- It opens and sets up its internal databases, statistics *etc*.
- It opens and reads in the current log (as specified in the *.INI* file for the current configuration), displaying recent QSOs in the logbook pane if shown.
- It opens communications ports such as the CAT port/s for your radio/s, and starts communicating through them.
- It launches any utility programs that have been ticked to auto-start.
- If the cluster log-on option is ticked, it logs you on to the DX cluster/s, grabbing, processing and displaying recent DX spots on the DX cluster and DX spots windows and BandMaps if shown. This may take a while if there are lots of spots to process and show.

Copyright © 2026 Robert C. Furzer K4CY

- If you had been running JTDX|WSJT-X when Logger32 was last closed (having launched it from the UDP BandMap **Start** menu), it re-opens the same digimode software.

If you are keen to get going, you can start chasing DX spots, changing bands or modes on the rig, and logging QSOs at this point, while Logger32 finishes winding up the rubber bands as the final activity happens in the background …

- If so configured, after a short delay, it checks for new versions of Logger32 and/or updates to the DXCC prefix allocations and exceptions from Club Log. If found, it downloads and applies them.

Hinson tip: occasionally, you may find that clicking menu options does nothing, as if you hadn't clicked at all. By all means click again, harder if you like, pointing the mouse cursor precisely and deliberately at the very option you want to action. Try double-clicking or right-clicking maybe to see if anything happens, or click something outside of Logger32. If your clicks are seemingly being ignored, your PC may simply be too busy with some other activity to process the mouse clicks at that point … so wait a moment and try again. Conceivably, if it continues rudely ignoring you, your wireless mouse may need feeding (change the battery or recharge it), disconnecting-and-reconnecting, cleaning (polish its balls or dust off its sensors) or updating/reloading the mouse driver. The wireless receiver dongle may be suffering QRM from your transmitter or other wireless devices in the area – so try moving it nearer the mouse pad and perhaps swap to a different USB port. Rebooting the PC resolves *most* software issues. Physical replacement may be the last remaining option if you have worn out the mousy hardware by excessive pointing-and-clicking, or it has been savaged by your CAT. Borrow another mouse or entice one out of its hole in your IT junk pile to see if that magically resolves all your problems. *Good luck Jim.*

Copyright © 2026 Robert C. Furzer K4CY

5 Country, county and island databases

> "Data is the key
> to understanding everything
> that's happening in the world"
>
> *Nate Silver*

The primary administration (DXCC entity/country and prefix), secondary administration (*e.g.* US county) and other (*e.g.* IOTA) databases can all be maintained using functions built-in to Logger32. Automated updates using DXCC information from Club Log make it easy to keep up with changes and DX operations with ambiguous prefixes such as E51D from *North* Cook Islands in 2023.

5.1 Country and prefix database maintenance

The primary administration database contains pertinent details from the current ARRL DXCC list. For each DXCC entity, the database records:

- The prefix designated by ARRL (this is a key field: values must be unique).
- The entity name, as given on the official DXCC list.
- The latitude and longitude of the approximate center of the DXCC entity.
- The CQ zone/s.
- A flag if the entity has been deleted from the DXCC award.

As these details change from time to time (*e.g.* "Swaziland" became "Kingdom of eSwatini"), you may want to update the database originally supplied with Logger32 … in a way that does not prevent it being used to identify DXCC entities, calculate rough bearings *etc*.

The primary administration database includes the primary administrative subdivisions laid out in the ADIF standard – US states for example, plus the District of Columbia.

5.1.1 Determining a DXCC entity from a callsign

This is the process that Logger32 follows systematically in attempting to identify the DXCC entity from a callsign entered into the Call field of a log entry pane:

1. **Strip the callsign down to its essential parts.** Does the callsign appear in the alternative prefixes database? If it does, then the first part is complete. This is done so that for certain callsigns (for example, SV8ASP/A in Mount Athos, rather than the normal /A meaning of operating from an alternate QTH), an alternative prefix or exceptional callsign can be defined. This check would find a perfect match and stop checking. Otherwise, it would proceed by ignoring the suffix (*e.g.* /M, /P, /A, /MM, /AG, /AA, /AE). These suffixes have nothing to do with the entity which was already determined. Determine which piece of a portable callsign is the portable piece. Is KI0K/VP2M in the US, or Montserrat? The rule applied is: the shortest piece (KI0K or VP2M), or if they are the same length the first part (following ITU convention),

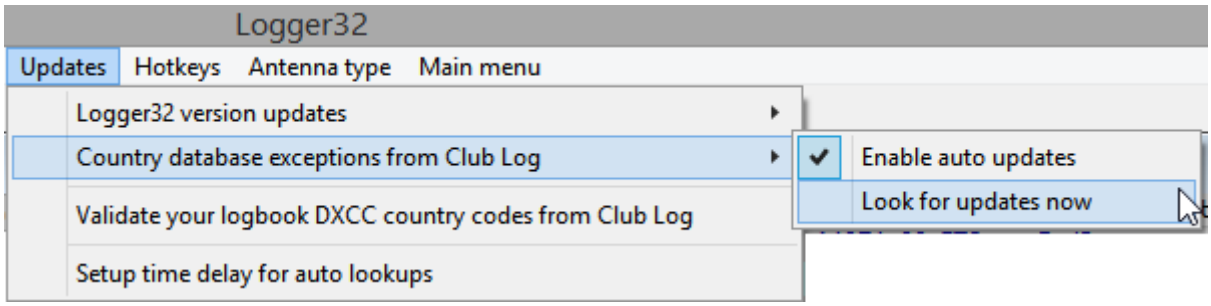
Copyright © 2026 Robert C. Furzer K4CY

specifies the entity. With KI0K/VP2M the callsign is (internally) treated as VP2M/KI0K. A station using or being logged with the callsign "N6BFM/9K2" is treated (internally) as if it were 9K2/N6BFM

2. **Identify the entity prefix.** Callsigns with just a number following the slash are treated as if there was no slash+number *e.g.* N6BFM/4 is equivalent to N4BFM. Now reduce the length of the callsign by one character at a time, repeating as necessary, to find a known prefix *e.g.*
   - N6BFM → N6BF → N6B → N6 → N *STOP!* N is an alternative prefix for W (USA).
   - SV2ASP/A: the first pass of the loop finds that SV2ASP/A is an alias for SY - Mt. Athos.
   - KI0K/VP2M is equivalent to VP2M/KI0K. Looping, that becomes VP2M/KI0, then VP2M/, then VP2M which matches with Montserrat.
3. **If there is any offset information on the entity, determine the exact offset.** Check if the callsign is in the large country prefix offset database. N6BFM does not sign as portable 4 so you could, if you choose, put N6BFM into the offset database (under W for USA) with the details of zone, latitude and longitude of Atlanta. If a match is found, this step is complete. Check if the prefix of the callsign (not the official entity prefix) is in the offset database, for example CK1XX in Canada. Under the offsets for Canada is there a CK1? If so, the offset is found, and this step is complete. Take the country prefix – for example EX0MM that has the country prefix of UA9. If the number in the callsign differs from the number in the prefix, as it does in this case, strip the number from the country prefix (we now, temporarily, have UA as the country prefix), and add the suffix of the callsign (we now have UA0MM). We now loop, decrementing the length of the callsign each loop (first pass UA0MM, second pass UA0M *etc*.) and check for a match in the Offset database. Coincidentally, there is a UA0M in the Offset database with CQ Zone 16 (the default CQ zone for Asiatic Russia is Zone 19). You can see that by using this method, all stations in Asiatic Russia with the Number 0 and the first letter of the suffix as M (UA0Mxx, RA0Mxx, EX0Mxx, RX0Mxx, *etc*.) can be identified as being in Zone 16. Easy, huh? If an offset is found to match during the third step of the process, the CQ zone, latitude and longitude of the offset are used in the calculations.

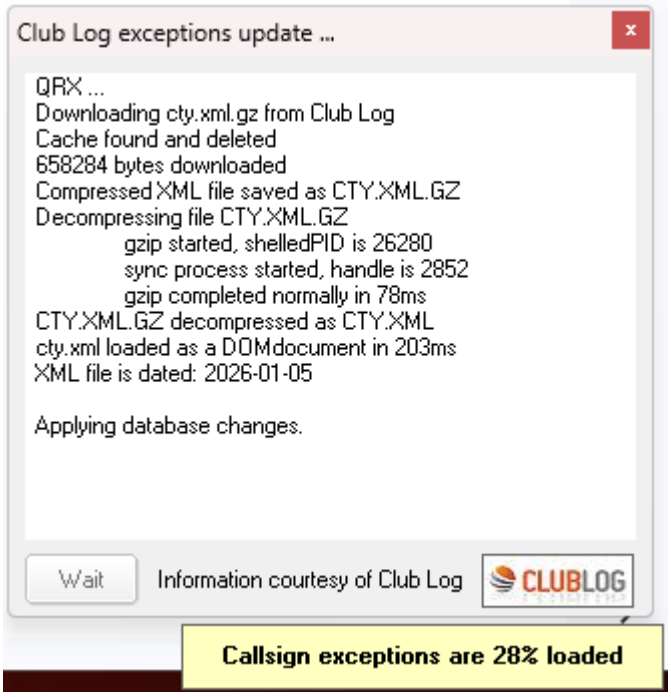
5.1.2 Updating country database/prefix exceptions from Club Log

Hinson tip: by far the easiest and safest way of maintaining your DXCC database in Logger32 is to use Club Log ▼ Club Log's information is actively maintained and highly accurate with very few errors/anomalies, at least compared to *my* efforts. Maybe *you* are angelic?



Club Log keeps constantly up to date with any changes to the ARRL DXCC list, such as new or deleted countries and prefixes, country name changes, active DXpeditions *etc*. Changes are made to Club Log's data within *hours* of their announcements or on the effective date (if later).

Copyright © 2026 Robert C. Furzer K4CY



◀ **Setup ⇨ Updates ⇨ Country database exceptions from Club Log** gives you the option of automatic updates: soon after it starts up, Logger32 checks for any updates from Club Log, downloading and applying them if found.

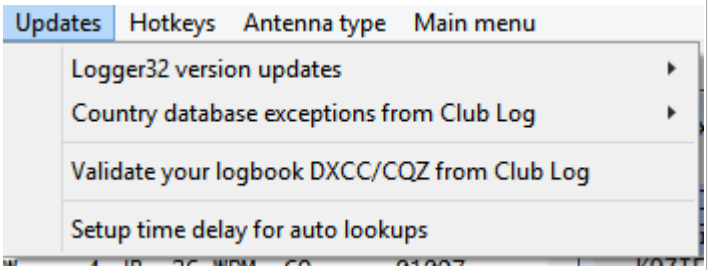
You can also trigger updates on demand to check and update right now.

This facility is extremely useful for identifying the correct DXCC entity for DX stations with ambiguous prefixes (*e.g.* North and South Cooks both use the E51 prefix, while many French overseas territories share the TO prefix). More than a few US stations continue using their US island callsigns while operating on the mainland: it's not unusual these days to work a KP4 station in Florida, or a KH2 or WH6 in California.

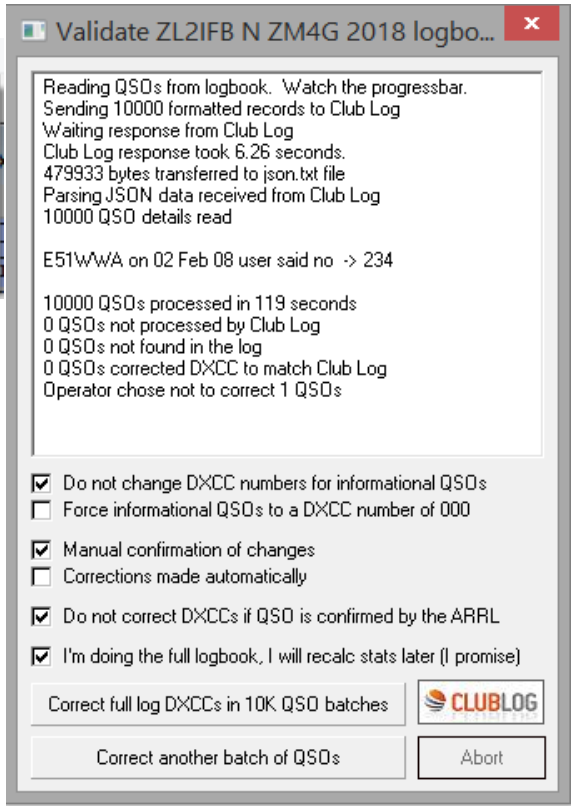
Hinson tip: these country database exception updates are ***not*** retroactive. They only affect lookups for *future* QSOs when you are logging them. QSOs *already* logged in your logbook are not touched. DXCC entities recorded in your logbook remain as they are … *unless* you choose to validate and edit those individual QSOs for some reason, using a separate process.

5.1.3 Validate DXCC entity codes and CQ zones using Club Log

As well as updating your databases, Club Log can systematically check QSOs already in your log to find any that *appear* to have the wrong DXCC entity code or CQ zone allocations.



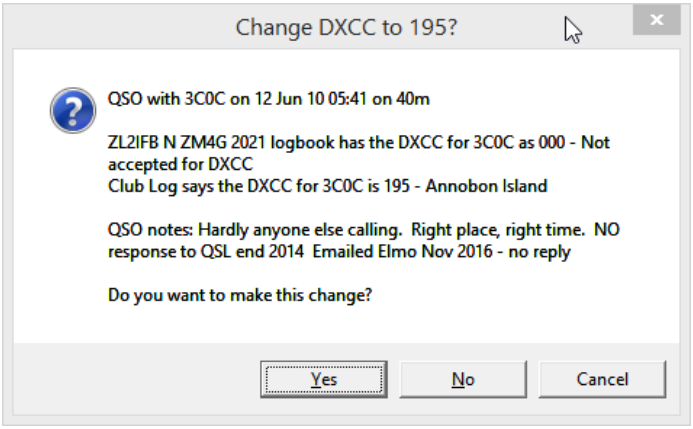
▲ Run the checker using **Setup ⇨ Updates ⇨ Validate your logbook DXCC/CQZ from Club Log**



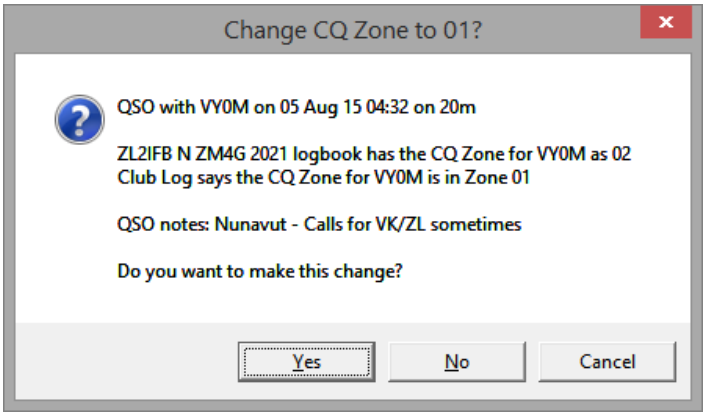
Informational logbook entries can be skipped or set to DXCC entity number 000 ('Not accepted for DXCC') ▶

Copyright © 2026 Robert C. Furzer K4CY

If you are brave/lazy, you can let Logger32 make all the changes *automatically* as they are suggested by Club Log. The safer *manual* but slightly tedious option gives you the chance to accept or reject each change individually, checking your notes ▼▼



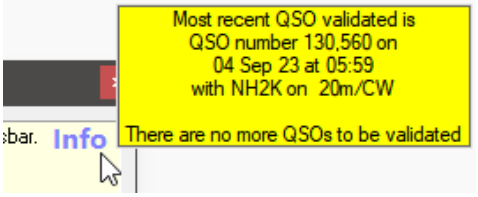
<**Do not correct DXCCs if QSO is confirmed by the ARRL**> stops the process 'correcting' QSOs that have already been accepted as valid by ARRL *e.g.* on receipt of LoTW confirmations or checked QSL cards. Whether right or wrong, they have been accepted towards DXCC or other awards, so why would you want to change them? Leave them be! It's water under the bridge. Suck it up.



<**I'm doing the full logbook, I will recalc stats later (I promise)**> skips updating the logbook statistics as any individual changes are made in order to speed up the checking process … but if, when it's all done, you neglect to update your statistics, your DXCC entity and CQ zone counts and other statistics may be inaccurate if you made any changes.

To start checking, click the upper large button <**Correct full log DXCCs in 10K QSO batches**> and watch the screen to see what it's doing.

If the validation function finds apparent DXCC or CQ zone discrepancies in a batch of 10,000 QSOs, and if you have wisely elected to confirm changes manually, you will be presented with information about each dubious QSO record (including any QSO notes from your log) and given the chance to accept or reject the suggested changes. Once the batch is completed, you can press ahead with the next batch by clicking the lower large button <**Correct another batch of QSOs**>.



Point your mouse at **Info** in the top right corner of the window or additional information about what is going on ▶

The validation checks are amazingly quick when you consider the amount of work going on but if you need to stop the process in a hurry, hit <**Abort**>. Logger32 keeps a running count of which QSOs have been validated, so after a batch is completed you can close the validation function using the corner X. When you return later, you can either start over from the top (click the upper button) or pick up where you left off last time (click the lower button).

If Logger32 identifies a potential discrepancy/anomaly in your log, you might like to research it further since, although Club Log's information is excellent, it is not perfect. Click to browse to the Club Log website.

Copyright © 2026 Robert C. Furzer K4CY

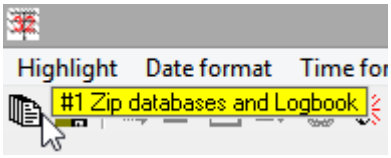
```
VP8ORK on 06 Feb 11 user said no  -> 238
GR2HQ on 13 Jul 14 user said no  -> 223
VY0M on 05 Aug 15 user said no 02 -> 01
KH7Y on 18 Feb 16 user said no to DXCC and CQ
9G5X on 14 Mar 17 user said no  -> 424
I88DX on 01 May 17 user said no  -> 022
KH7X on 24 Jun 17 DXCC/CQZ changed to 110/31
KR0YD on 05 Mar 18 user said no  -> 047
7Q7EI on 31 Mar 18 user said no  -> 440
FS/PE1IGM on 12 May 18 user said no  -> 213
KH1/KH7Z on 28 Jun 18 user said no  -> 020
KH1/KH7Z on 28 Jun 18 user said no  -> 020
```

◀ Temporary file *C:\Logger32\Logbook changes.txt* shows the changes that were suggested, whether you manually accepted or rejected them. Don't delay though: the file is wiped when Logger32 next starts. Make a copy if you want to check it later.

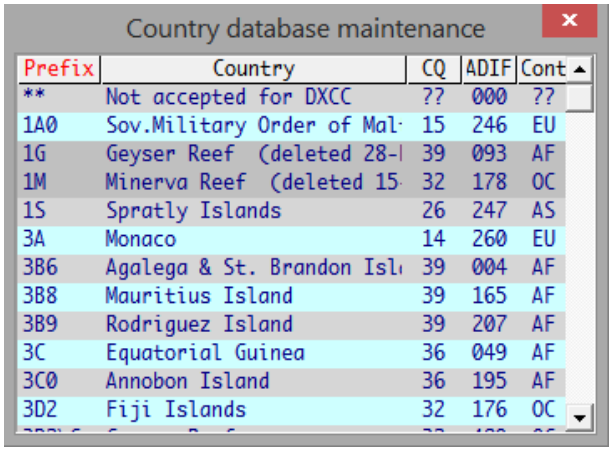
Behind the screen, the Club Log DXCC validation function quietly keeps track of the QSOs that have been checked already. Having previously checked your complete log, when you log further QSOs, the newly-added QSOs will be checked if you simply re-open the function and click <**Correct another batch of QSOs**>. If you wish, you can also re-start the process and check your entire log again at any time using <**Correct full log DXCCs in 10k QSO batches**>.

5.1.4 Manual database updates

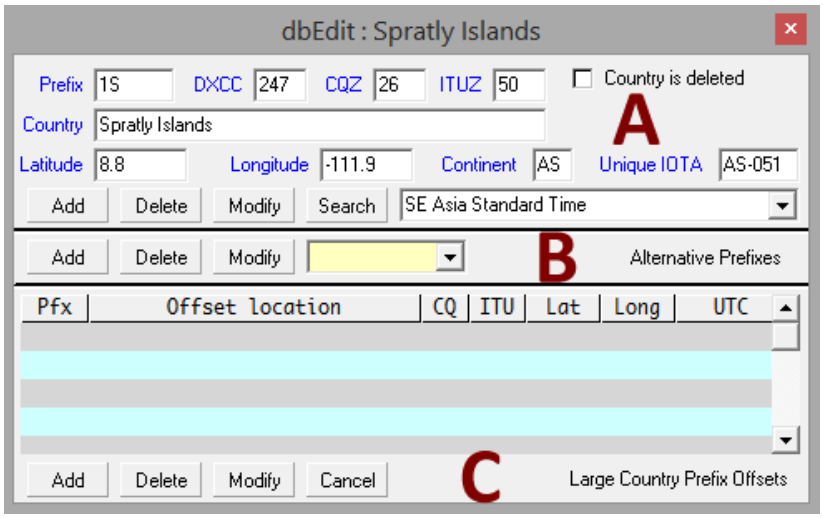
Before modifying any of the databases manually, make a Logger32 backup, just in case, using the left-most 'pages' toolbar icon #1 ▼



Use **Tools ⇨ Database maintenance ⇨ Country/Prefix maintenance** to open the country/prefix database maintenance function ▶



To change a particular entity (country), double-click its row to open the database edit form ▶



The form is divided into three layers or sections[77]:

A. Basic entity information across the top.

B. Alternative Prefixes across the middle (the yellow box).

C. Large Country Prefix Offsets across the bottom. We'll explain each section in turn …

[77] Notice that there are three <**Add**>, <**Delete**> and <**Modify**> buttons on this form, one set per section. Be sure to click the right ones!

Copyright © 2026 Robert C. Furzer K4CY

5.2 Maintaining DXCC entities (layer A)

The **prefix** links all three databases. This unique identifier, taken from the definitive ARRL DXCC list, is the key to getting anywhere in any of the three databases.

- **Add a new prefix:** click the blank (empty) line at the bottom of the Country database maintenance form. Logger32 expects valid information in all the edit boxes and will prompt you if any are missing or contain invalid data.[78]
- **Time zone**: the pull-down list at the lower right in section A lets you select the correct time zone for any entity. Clicking the down-arrow lists available time zones. If you are unsure, you can leave this entry blank for now *but* without a time zone here, Logger32 will not be able to determine the local time for that prefix. When satisfied that all is correct, click <**Add**>.

5.2.1 Deleting DXCC entities

There are two ways to delete an entity. First, select the row you wish to delete, then:

1. To *mark* the entity as having been deleted from the ARRL DXCC list but retain the information in the database, click <**Deleted**> then click <**Modify**>. The row gets a distinctive darker gray background in the Country Database maintenance form. If you made a mistake, or if the deletion is reversed by the ARRL committee responsible for DXCC (which *has* happened!), this change can easily be reversed by clicking <**Deleted**>. ***Marking* entities as deleted from the DXCC award scheme is the recommended method**.
2. If for some reason you feel you absolutely *must* remove the entity completely from the database, click <**Delete**>. Logger double-checks that you really wanted to delete it, since if you made a mistake, you would need to redefine the entity from scratch to reinstate it.

Actually deleting entities *may* create problems in Logger32's internal database. It is risky and is not recommended. **Are you *certain* you have usable backups before even attempting this?**

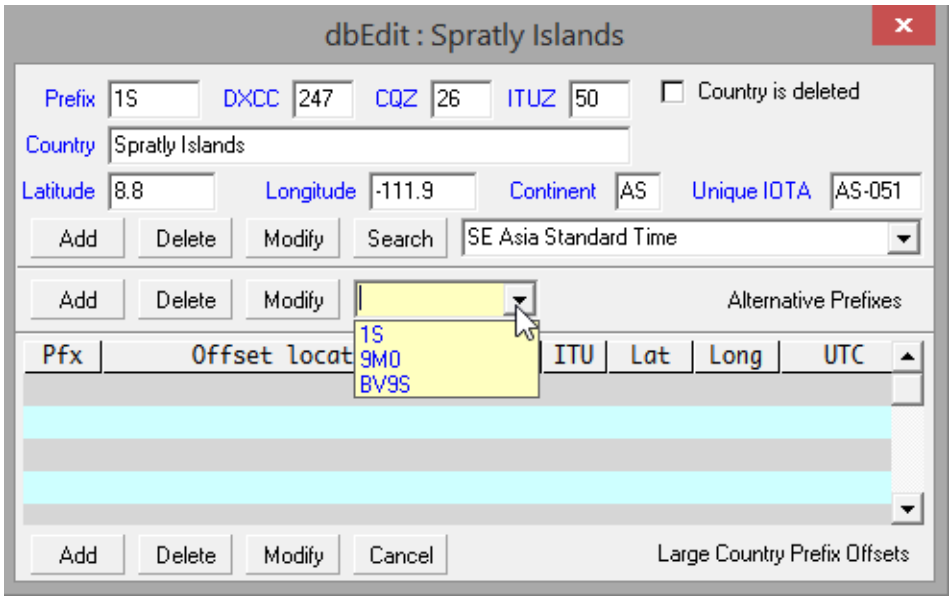
5.2.2 Modifying DXCC prefixes

To modify a prefix, select the row containing the entry you wish to modify, make the changes by editing the data, then either click <**Modify**> to save the modifications or, if you change your mind, click <**Cancel**> at the bottom to close the form without committing (saving) any of the changes.

[78] To enter an entity name that includes the "&" character, double it up. For example, "Antigua & Barbuda" should be typed as "Antigua && Barbuda".

Copyright © 2026 Robert C. Furzer K4CY

5.3 Alternative Prefixes (layer B)



The Alternative Prefixes database ▲ contains information about prefixes for the entity aside from the main entry (according to the current ARRL DXCC list). For example, the official DXCC prefix for USA is W. Valid alternative prefixes include A, K and 4U1WB. 4U1WB is included as an alias since that UN callsign, allocated to the radio club at the World Bank HQ in Washington, DC, counts as USA for the DXCC award, in the same way that 4U1VIC counts as Austria for DXCC.

5.3.1 Add an alternative prefix or specific callsign exception

To add an alternative prefix, click the down-arrow then click the blank row from the Alternative Prefixes edit box pull-down list. Type in the new alternative prefix and click the <**Add**> in the central part of the form - layer B.

As well as prefixes, you can include specific callsign exceptions in this section if required (see KC4/A - Antarctica). Enclose a specific callsign in angle brackets *e.g.* <K4CY>.

5.3.2 Delete an alternative prefix or specific callsign exception

To delete an alternative prefix|callsign, open the pull-down list, click to select the relevant prefix|callsign and then click the <**Delete**> button to the left of the yellow box in layer B.

5.3.3 Modify an alternative prefix or specific callsign exception

To modify an alternative prefix|callsign, first open the pull-down list and click to select it, modify it as required then click section B's <**Modify**> to commit the changes.

Copyright © 2026 Robert C. Furzer K4CY

5.4 Large Country Prefix Offsets (layer C)

The Large Country Prefix Offset database contains offset information for geographically large countries spanning several CQ zones.

For example, an offset is provided for each of the US call districts, such that the CQ zone defaults to 3 if the callsign is a W6, or zone 5 for a W4. Also, latitude and longitudes of the approximate center of each offset are maintained to estimate headings and distances. In the database, the offset prefix (W1, W2, VE1, VE2, VK6, VK7 *etc*.) must be unique.

5.4.1 Add an offset

To add a new offset, click to select the first available blank row in the list, usually the row at the bottom of the list. The time details are entered from the Time Zone list presented when you right-click the UTC field of the selected row. When an entry has been completed, click section C's <**Add**> button to commit the data to the database. **Always <Add> one line at a time.**

5.4.2 Delete an offset

To delete an entry, click anywhere on the row to be deleted and click <**Delete**>.

5.4.3 Modify an offset

To modify an entry, right-click the row to be modified, make the changes and click <**Modify**>.

5.5 Recalculate statistics

Facts are stubborn,
but statistics are more pliable.

There are lies, damned lies,
and DX statistics.

Apologies to
Mark Twain

Logger32 routinely checks and updates statistical information as QSOs are logged, stations are spotted or decoded *etc*. The statistics are held in tables within the countries database. Checking and updating the records involves some fairly intensive database activities that, for various reasons, don't always go entirely to plan *e.g.:*

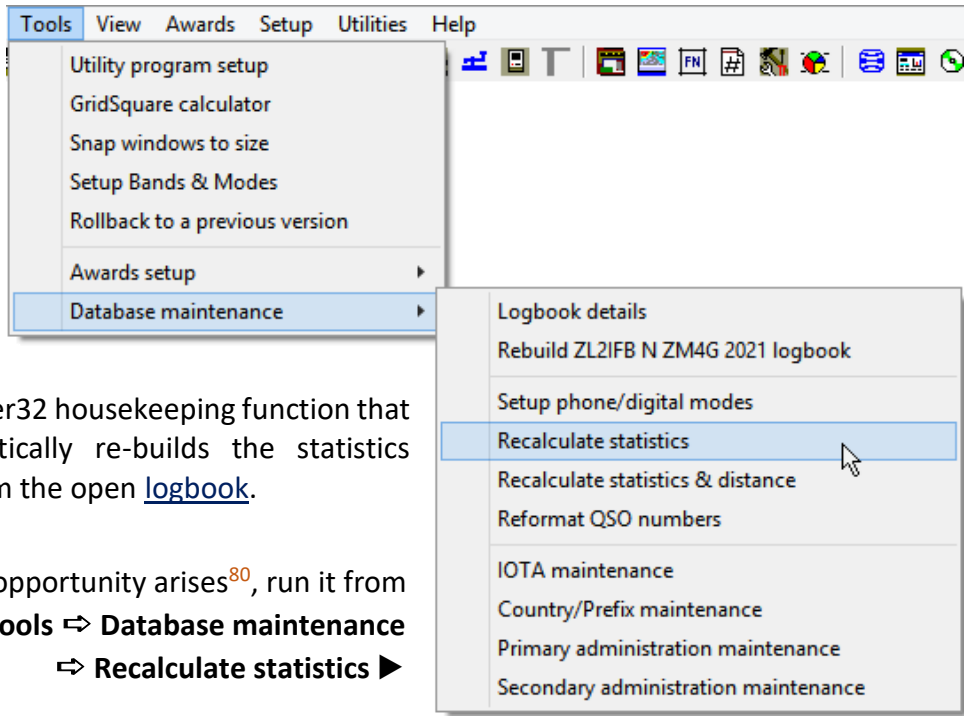
- We may *choose* to import QSOs and validate our logbooks *without* checking and updating the statistics, because the checks and updates slow the processes down. We are supposed to recalculate the statistics at the end, but may not always do so.

Copyright © 2026 Robert C. Furzer K4CY

- Changing QSO records, countries, awards criteria *etc*. should be accompanied by the system checking and updating the statistics … but these are distinct activities that may get separated for various reasons (*e.g.* inopportune power cuts and system shutdowns or crashes).
- Statistics tables may get corrupted by design flaws or bugs in the database management system, operating system, application software or other causes (*e.g.* backup, antivirus or other utilities may conflict with database maintenance activities).

Therefore, despite numerous internal controls to avoid, detect and recover from problems, they cannot be entirely eliminated in practice – just as sending, copying/decoding and typing errors and omissions are almost inevitable in our logs.

The odd busted callsign or mis-heard report is no big deal, in general, unless it happens to involve a particularly rare or unique QSO, and again various controls (particularly around confirming QSOs) help us keep the error rate as low as practicable – ideally affecting no more than one or two percent of our logged QSOs[79]. However, errors and omissions from the statistics tables are more serious since they can affect numerous awards tables and functions - such as the visual highlighting of new ones. Imagine if, say, P5DX genuinely showed up on 40m RTTY one evening, but despite the path being wide open while you were in the shack, the Logger32 alerts and alarms failed to go off hence you missed out on the opportunity for a vanishingly-rare QSO due to a mistake in your records ☹



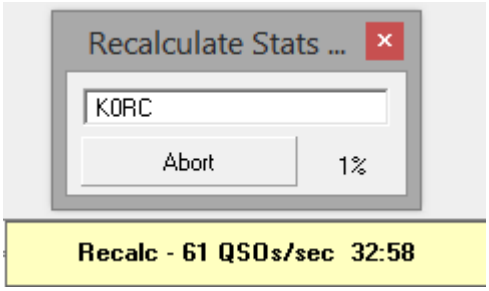
The solution is a Logger32 housekeeping function that erases and systematically re-builds the statistics tables, using data from the open logbook.

When the opportunity arises[80], run it from
Tools ⇨ Database maintenance
⇨ Recalculate statistics ▶

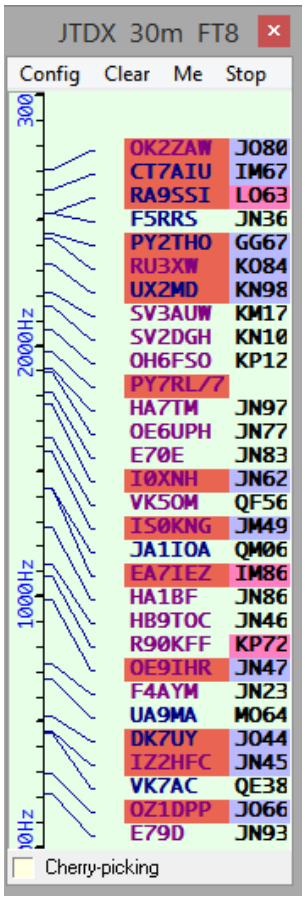
[79] Avid contesters and awards-chasers are painfully aware of this because contest and awards submissions are extensively checked by the adjudicators. Perfect logs are rare indeed. We are, after all, only human.

[80] Recalculating statistics ties up your PC for some while, processing roughly 200,000 QSOs per hour on my previous steam-powered shack PC. Bear this in mind if you have a big log. Meanwhile, you can't log any new QSOs, so pick a quiet time to run it … or else abort the run to retry later on - however, since your statistics are currently missing or incomplete, numerous spotted stations may appear to be 'new ones'.

Copyright © 2026 Robert C. Furzer K4CY



◀ As it runs through all the QSOs in your logbook, regenerating the statistics and awards tables, you'll see the callsigns flashing past and a percentage complete in the recalc window, plus a status message near the Windows clock showing the QSO checking rate and *estimated* time to completion in minutes:seconds.



Note: the recalculate statistics function starts by deleting the current statistics … hence any callsigns that are received from JTDX|WSJT-X as UDP messages while the recalc is in progress are initially flagged as all-time new ones, filling the UDP BandMap with spurious highlights, repeatedly sounding your audio alerts (if applicable) ▶

Likewise with the BandMaps receiving DX spots from the cluster.

As the statistics tables are rebuilt, the highlighting gradually evaporates until eventually, when the recalc completes, any highlights that remain are genuine new ones.

Hinson tip: the recalc runs a little faster if you close the UDP BandMap (and JTDX|WSJT-X), and disconnect from the DX clusters beforehand. Reserve more of your CPU power and disk cache for the calculations, especially on a tired old PC. A reasonably current 'office-grade' shack PC should process 5,000 QSOs in less than a minute, so if you know roughly how many QSOs you've logged, you can estimate the time required to recalculate the statistics for your complete log.

Hinson tip: if recalculating your statistics fails to resolve issues with your award reports, that suggests there may be integrity problems in the database holding your log, a potentially serious issue. Without delay, export your entire log as an ADIF file, then try rebuilding your log. If that doesn't work, try opening a new log and importing the ADIF file. If you are *still* out of luck, check through the ADIF file manually using a text editor and/or ask for help on the Logger32 reflector.

Hinson tip: as soon as you start the statistics recalc, Logger32 instantly deletes and then starts laboriously rebuilding its internal statistics records … so if you abort the recalc, you are left with incomplete stats … which means many DX spots will be incorrectly highlighted as new ones, whereas in fact they aren't. Get back to normality by re-starting the stats recalc and patiently letting it run to completion. Meanwhile, ignore those visual highlights and audio alerts. Don't abort it! Logger32 does not remember where it was up to when the stats recalc was aborted. *Next* time, don't even start the stats recalc unless you have the patience to wait for it to complete. With a big log on a tired old PC, you might leave it running overnight, having disabled or extended the Windows inactivity timeout for good measure.

Copyright © 2026 Robert C. Furzer K4CY

5.6 Recalculate statistics *and* distances

How far away are the stations that you have contacted? While you are working and logging them, Logger32 calculates and displays short path great circle distances[81] between your *currently-defined* station location and the likely locations for DX stations, on this basis:

- If a DX station's grid square (Maidenhead locator) is known[82], use that.
- Otherwise, use the center of the Secondary administration if applicable (*e.g.* a county if known for US stations).
- Otherwise, use the center of the Primary administration if applicable (*e.g.* the middle of the state for a station in Texas).
- Otherwise, use the center of the call district in the DXCC entity if applicable (*e.g.* central and lower North Island, New Zealand for a ZL2 station[83]).
- Otherwise, use the center of the DXCC entity.

Calculated distances can be logged at the time each QSO is made, using the best locational information available at that point. However, if a logged QSO is updated with a contacted station's location – maybe a different state or county or a 4-, 6- or even 8-character grid square as confirmed on a QSL card, or from QRZ *etc*., the logged distance may need to be updated as well[84].

Tools ⇨ Database Maintenance ⇨ Recalculate statistics & distance runs in the much same way as plain Recalculate statistics, except it *also* recalculates and updates the logged distances for QSOs that specify grid squares.

Note: if you maintain a consolidated log recording QSOs made from markedly different places (*e.g.* previous homes, or while out operating portable, mobile, at the radio or contest club, from a friend's shack, on holiday somewhere or on a DXpedition), the distances need to be calculated from the station location at the time the QSOs were made. Logger32 warns you if the open log has QSOs made by more than one operator (*e.g.* some at home, others while you were mobile using /M or portable using /P) but it often doesn't know about other QTH changes.

Hinson tip: the capability to record station locations with date ranges in order to be able to recalculate distances for all logged QSOs was mooted but not coded – yet.

If you are concerned about the accuracy of distances for logged QSOs, you could:

- Split your log into segments for QSOs made in different locations using partial log exports **File ⇨ Export logs ⇨ ADIF (.adi file)** based on the operator (your callsign/s) and/or the date ranges when you operated from each location (*e.g. Log at RF80.adi* and *Log at RE66.adi*).
- For each log segment:
    - Change logbook and create a new logbook with a name indicating the location.
    - Import the ADIF file or files for that location.

[81] Note that these are approximations with various degrees of precision. The calculations take account of the earth's non-spherical shape, but not the actual ionospheric path traced by the radio waves.

[82] While logging QSOs, Logger32 gets stations' locators from user inputs, from their FT# CQ messages, from callsign lookups, or from previously logged QSOs with those stations – in that order.

[83] As in the US, callsign numbers indicating specific areas are no longer obligatory within mainland ZL: call numbers are only a rough, error-prone guide to a ZL's location in the absence of better information.

[84] The LoTW synchronization function *does* recalculate distances when grid squares are confirmed on LoTW.

Copyright © 2026 Robert C. Furzer K4CY

- Specify the station QTH from where you made those QSOs.
- Update the distances using recalculate statistics and distances.
- Export the updated log as an ADIF file.
- Repeat for the remaining locations and log segments.

- Reassemble a consolidated log by importing all the updated ADIFs into a new logbook.
- Specify your present station QTH.
- Remember *not* to recalculate statistics and distances on the whole log.

5.7 The county database

Logger32 uses this database to check the validity of US county names against the applicable US states. Logger32's county database contains information about:

- US states.
- US counties.
- The latitude and longitude of the approximate center of each county or its biggest city or town.

When gerrymandering politicians meddle with the details, you can update the county database accordingly.

5.8 The IOTA database

The official IOTA listing[85] covers at least 1,100 Island Groups with more than 15,000 individual islands. IOTA-related information collated from various public domain sources has been compiled into a comprehensive but *unofficial* database provided with Logger32. It is not definitive and is provided for your convenience only.

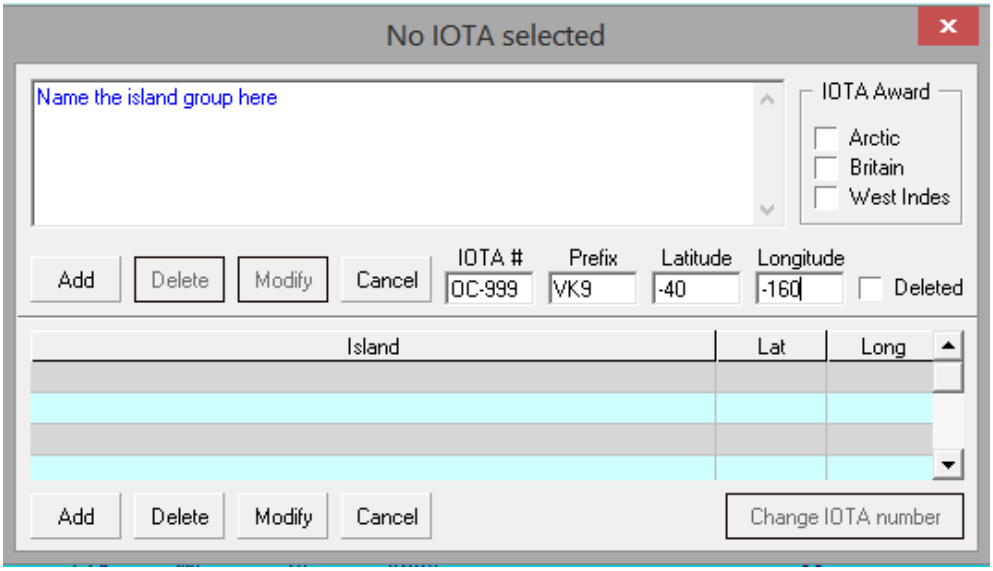
You are welcome to maintain the database if you are chasing the IOTA award. To do so, click **Tools ⇨ Database maintenance ⇨ IOTA maintenance.**

5.8.1 Add a new Island Group

To add a new IOTA Island Group to the database, click the blank row at the very bottom of the IOTA list to open a database maintenance form.

[85] The official IOTA listing is protected under copyright by IOTA Ltd., and is not provided as part of Logger32.

Copyright © 2026 Robert C. Furzer K4CY



◀ Name the Island Group in the large text box[86].

Supply the IOTA reference number, prefix, latitude and longitude.

If this Island Group qualifies for the Arctic, Britain or West Indies IOTA awards, click to tick the relevant box.

Click the upper <**Add**> button to complete the process and commit (save) the new Island Group to Logger32's IOTA database on your system.

Thank you Bob and team! I upgraded to v4 and then to the current version of v4 this weekend. I waited until people had more experience with the upgrade in case I had problems. I had an issue with the auto-update but the v4 manual addressed my issue and it is working great. Thank you for all of the effort on the program and the documentation. Your work is appreciated.

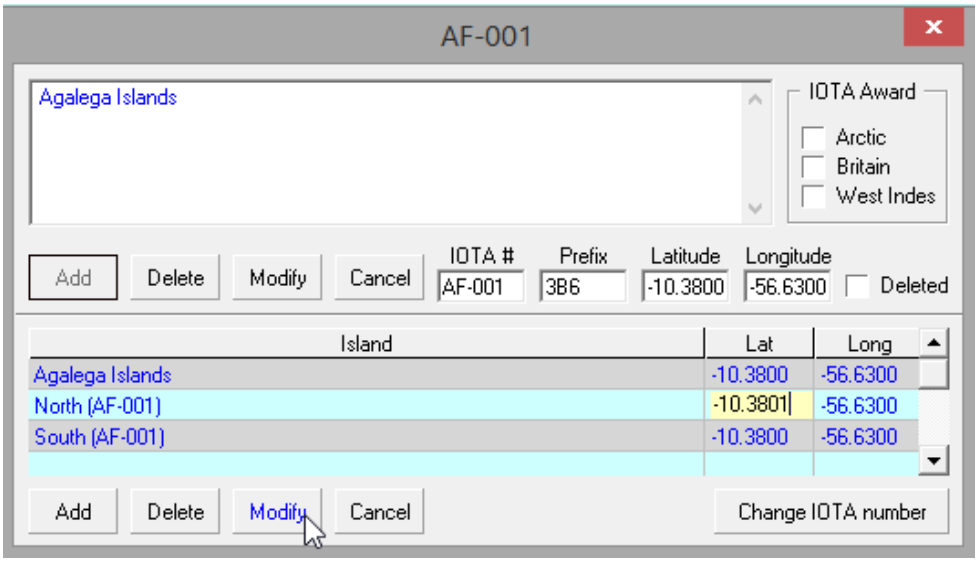
David W9PH

[86] Yes, there is room to write an entire paragraph … but stick to the official name of the Island Group.

Copyright © 2026 Robert C. Furzer K4CY

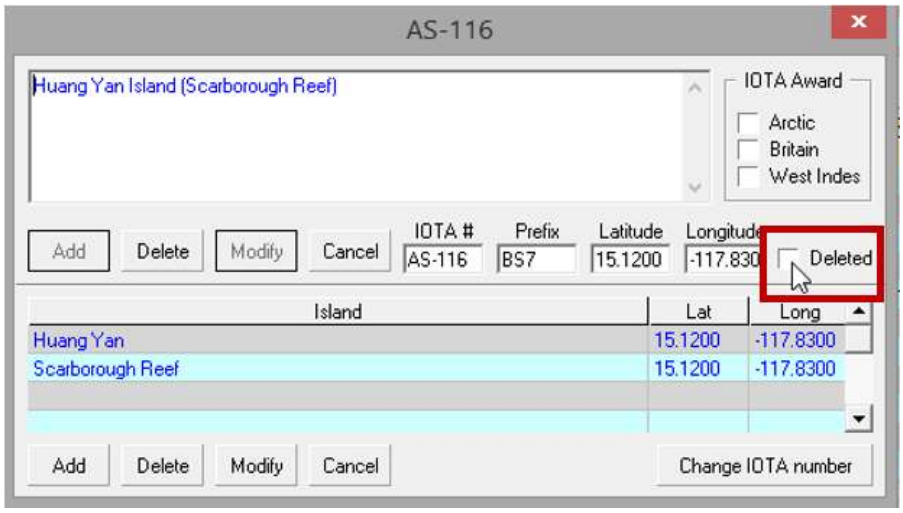
5.8.2 Modify an Island Group[87]

To modify an Island Group (*e.g.* to define individual islands within the group), click the island's row of the IOTA Maintenance form, make the required modifications then click the lower <**Modify**> to complete the changes ▶



5.8.3 Delete an Island Group

If an island group has been deleted and is no longer valid for the IOTA award, simply click to tick <**Deleted**> giving it a dark highlight (a lowlight? A highdim?) on the IOTA lists, while leaving the name and details available for reference purposes ▶



Alternatively, if for some reason you need to delete an Island Group entry completely from the database, click the appropriate line in the IOTA table, then <**Delete**>.

For more on Logger32's IOTA functions, see the IOTA chapter.

5.9 Database FAQs

Q. Which prefix was I editing? I've forgotten!

A. Do you recall the country name? Click the country heading to sort the database by that column. Maybe that will prompt your memory.

[87] To be clear, this is about changing the *information* concerning an Island Group, not changing the actual islands. Running a shack *does* contribute to climate change but not materially, so hams are not *entirely* to blame for submerging Scarborough Reef, Vanuatu and other low-lying places beneath the waves.

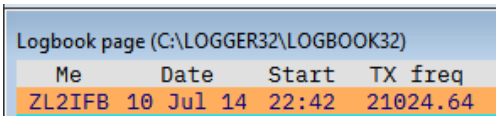
Copyright © 2026 Robert C. Furzer K4CY

Q. Why are some of the prefixes in my logbook wrong?

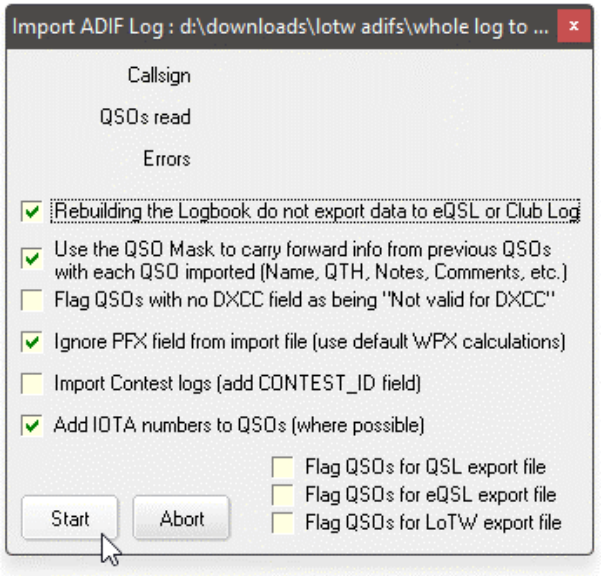
A. Are you sure? The prefix column is *not*, as many think, used for the DXCC prefix *i.e*. the default prefix for an entity as stated in the ARRL DXCC list. The ADIF standard specifies that the prefix field holds *WPX* prefixes for the WPX awards.

If necessary, the following process can update/correct the prefixes in your log:

1. Export your logbook in ADIF format. Remember or jot down the folder and file name.
2. Check the header line for the log page for the name of your currently open logbook ▼ Remember that too.



3. Shutdown Logger32.
4. Open Windows **E**xplorer using <**Win**+**E**> then:
5. Navigate to C:\Logger32 or wherever you have stashed Logger32 on your PC;
6. Open the Detailed folder view;
7. Click the file name column header to sort the files by name;
8. Select then delete[88] all the current logbook files – that is, ***all* the files whose names start with the name of your logbook** as noted at step 2, regardless of case.
9. Restart Logger32: it will open with your usual screen layout, color-coding *etc*. and a new empty logbook will be created automatically, with the same logbook name as before but no QSOs at this point. It's scary but *don't panic!* Continue calmly …
10. Import the ADIF log file saved in step 1with <**Ignore PFX field from import file (use default WPX calculations)**> ticked in the import options: that tells Logger32 to disregard any PFX fields in the saved ADIF file, identifying the correct ADIF-compliant WPX prefix values for each QSO as it is imported. You probably also want to tick several other import options, particularly <**Rebuilding the Logbook: do not export data to eQSL or Club Log**>, and un-tick the final 3 options ▶



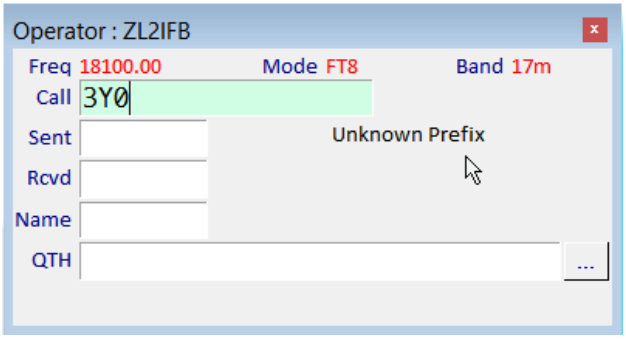
11. Wait patiently for the import to complete, then check your log and statistics to make sure all is well. If you correct and re-import any reported errors, you *may* want to correct your QSO numbers to bring them back into the proper sequence. You *may* need to recalculate statistics if the stats for awards *etc*. are broken - hopefully not. You *may* need to recover the files you deleted/moved earlier if the process fails catastrophically - hopefully not again. fingers crossed. Finally, if you feel there are incorrect DXCC entities and CQ zones in your log, use the Club Log validation function or zone-check.eu.

[88] Or move them to a temporary folder just in case later steps fail, and you can't retrieve them from the Windows recycle bin for some reason. For belt-n-braces, you might prefer first to save a backup copy of the *entire* Logger32 folder and any subfolders while you are at it, if you are as paranoid as me.

Copyright © 2026 Robert C. Furzer K4CY

Q. If I type 3Y into the log entry pane, why is it shown as an 'Unknown Prefix'? Have I broken Logger32? How do I fix this? A DXpedition to Bouvet has just been announced and I'm hyper-excited! I don't want to miss it! If Logger32 doesn't recognize their 3Y callsign as Bouvet, it won't be flagged as a new one for me, and I might miss out. It won't tell me which way to point my 5-over-5-over-5. I may not have time to warm up all my amps. Woe is me! I've been waiting patiently for *decades* for this one. I deserve it! I am *appalled* that Logger32 doesn't know that 3Y is Bouvet. How hard can it be? […]

A. Logger32 can't determine for sure where a station whose callsign starts with 3Y0 is *currently* operating from ▶



It *may* be a genuine DXpedition to Bouvet Island, a researcher who has just landed on Peter I Island, a hardy scientist based at Norway's Antarctic research station, someone in another Norwegian territory (since Norway officially 'owns' the 3Y prefix according to the ITU) or somewhere else entirely. It could of course be a pirate or simply a busted callsign.

In this case, updating Logger32's prefix records in the DXCC entity database with fresh information from Club Log about the 3Y0 prefix won't help because Club Log is in the same position: 3Y0 is an ambiguous prefix that can legitimately indicate one of several locations. Only when a unique 3Y0 *callsign* is allocated and goes on-air from the indicated DX location will the correct DXCC entity be known for sure.

You are welcome to check the latest set of entity allocations downloaded from Club Log: look for the file *C:\Logger32\cty.xml* – a plain text file containing xml data specifying each known DXCC entity. Here are the first two xml records from mine ▼

```
<clublog date='2021-04-21T19:30:34+00:00' xmlns='https://clublog.org/cty/v1.3'>
<entities>
        <entity>
                <adif>1</adif>
                <name>CANADA</name>
                <prefix>VE</prefix>
                <deleted>FALSE</deleted>
                <cqz>5</cqz>
                <cont>NA</cont>
                <long>-80.00</long>
                <lat>45.00</lat>
        </entity>
        <entity>
                <adif>2</adif>
                <name>ABU AIL IS</name>
                <prefix>A1</prefix>
                <deleted>TRUE</deleted>
                <cqz>21</cqz>
                <cont>AS</cont>
                <long>45.00</long>
                <lat>12.80</lat>
                <end>1991-03-30T23:59:59+00:00</end>
        </entity>
```

You can also check Club Log's exceptions, as processed by Logger32, in the data file *C:\Logger32\Club Log Exceptions.txt* ▶

```
VE1ST/NA14 is operating from DXCC 001 (Canada)
VE9ST/NA14 is operating from DXCC 001 (Canada)
VA3RLG/PM is operating from DXCC 001 (Canada)
VE2EDK/Z2 is operating from DXCC 001 (Canada)
VE2LJ/Z2 is operating from DXCC 001 (Canada)
VE2EDL/Z2 is operating from DXCC 001 (Canada)
K2A/KL7 is operating from DXCC 006 (Alaska)
K1Y/KL7 is operating from DXCC 006 (Alaska)
WL7AZC/AR is operating from DXCC 006 (Alaska)
N7JUX is operating from DXCC 006 (Alaska)
N1TX is operating from DXCC 006 (Alaska)
K5RD is operating from DXCC 006 (Alaska)
KA0SIM is operating from DXCC 006 (Alaska)
KA1NVZ is operating from DXCC 006 (Alaska)
KA7ETQ is operating from DXCC 006 (Alaska)
KA7TMU is operating from DXCC 006 (Alaska)
KA7ZRI is operating from DXCC 006 (Alaska)
KB2ZME is operating from DXCC 006 (Alaska)
KB6HPY is operating from DXCC 006 (Alaska)
```

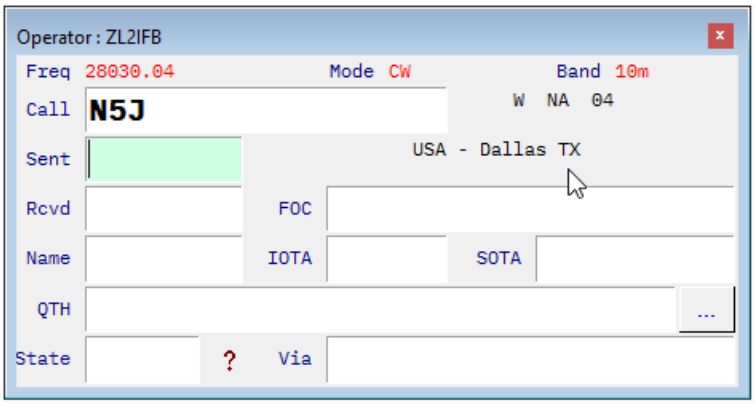
Copyright © 2026 Robert C. Furzer K4CY

Searching that file for "3Y" reveals an entry such as "LB4LC/3Y is operating from DXCC 013 (Antarctica)" – so straight away there's an example of a 3Y callsign that Logger32 knows is *not* Bouvet Island.

If a specific 3Y callsign is officially allocated for a forthcoming DXpedition to, say, Bouvet, and is disclosed to the ham community, Club Log will (very soon!) be updated to include an explicit country allocation for the full callsign (*e.g.* 3Y0B), with start and end dates corresponding to the expected duration of the operation[89].

The situation is different with some other ambiguous prefixes – E5 for example. E5 callsigns are *usually* used in the South Cook Islands, so if we copy a new E5 callsign, it is reasonable to presume (by default) that the DXCC entity is South Cook Islands … unless it turns out that someone is, in fact, operating from the North Cooks, in which case an explicit exception can be applied.

Q. I'm *so* excited! N5J is QRV from Jarvis Island, an all-time new one for me … but N5J spots are not highlighted, my alarm bells don't ring and when I go to log him, Logger32 tells me he's in USA Dallas TX ▶ I'm deflated … and slightly annoyed that I nearly missed this one …



A. N5J is a short 1x1 special event *callsign*. The N5 *prefix* almost always indicates a USA mainland station, so by default Logger32 naturally assumes that an N5 is somewhere on the USA mainland – probably in Texas – *unless* it has been told otherwise, for example by running the Club Log exceptions updater function or by manually adding a callsign exception to the DXCC database entry for KH5 Palmyra & Jarvis Islands.

Hinson tip: you'd better have a jolly good reason for *not* using the Club Log updater.

If you *insist* on manually defining an exception, add an entry for **<N5J>** to the Alternative Prefixes middle section of the form that opens with **Tools ⇨ Database maintenance ⇨ Country/Prefix maintenance**.

Those angle brackets around the call are important. They tell Logger32 that you are defining an exceptional *callsign*, as opposed to an exceptional *prefix*. If you accidentally omit the angle brackets, other ordinary run-o-the-mill N5Jsomething callsigns such as N5JAA will be incorrectly flagged as Palmyra & Jarvis too – in which case, you'll have to:

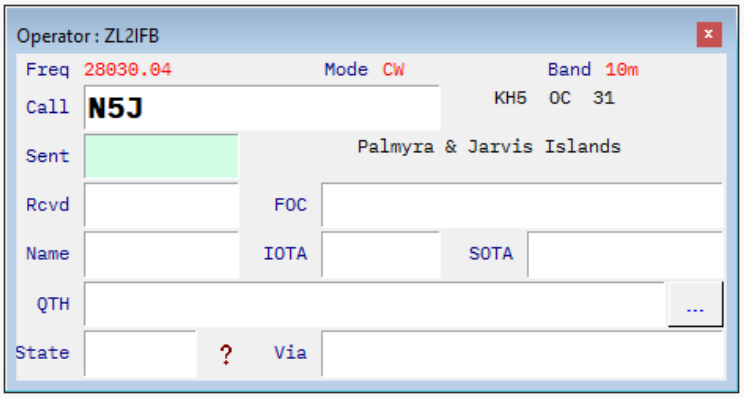
- Re-open the form using **Tools ⇨ Database maintenance ⇨ Country/Prefix maintenance**;
- Find the KH5 entry again;

[89] The dates are necessary to avoid sounding 'new country' alerts all around the world when someone naïvely spots the callsign prematurely on the DX cluster network to check his alerts, or when some idiot pirates the callsign from home prior to, or after, the DXpedition.

Copyright © 2026 Robert C. Furzer K4CY

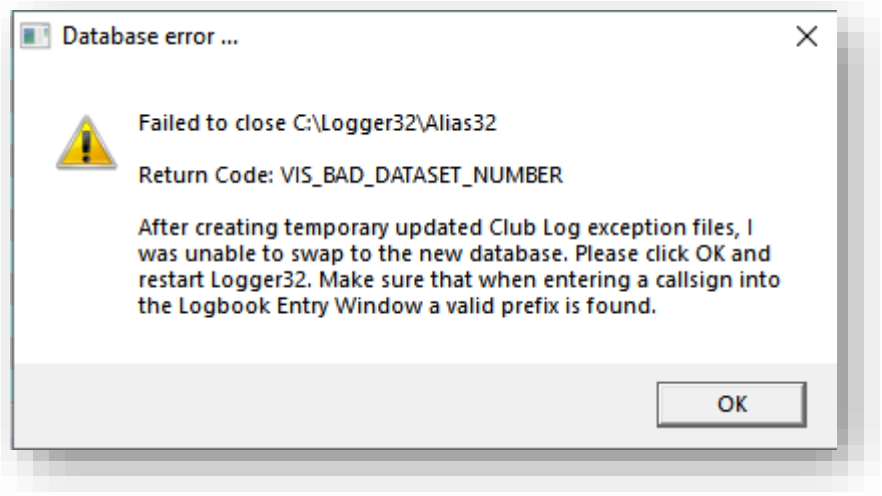
- Add the angle brackets to N5J in the middle section of the form; and
- Click the <**Modify**> button in the middle of the form to save and close it.

Check that N5J is now shown correctly as Palmyra & Jarvis Islands ▶



Finally, you may need to correct the logged DXCC entity for any logged *N5J QSOs with the Jarvis DXpedition*, if they are incorrectly shown as USA. If you have already excitedly exported, signed and uploaded the QSOs to LoTW, do it again in the hope of gaining the KH5 DXCC credit you crave. Don't 'correct' the DXCC entity for any other N5J QSOs that were *not* made from Jarvis. That would be, well, incorrect.

Q. Why do I get this error ▼ while updating DXCC exceptions from Club Log?



A. Logger32 was unable to swap the newly-updated country database for the live one already in memory, most likely because at the very instant it tried to swap, the database in memory was being accessed and so was locked. This can happen if you are looking up or logging QSOs at the same time as the Club Log exceptions updater runs, perhaps in JTDX.

The solution *may* be as simple as running the Club Log exceptions update again. The conflicts that prevent the database update are quite rare so you will *probably* be lucky next time.

Conflicts are more likely if you have a slow or busy PC, in which case temporarily avoid looking up callsigns or logging QSOs while you are running the Club Log exceptions updater.

Do one thing at a time. It's safer that way.

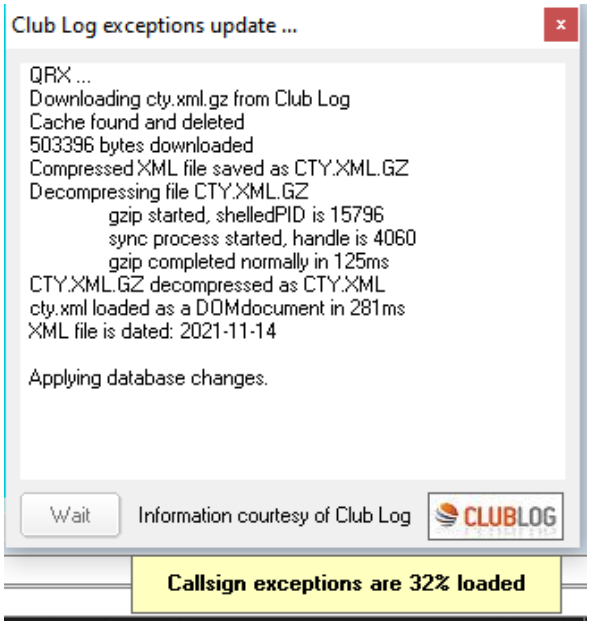
Hinson tip: if you ever get this error, check the DXCC entities for any QSOs that have just been logged. The same conflicts can prevent Logger32 looking-up the entities, hence they may be logged with entity code 000 meaning "Not accepted for DXCC".

Copyright © 2026 Robert C. Furzer K4CY

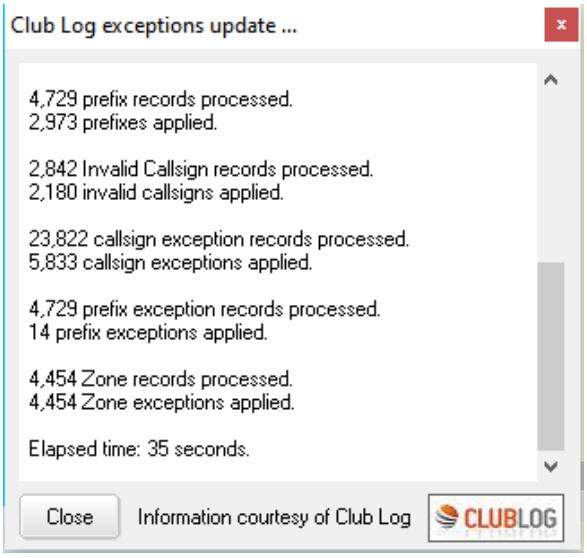
Q. OK, what *should* happen when I update DXCC exceptions from Club Log?

A. If everything goes to plan, the Club Log exceptions update process is as follows:

1. When the update process commences (either automatically soon after Logger32 starts up if so configured, or initiated manually by the user), Logger32 connects to the Club Log server with a specific URL (https://cdn.clublog.org/cty.php), passing an **A**pplication **P**rogramming **I**nterface key originally issued by Club Log for Logger32 use.
2. Club Log authenticates the API key … and [presumably] sends a return message before aborting the connection, if it is *not* authorized.
3. Club Log generates an e**X**tensible **M**arkup **L**anguage file (*CTY.XML*) containing the current set of DXCC entity allocations and specified exceptions, some with date ranges.[90]
4. Club Log compresses *CTY.XML* into a *.GZ* archive.
5. Club Log sends the *.GZ* archive to Logger32.
6. Logger32 receives and decompresses the *.GZ* archive using *C:\Logger32\gzip.exe* (installed during the Logger32 installation).
7. Logger32 loads *CTY.XML* as a **D**ocument **O**bject **M**odel object – essentially, a structured document, updating its DXCC database (containing callsign, zone and prefix exceptions), showing status messages as it does so ▶



8. Logger32 unloads the original DXCC database and loads the updated database instead.
9. Logger32/the database system checks the updated database integrity, displaying an error message if it fails.
10. Provided it completed successfully, Logger32 shows some final messages and the counts which it totted-up during the process ▶



If the Club Log update process fails, carefully check any error/status messages from Logger32 for clues about what might have happened. Possible culprits include:

- **Network/connectivity issues** – either no network connection at all to the Club Log server, or intermittent service. Did you break the Internet? Maybe Club Log's servers are down, but more likely there is a

[90] In fact, I think it uses a cached copy of this file. When the exceptions are next changed, I presume the cached copy will be replaced, and then fed to those who request it.

Copyright © 2026 Robert C. Furzer K4CY

networking/security issue on your PC/shack LAN, such as a problem with TLS or a firewall blocking any access to the URL noted in step 1 above.

- **Bugs and flaws in Logger32**: Bob and the beta test crew do our level best to eliminate all nasties before Logger32 updates are released but we admit to being merely human. Occasionally, the software runs fine, passes all the tests and is released … but then fails in service for *some* users under *some* circumstances – in which case the challenge is on to identify, diagnose and then fix the cause/s, most likely something unusual about the PC configuration or usage for those users who experience the failure, as opposed to those who don't. Additional information (such as screenshots of any error or status messages and details about exactly what steps will reproduce the failure) usually helps, and your patience is much appreciated.
- **Bugs, flaws or configuration issues in Windows, or in Club Log, or in some other app or utility or library associated with the process**: these can be even harder to diagnose and fix, and sometimes are beyond repair … in which case, tough luck: you may need to give up on the Club Log DXCC exceptions while patiently waiting a miracle. Meanwhile, it's worth checking that Windows and other software (such as your antivirus, and of course Logger32) is fully patched and up to date, in the hope that magically solves the problem.
- **Coincidences** such as callsign lookups happening at *exactly* the same time Logger32 attempts to update its country database (race condition).

Q. What are all Logger32's files for?

A. When Logger32 is installed and used, numerous types of files may be stored on disk in the chosen folder (typically *C:\Logger32*). The file extensions are a clue as to what they are:

- ***.ADI*** denotes an **ADI**F-formatted log file, an XML file that prepends the name and length of each ASCII data field in every QSO record, along with specified headers and footers.
- ***.BMP*** is a **b**it**m**a**p** image, such as the world maps generated by and used in Logger32's tracking window.
- ***.csv*** is a **c**omma-**s**eparated **v**alue file – yet another data communication format in which data values within each record are delimited by commas, tabs, semicolons *etc*. The first line of the file *may* specify the field names. Every record has the same sequence of fields.
- ***.db*** is a simple data file, essentially a table such as the one holding our customized bandplans (*MyBandMode32.db*).
- ***.dll*** is a **d**ynamic **l**ink **l**ibrary containing shareable functions that are dynamically loaded and used as and when required by one or more programs.
- ***.doc*** is a **doc**ument file, such as a merge file to generate QSL labels from log data using MS Word.
- ***.exe*** is an **exe**cutable application program file containing compiled computer code.
- ***.ini*** is an **ini**tial configuration file used to set up a program or function for use. Logger32 stores all its settings (configuration parameters) in *.ini*s rather than in the Windows registry – except for the Logger32 uninstall information.
- ***.isd*** is a **d**ata file containing details such as lists of states and counties.
- ***.isf*** is a file type that was used in Logger32 v3 but is no longer used with v4.
- ***.isl*** is a database **l**ock file. It is 0 bytes long and is not used by Logger32 which handles database record locking by a different mechanism.
- ***.ism*** is a map file, plotting the path of flight MH370 maybe?

Copyright © 2026 Robert C. Furzer K4CY

- ***.isx*** is a temporary file that may or may not be present in *C:\Logger32* whenever you display the folder contents using <**Win**+**E**> to open File **E**xplorer.
- ***.json*** is a **J**ava**S**cript **O**bject **N**otation file used to communicate data between systems and programs.
- ***.msi*** is a **M**icro**s**oft **i**nstaller – a package of files and instructions used to install or update an application on the computer.
- ***.txt*** is a plain **te**x**t** (ASCII) file, such as a list of amateur bands specified in the ADIF standard.
- ***.xml*** is an e**x**tended **m**arkup **l**anguage file containing named data field and data.

The name of your current logbook (*e.g*. "My logbook") is used in the file names for the associated database files (*e.g*. "*My logbook.isd*"). Since, unlike Linux, Windows is (largely) case-insensitive, it doesn't (usually) matter whether disks, folders, filenames and file extensions are in CAPITALS, lowercase or MiXeD.

"I just want to thank Bob and the rest of the crew for all the exceptional work you do to keep Logger32 up and running, including pertinent updates and new features. I have had very minimal problems and when I do it is mainly attributed to operator error. Logger32 is an amazing and functional program that does a lot more than most of the ones that actually cost money. I am sometimes aghast to see some of the requests and complaints that come in. Evidently people have no idea of what it takes to develop and update it. Just put in the request, wave the magic mouse, and it is done. Just my opinion and it may not be worth the data bits it is written in but – people enjoy the program provided. It cannot ever be all things to all people."

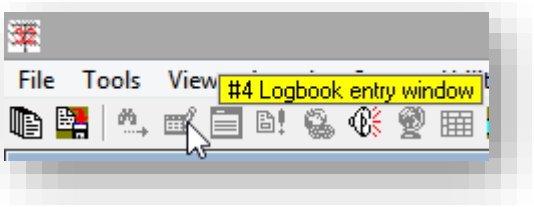
Larry K8YYY

Copyright © 2026 Robert C. Furzer K4CY

6 The log entry pane

> "Data is a precious thing
> and will last longer than
> the systems themselves"
>
> *Tim Berners-Lee*

Open the log entry pane with pen-and-paper icon #4 on the toolbar▶



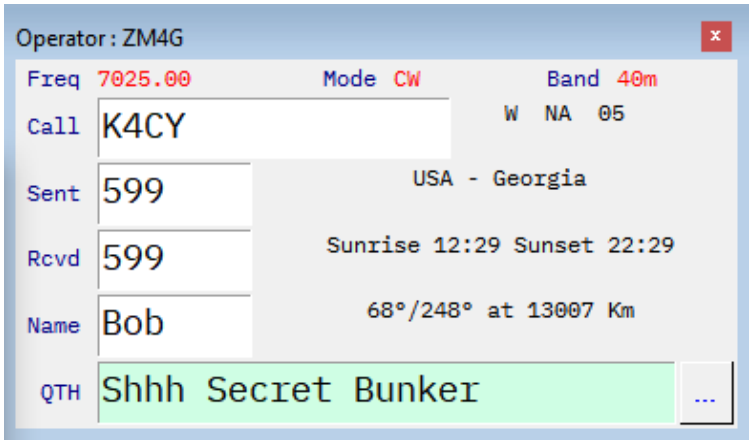
The log entry pane is where you enter details of QSOs in progress, recording each QSO into the open logbook, making it arguably *the most critical and useful window* in Logger32.

There is lots going on in this pane with numerous configurable options and shortcuts for common functions. *Please* study this section of the User Manual carefully to get the most out of Logger32.

Pay attention at the back: this stuff is important!

6.1 What we see in the log entry pane

Since the log entry pane is highly configurable, yours may not look *exactly* like mine but it will be vaguely similar ▶



- The top-line caption clearly shows the current **operator**: me, ZM4G in this case. On your system, that's you … unless I am there as a guest op! QSOs will be logged with the callsign of the operator, so ***be sure the callsign you are actually sending on-air is shown here***: simple enough if you only ever use the one callsign but all too easily overlooked if you use several (*e.g.* regular and contest calls, portable or mobile calls, club calls, special event calls …) or if you host guest ops in your shack and they (quite sensibly!) want to use *your* wonderful Logger32 system to log QSOs they make under *their* own callsigns.

Hinson tip: if you are, say, a QSL manager just reviewing and updating your clients' DX station logs rather than logging new QSOs, feel free to click the corner X to close the log entry pane.

Copyright © 2026 Robert C. Furzer K4CY

- The next line shows the **Freq**uency, **Mode** and **Band** that will be logged:
  - The **Freq**uency is captured directly from our CAT-connected radio. If the radio is CAT-unconnected, we can click here then type in the frequency to be logged for a QSO[91].
  - The **Mode** and **Band** are normally determined from the frequency according to our bandplan, but they may also be entered manually or captured from our CAT-connected radio (*e.g.* if we are using CW in what would normally be the sideband part of a band).
- Below that are several data-entry fields for:
  - **Call** (callsign) of the DX station you are contacting. This is a mandatory field: a QSO cannot be logged without a callsign. A QSO without a pair of callsigns is not a QSO[92].
  - The reports we **Sent** to and **Rcvd** (received) from the other station *e.g.* 599. The report format varies according to the mode. These fields may be pre-filled with default reports if defined for the current mode: we simply edit the text if the default values are not appropriate for any contact, or update/remove the default reports from the Bands & Modes table if we really don't like them.
  - The DX station operator's **Name**. If we have worked and logged him/her before, his/her name can be looked up in our log and shown here as soon as we type his/her callsign and move the cursor to another field. Otherwise we can type it in.
  - Further *optional* fields for other QSO-related information *e.g.* his/her **QTH**, **IOTA** reference, **State, via** (his/her preferred QSL route) *etc*.
- Below the **Band** is an area[93] for information about the DX station's location – specifically:
  - Notional prefix taken from the current ARRL DXCC list for the DX entity (not necessarily the first bit of the actual DX callsign being used on-air), plus the continent and CQ zone.
  - DXCC entity, mostly conventional 'countries' plus various DXCC-specifics and islands.
  - Sunrise and sunset times at the DX location in UTC (or polar day|night).
  - Short and long-path headings and short-path (great circle) distance from us or, more precisely, from our specified QTH latitude and longitude[94] to the likely DX location.

Hinson tip: *not* shown in this pane are the UTC dates and times for the start and end of the QSO that will be recorded when it is logged. Logger32 calculates and logs the UTC dates and times from the system clock, adjusting automatically for our local time zone including daylight savings, if any.

[91] Typing in a frequency here merely logs it and does not QSY a CAT-connected radio. To QSY a CAT-connected radio to a particular frequency, use the Quick QSY function … or click a DX spot on/near that frequency … or reach out laboriously across the shack desk to use the radio's band buttons and VFO knob if they haven't seized up yet due to lack of use.

[92] If we want - or are legally obliged - to log our interference tests, CQ calls, tune-ups, EMF measurements, shiny new radios/antennas *etc*., we can simply log informational entries (pseudo-QSOs).

[93] If we choose to display user fields in this area, there may be no room for the DX location information in the log entry pane, but the information always appears on the DX information bar near the bottom of the screen – provided there is at least a recognizable prefix in the Call field. If not, there is no DX location to look up! The identified DX location may be refined as we type the call area number or suffix.

[94] If headings and distances to stations we are logging are all wrong, our station QTH for the current operator is probably the culprit. If headings and distances are only wrong for specific stations, *their* location details are the problem.

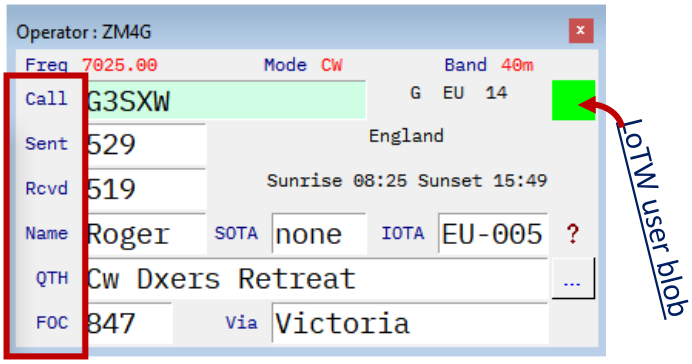
Copyright © 2026 Robert C. Furzer K4CY

6.1.1 Things to do in the log entry pane

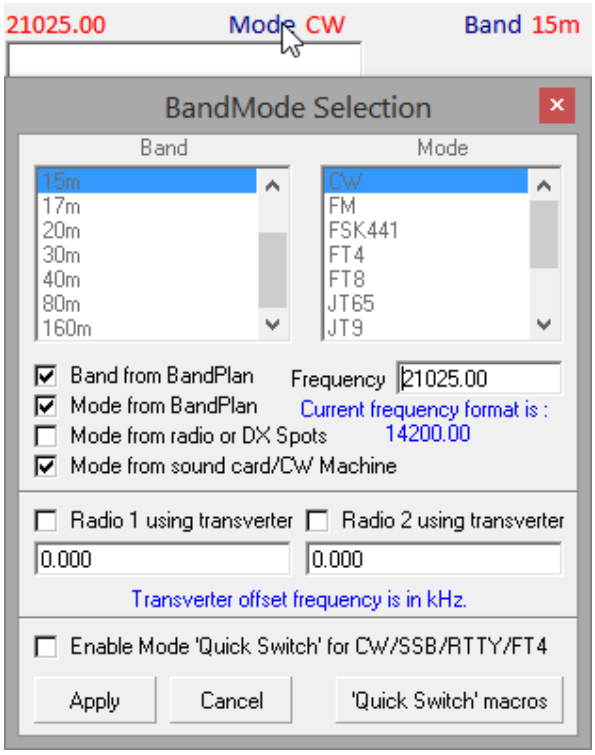
Mostly, you use the log entry pane to enter and record information about a QSO in progress, typing the details into the relevant fields and using the **tab** or **arrow** keys or **mouse clicks** to move between the fields. If you've worked the same station previously, Logger32 can pre-fill various fields with information from your log such as his name and QTH, using the QSO mask.

When you have completed all the information you wish to record against the QSO, hit <**Enter**> or <**Ctrl**+**L**> to save the QSO to the logbook (*i.e.* **L**og it!) and clear the log entry pane, ready to log your *next* QSO.

To clear the QSO data *without* logging it (*e.g.* if you have been calling some rare DX in a pileup, in vain, without completing a QSO) hit <**F11**>, <**Ctrl**+**C**> or <**Alt**+**W**>, or simply click the left side of the pane in the area of the field labels ▶



6.1.2 ADIF MODE and SUBMODE



For your convenience [95], Logger32 displays a custom logbook field for mode, neither ADIF MODE nor ADIF SUBMODE. To explain:

- CW is one of the ADIF MODEs. If the radio is in CW mode and <**Mode from BandPlan**> is selected in your BandMode Selection form ▶ (right-click "Mode" in the log entry pane), the log entry pane shows "Mode CW". QSOs are logged with mode=CW.
- SSB is also one of the ADIF MODEs, while LSB and USB are ADIF SUBMODEs of the ADIF MODE SSB. If the radio is in LSB or USB and <**Mode from BandPlan**> is selected, the log entry pane shows "Mode SSB" and the QSO will be logged with mode=SSB.
- PSK is one of the ADIF MODEs, while PSK31, PSK63 *etc*. are ADIF SUBMODEs of PSK. If you are making QSOs in PSK31 using MMVARI with the radio in the appropriate mode for PSK31 (*e.g.* DATA) and <**Mode from sound card/CW Machine**> enabled, the log entry pane conveniently shows "Mode PSK31". In the logbook, this QSO shows mode=PSK31[96]. However, when exported as an ADIF file, the QSO record would have the appropriate ADIF fields *i.e.* <MODE:3>PSK *and* <SUBMODE:5>PSK31.

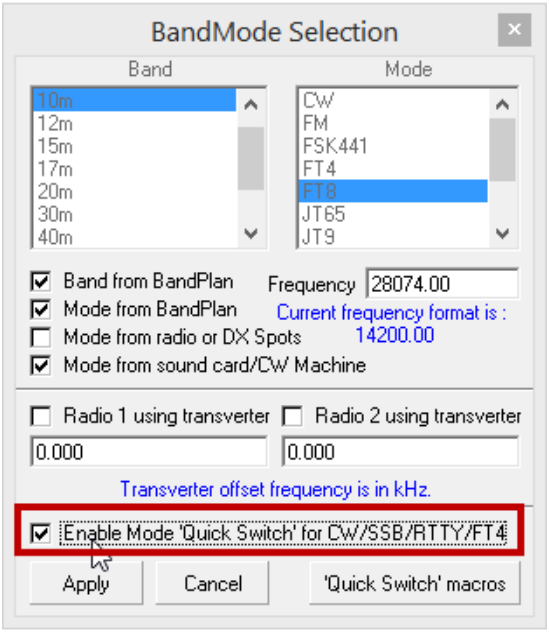
[95] The *real* reason for this capability in Logger32 is to gloss-over/handle/hide inconsistencies in the way radio modes & protocols are specified in the ADIF standard. Some are "modes", some are "submodes".

[96] If we prefer to log this QSO showing "Mode PSK", we can disable <**Mode from soundcard/CW Machine**> and enable <**Mode from bandplan**> instead.

Copyright © 2026 Robert C. Furzer K4CY

The text file *C:\Logger32\ADIFModes.txt* lists valid ADIF MODEs and SUBMODEs. It is used by Logger32 to offer only recognized modes and submodes, to generate the correct mode and submode statistics, and to output ADIF-compliant files.

6.1.3 Quick Switch



◀ Right-click **Mode** at the top of the log entry pane and click to tick and enable **Quick Switch**.

Now, when you click a DX spot for a mode with a mode-specific function, the relevant function starts automatically. If you are currently on SSB when you click a CW spot, the CW Machine can (optionally) be opened. If you then click an FT8 DX spot, the CW Machine is closed and the UDP BandMap is opened instead, then JTDX|WSJT-X is launched on FT8 mode.

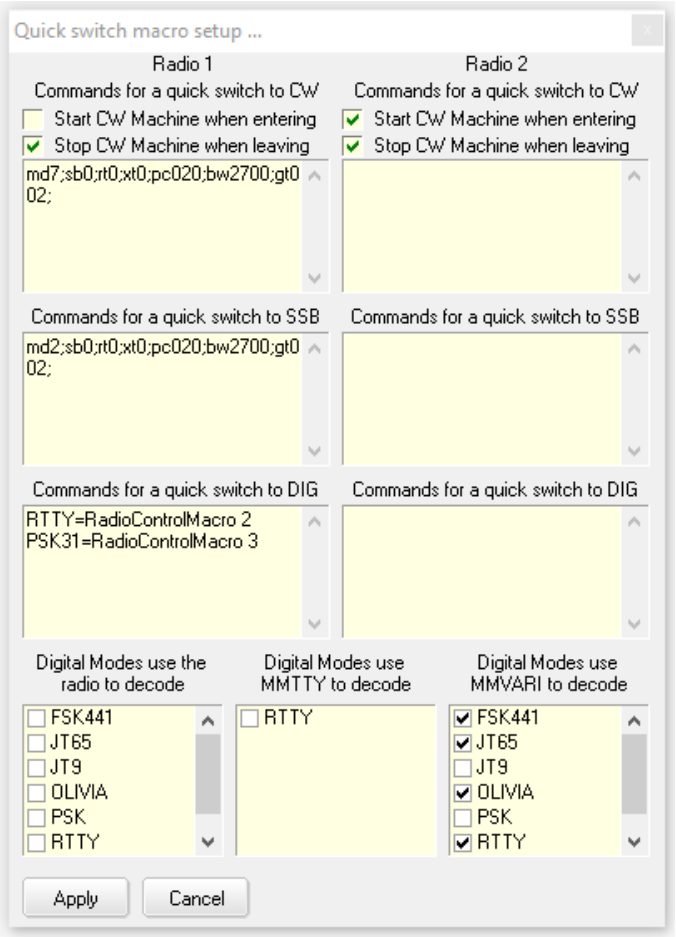
You can also trigger a **Quick Switch** by opening another window yourself[97] *e.g.* after working SSB with the DVK open, if you open the UDP BandMap to operate FT# through JTDX, Logger32 closes the DVK automatically, its job done for now.

The <**'Quick Switch' macros**> ▲ button lets you specify macros and CAT commands to be sent after switching modes ▶

De-select (un-tick) <**Start CW Machine when entering**> near the top if you *don't* want Logger32 to launch the CW Machine automatically when you click a CW spot.

Select (tick) <**Stop CW Machine when leaving**> to close the CW Machine automatically when you change to a different mode, whether it was opened automatically or by you.

Although FT# modes are not shown on this form, when you click an FT# spot, Logger32 dutifully opens the UDP BandMap and launches the digimode software (JTDX or WSJT-X) if it is not already going, running any starting macros defined under **Start|Stop ⇨ Setup shortcuts** on the UDP BandMap menu. It also starts the JTDX Control Panel if applicable. Likewise, Logger32 closes the digimode software, runs any closing macros



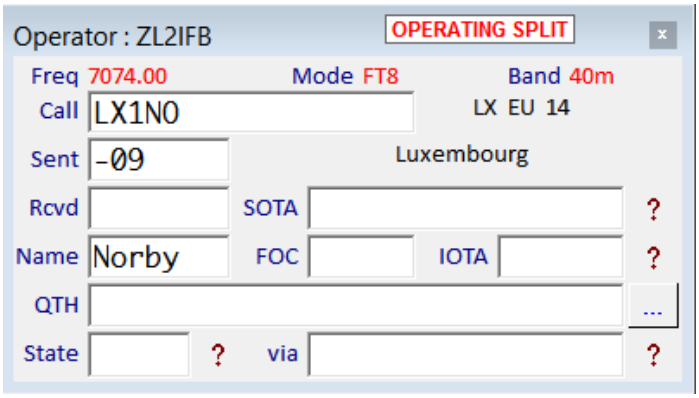
[97] Unfortunately, the Quick QSY function does *not* trigger a Quick Switch. That would be way too cool.

Copyright © 2026 Robert C. Furzer K4CY

defined under **Start|Stop ⇨ Setup shortcuts** and closes both the UDP BandMap and the JTDX Control Panel (if it was open) when you click a legacy mode DX spot.

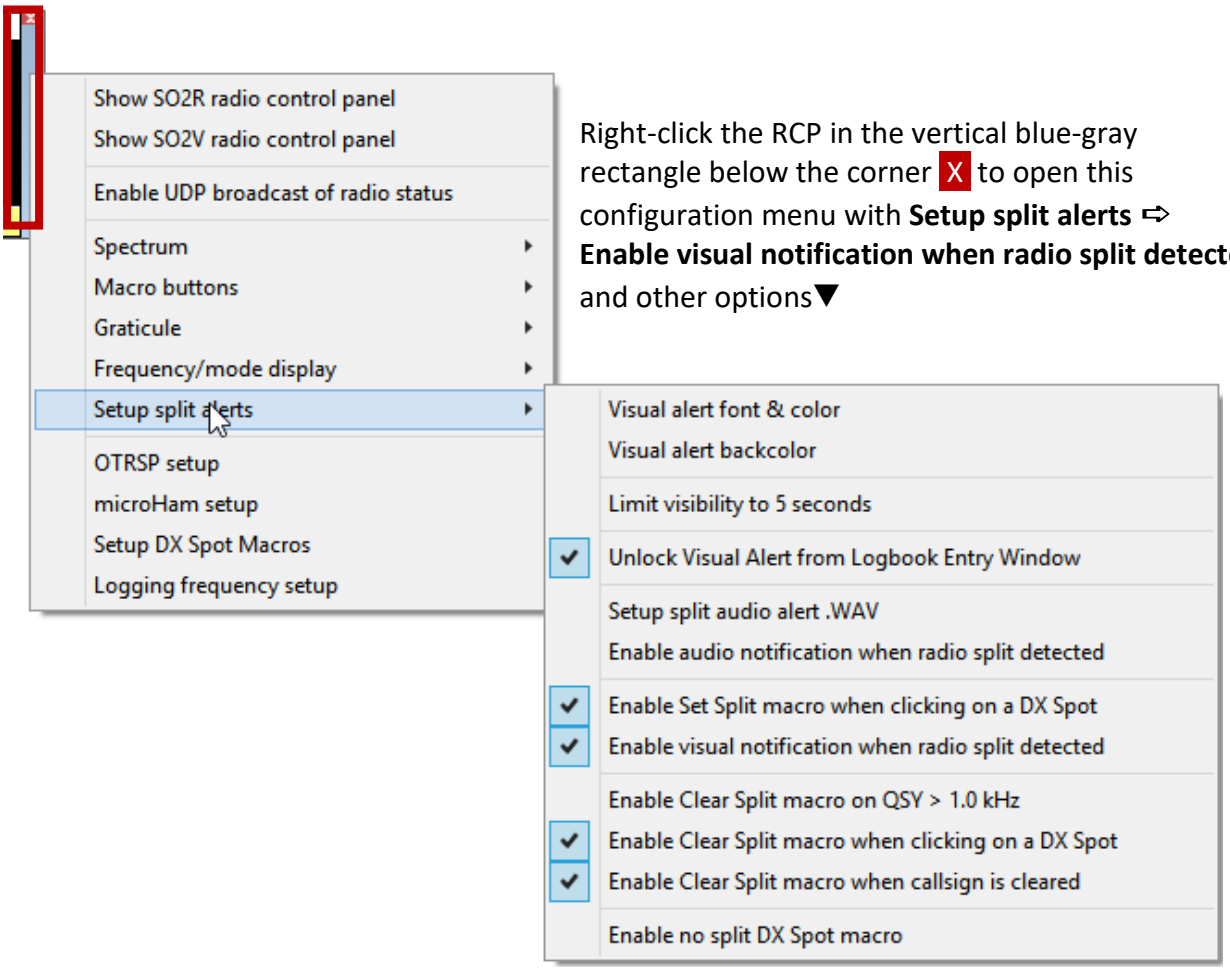
6.1.4 Split operation

After clicking a DX spot with a recognized split syntax in the comments (*e.g.* "UP 5"), the log entry pane header can show **OPERATING SPLIT** as a prominent reminder that the radio is using different TX and RX frequencies ▶



Click the OPERATING SPLIT indicator to cancel split, returning the radio to simplex (single TX/RX frequency) operation.

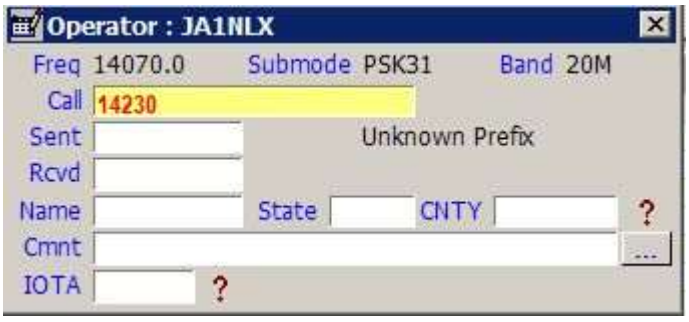
To enable and configure the split indicator to your liking, see the **R**adio **C**ontrol **P**anel chapter. Aside from changing the colors, you can unlock it from the log entry pane to move it to another part of the screen, generate a sound when it appears and make it disappear from view after 5 seconds, trigger radio macros *etc*.



Right-click the RCP in the vertical blue-gray rectangle below the corner X to open this configuration menu with **Setup split alerts ⇨ Enable visual notification when radio split detected** and other options▼

Copyright © 2026 Robert C. Furzer K4CY

6.1.5 Quick QSY



◀ Logger32 can QSY your CAT-connected radio directly to a chosen frequency.

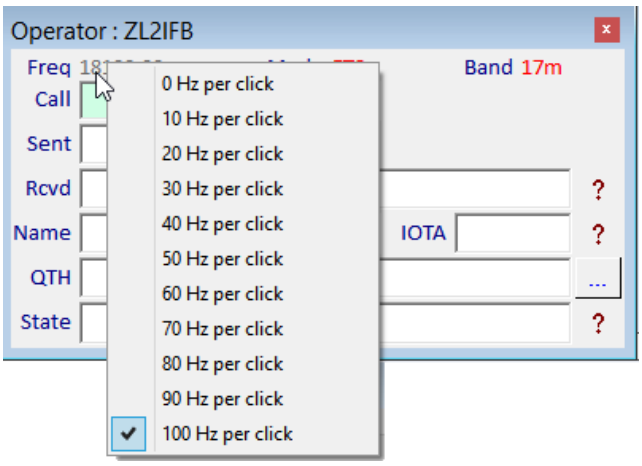
For instance, if someone asks you to QSY to another band during a QSO, simply type the agreed frequency into the Call field using the same format as the frequency shown directly above the Call field, and *when you are ready to QSY*, press <**Enter**> or <**Tab**> as if you were logging a QSO[98]. The radio will *instantly* QSY and may also change mode in accordance with your bandplan (Bands & Modes table). The antenna will also switch automatically if used.

6.1.6 Mouse QSY

The current VFO frequency of a CAT-connected radio can be changed by turning the scroll wheel on the mouse (or a USB knob programmed to behave like a mouse wheel). To use it, point the cursor at the log entry pane, then turn the wheel. Turning one way moves Radio 1's VFO-A HF, the other direction moves it LF.

The *rate* at which QSY commands are sent to the radio depends on the polling interval of your CAT interface: the response may be sluggish compared to reaching over the shack desk to spin the radio's VFO knob.

However, you can adjust the *step size* after right-clicking the frequency value ▶



Hinson tip: thanks to using 'reverse split' (TX as usual on VFO-A but RX on VFO-B, the sub-receiver), I *could* use this to tune through a DX pileup, looking for a clear frequency or one recently vacated by a successful caller on which to call the DX. I prefer to twiddle my knob, though.

6.1.7 The function keys

While the focus is on the log entry pane, clicks on the F-keys (function keys) get passed through to the following windows, in the following priority sequence:

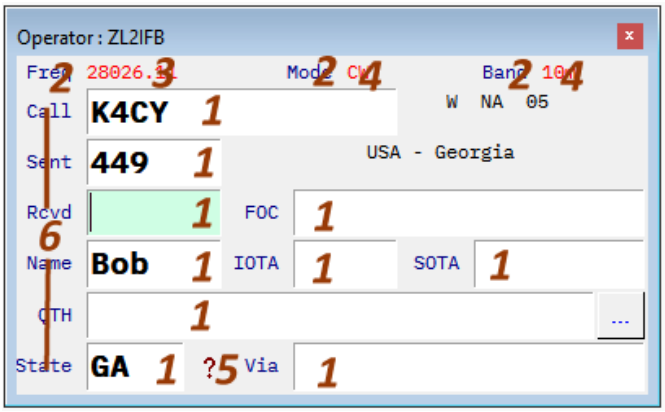
1. The CW Machine *if* it is open.
2. Otherwise, the Sound card data window if it is open.
3. Otherwise, the DVK window, if it is open.
4. Otherwise, they end up back at the log entry pane itself. One or two F-keys have defined actions on the log entry pane (*e.g.* F11 clears the data from the pane and F12 selects the File menu). Otherwise, they have nothing to do and are simply ignored.

[98] The frequency is automatically wiped from the Call field as Logger32 sends the QSY CAT command to the radio. Since it is all numeric and *obviously* not a callsign, it is not logged as such!

Copyright © 2026 Robert C. Furzer K4CY

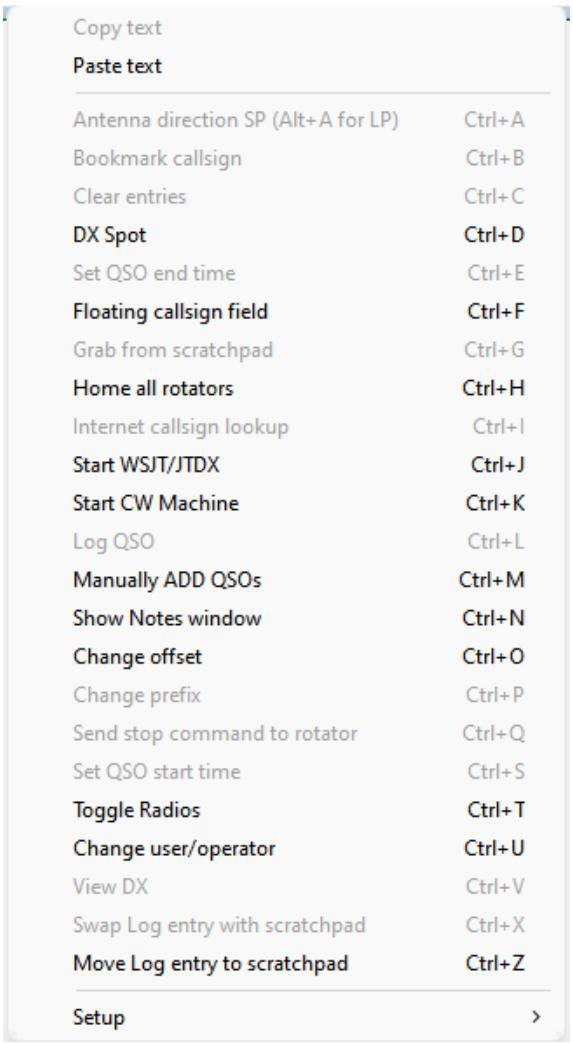
Hinson tip: it makes sense (to me) to define F11 similarly in those other windows *i.e*. to run the $**ClearLog**$ macro. But if you want to be obtuse and define it as something different, go ahead.

6.2 Log entry pane right-click menus and functions



◀ Several menus or functions are opened by clicking the log entry pane, depending on whereabouts in the pane the mouse cursor is pointing when you left- or right-click:

1: Right-click various areas such as the data entry boxes to open the log entry pane main menu ▼.



2: Right-click the words “Freq”, “Mode” or “Band” to open the Bands & Modes configuration.

3: Right-click the VFO frequency value to choose the rate at which rolling the mouse wheel QSY’s your CAT-connected radio.

4: Right-click the mode or band value to change to <**Right to left Reading order**>.

5: Click any **?** to show additional information about the entry currently in the field to its left (see below).

6: Click here on the left side to wipe any data currently in the log entry pane, instantly.

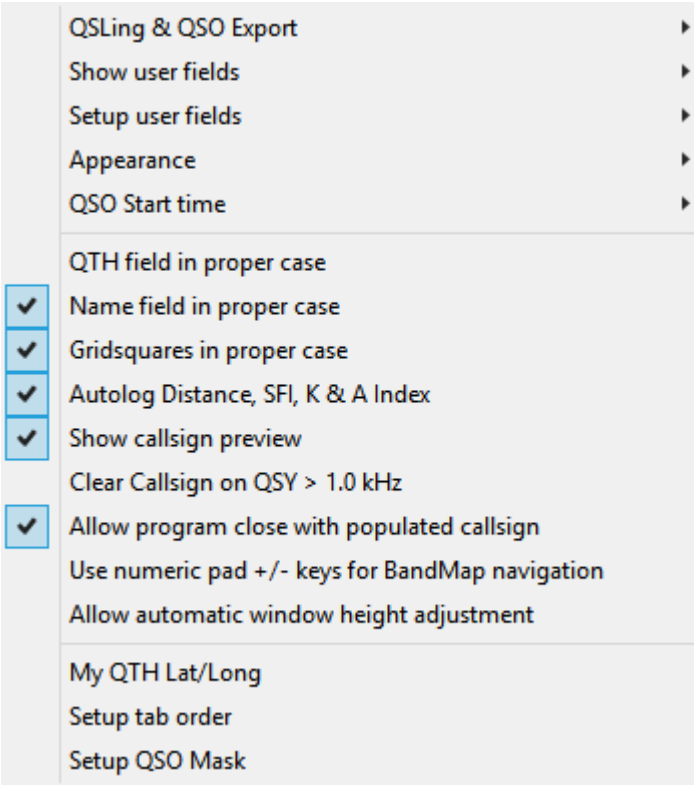
The main right-click menu lists a *stack* of program functions along with their convenient control-key shortcuts ▶

Click any function that *isn’t* grayed-out or press its shortcut key to run it. Most are self-explanatory. Functions that *are* grayed-out on the menu are currently unavailable *e.g.* <**Ctrl**+**B**> can only bookmark an interesting DX station *if* there is a callsign in the log entry pane’s Call field to bookmark! Likewise with functions that retrieve entries from the scratchpad: if the scratchpad is currently empty, they don’t work.

Hinson tip: most of these shortcuts only work while the focus is on the log entry pane *i.e*. while we are logging a QSO here. <**Ctrl**+**T**> (**t**oggle radios) and <**Ctrl**+**A**> or <**Alt**+**A**> (rotate antenna) may also work with the focus elsewhere *if* configured as global hotkeys in the Radio configuration and Rotator setup windows.

Copyright © 2026 Robert C. Furzer K4CY

6.2.1 Log entry pane right-click *Setup* menu



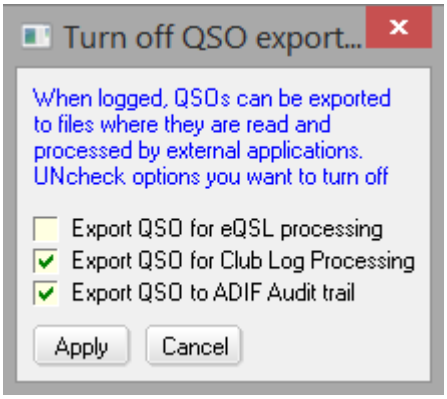
◀ Right-click any data-entry field in the log entry pane then click <**Setup**> at the bottom of the menu to access these 5 submenus and 12 configuration options. Read on for more …

6.2.2 QSLing & QSO Export

From the log entry pane right-click menu, **Setup ⇨ QSLing & QSO Export** opens a submenu to choose which if any QSL flags should be set when new QSOs are logged ▶



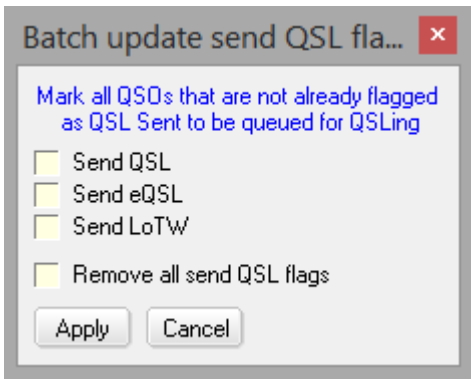
The top 3 options concern routinely QSL-flagging QSOs. For instance, select (tick) <**Flag new QSOs for LoTW**> to have Logger32 set (raise) the <**Send LoTW**> flag on every newly-logged QSO. When you are ready to sign and upload a batch of newly-logged QSOs to LoTW, you can then run the LoTW export function and pass the flagged QSOs through TQSL to LoTW, resetting the <**Send LoTW**> flag and recording the date in the process.



◀ Despite the misleading wording, <**Remove external Export QSL flags**> gives you the chance to enable (tick) *or* disable (untick) the sending of new QSO information to data files used by add-on utility programs such as L32 LogSync to update eQSL, Club Log and LoTW *automatically* as new QSOs are logged, one QSO at a time.

Copyright © 2026 Robert C. Furzer K4CY

Use <**Flag existing QSOs for QSLing**> to flag *your entire log* to be sent QSL cards[99], or to be uploaded to eQSL or LOTW, or to clear *all* the 'send QSL' flags ▶



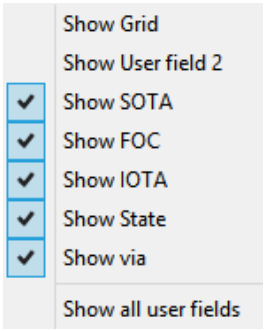
I collect Worked All Britain info using Logger32 awards functions plus the User 1, 2 and 3 fields. I have created Excel spreadsheets for each UK DXCC's WAB squares (Primary Admin) and also for other parts of the award (Secondary Admin) including Navigation Aids for Shipping (NATS), Triangulation Pillar (TP) references and UK ROC bunkers, plus the Worked All Ireland with Irish counties as Secondary Admins. For WAB I collect members' book numbers and UK islands using User 2 and 3 fields, changing the names accordingly. The spreadsheets plus a help document are saved to the Files area of the Logger32 reflector, along with a WAB Conversion Program to upload from the Logger32 ADIF files to the WAB award, making claiming the awards much easier.

Damian MOBKV

6.2.3 Show|Setup user fields

There are seven optional log entry fields available:

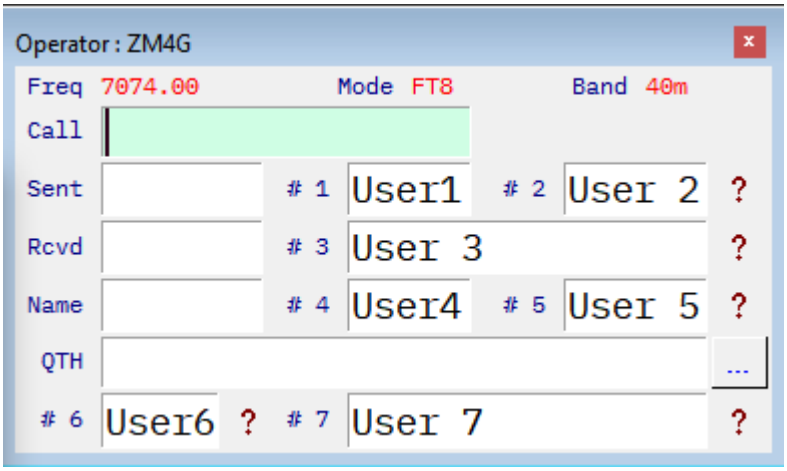
Right-click any log entry pane data-entry field then click **Setup ⇨ Show user fields** and tick whichever field/s you want to use ▶



[99] This is not eco-friendly. It *might* be appropriate to send cards 100% for relatively small, unique operations (such as special events commemorating something truly significant) where it is more likely that everyone worked would welcome a printed memento. Such situations are rare though. Save the planet!

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: if any of the optional user fields have been configured (see below), their names are shown in this menu *e.g.* until I configured field 3 for SOTA references and named the field "SOTA", the third option on this menu originally read <**Show User field 3**>.

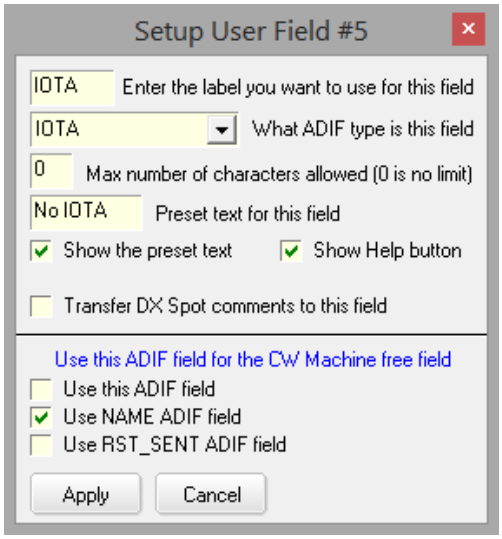


◀ Here is the log entry pane with *all seven* user fields and *all five* ? help buttons shown, with custom field labels and presets I have entered to show the layout for this image.

The log entry pane is a normal child window that can be resized by clicking and dragging out the corners. It cannot be moved beyond the edges of Logger32's main window.

The user field names, colors, fonts and font sizes are all user-configurable (read on for instructions). As I said, Logger32 is *highly* configurable!

Notice where the seven user fields appear in the log entry pane above. *None* of the fields can be repositioned within the log entry pane, unfortunately. However, each user field can be shown or not, and can be configured for any ADIF data:



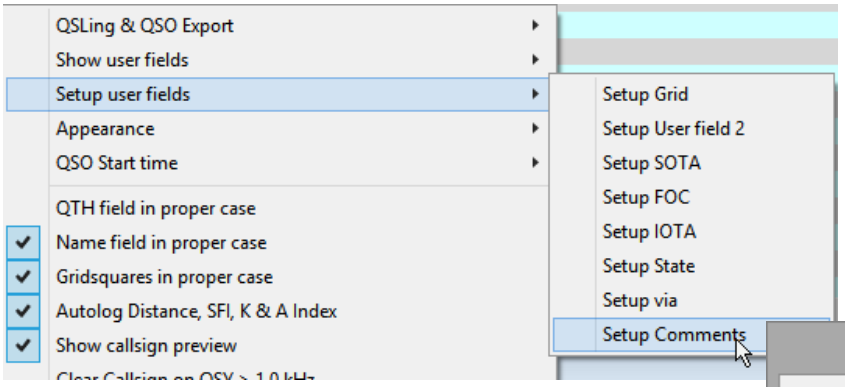
- First, you need to show the user fields you want to use on the log entry pane as noted above.
- To configure any visible user field, right-click any data entry field in the log entry pane, click **Setup ⇨ Setup user fields**, then click the field of your choice from the menu *e.g.* user field #5 opens a form like this ▶
- Type a short label (a field name) to be displayed just to the left of the field on the log entry pane.
- Click the down-arrow to open a drop-down list from which to select the field's ADIF data type[100].
- What is the maximum number of characters the field should accept? Use zero for free-text fields: entered text scrolls across once the data entry field is full of characters.
- Optionally specify some default 'preset' text for the field. If <**Show the preset text**> is also ticked, Logger32 pre-populates this field in the log entry pane as you are logging QSOs. Until you edit or delete the preset text or un-tick the option, it will also be recorded automatically in your logbook with all new QSOs that you log.

[100] ADIF fields that are already in use in the log entry pane are 'taken' and hence don't appear on the drop-down list. To move an ADIF field presently in use in a different user field, we must *first* remove it from the current user field, *then* you will be able to add it to another user field.

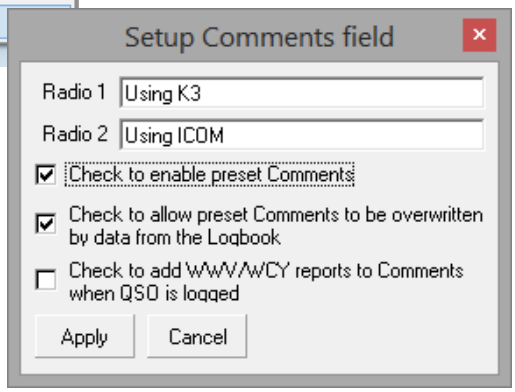
Copyright © 2026 Robert C. Furzer K4CY

- Optionally (if available[101]) tick the <**Show Help button**> to display a **?** in the log entry pane next to the user field. Clicking it brings up context-sensitive help on that field.
- Optionally, and only for ADIF fields USER_1|2|3, you can force the entered text to ALL CAPITALS.
- Optionally you can save DX spot comments in this user field automatically when you click commented spots.
- Click <**Apply**> to save the configuration for this user field.
- Repeat the process for any other user fields you wish to use.

There is further information on configuring the CW Machine free field.



◀ <**Setup Comments**> at the bottom of the <**Setup user fields**> configuration menu lets us type preset comments for either or both radios, with three options ▼



1. If we want the preset phrase to appear (and why else would we configure it?), select the first option.
2. If we want comments that were logged in previous QSOs with a station to overwrite the preset text the next time we work the same station (if Logger32 is configured to carry over prior comments), we select the middle option: otherwise the preset text will be added to all new QSOs.
3. The third option drops the most recent WWV/WCY solar and geomagnetic activity data received via DX cluster[102] into the <**Comments**> field when each QSO is logged.

In addition to that third option, solar/geomag info and calculated short-path distances can be recorded automatically in the log for new QSOs.

It is not necessary to have the SFI, distance, A and/or K index user fields visible on the log entry pane or the logbook, but if they are, we can type in values which will *not* be overwritten by Logger32.

[101] There is no room in the log entry pane for **?** buttons on user fields 1 and 4, apparently.

[102] Obviously enough, if no solar/geomag data have been received since Logger32 started up, they cannot be added to any field. Unlike ChatGPT, Logger32 does not hallucinate or randomly make things up.

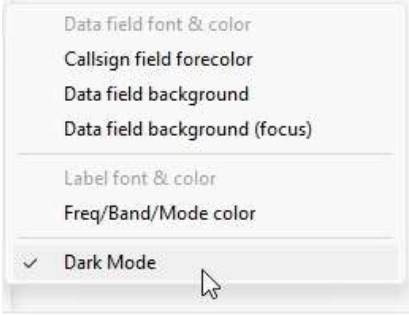
Copyright © 2026 Robert C. Furzer K4CY

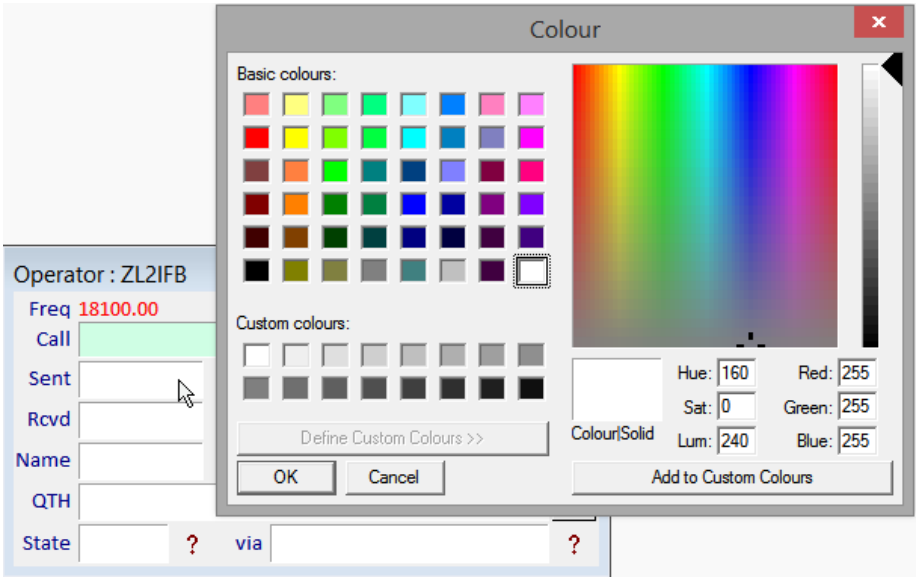
Logger is a great program. I have been using it for years. So, many thanks to Bob and the team for a great product. Great new manual as well. You do not generally get such a comprehensive manual with expensive professional software even.

Svein LA3PU

6.2.4 Appearance

This submenu of the log entry pane's right-click <**Setup**> menu lets us change the way various things appear in the log entry pane ▶





These are fairly obvious. Go ahead - adjust the font settings and try out various color combinations according to your individual preferences, monitor and eyesight.

◀ <**Data field background**> lets us specify the color of most log entry pane (and scratchpad) data entry fields.

<**Data field background (focus)**> sets the color of whichever data entry field has the focus *i.e.* the one we are currently writing or editing. Click another to shift the focus.

Hinson tip: personally, I prefer 'slashed zero' or 'dotted zero' *sans serif* monospaced fonts with crisp, clear typography for callsigns that can be ambiguous otherwise (*e.g.* given the right font, "FOØL" or "F0OL" is readily distinguished from "FØOL" or "F0OL" at-a-glance). You may need to hunt online for and maybe download and try out suitable fonts, such as *my* favorites: IBM Plex Mono and Monaco. If plain old Aerial is good enough for you, for now, then there's no need to change the default font. You can always come back here later to refine your setup.

Colors and fonts are equally important, *especially* if you are color-blind. If you anticipate DXing in the wee small hours, pay attention to contrast. Your bold choice of light orange text on a banana yellow background will come back to haunt you at some point, trust me.

Font size and **boldness** are important too, *especially* if your vision is impaired. Don't forget that you'll be using the log entry pane to record vital information about QSOs in progress,

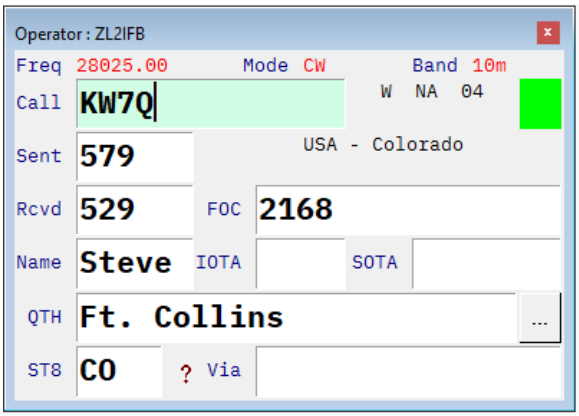
Copyright © 2026 Robert C. Furzer K4CY

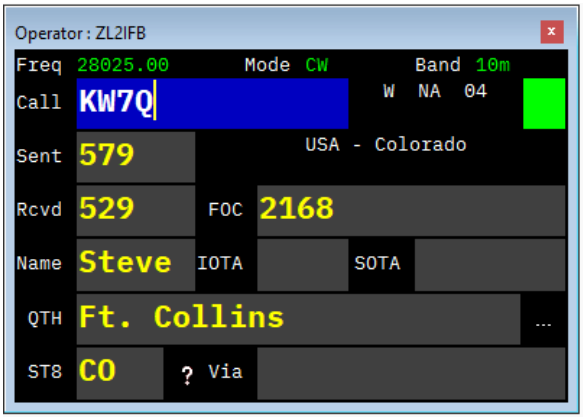
sometimes hurriedly. Can you read the details in the log entry pane reasonably reliably, ideally even *without* your glasses?

Hinson tip: having decided how the text appears on your log entry pane, you may need to adjust the height and width of the pane to accommodate all the info. The best way is to allow automatic window height adjustment, then mouse-over any edge or corner of the pane, then click and hold while you drag it to the right size (height *and* width), then release the click. It's easier to get this right first time if you have put typical text in all the fields.

Hinson tip: it takes time to use and become familiar with the new appearance after a change. Some changes will turn out OK, others grate on us until we revert to the original look or choose something different again. Once we find a combination of settings that suits us nicely – a stable configuration that we have used happily for at least a month or three – I recommend **archiving (stashing away somewhere safe) a copy of the *C:\Logger32\Logger32.INI* file** so that, if needs be, we can recover our preferred settings simply by restoring the archived *.INI* file to the *C:\Logger32* folder. A copy of the file in the same *C:\Logger32* folder with a different filename *may* suffice.

Dark Mode turns the usual ▼ light and airy log entry pane dark and foreboding ▼





6.2.5 QSO Start time

This is an interesting one.

QSO information is saved from the log entry pane into the logbook when we press <**Enter**> or <**Ctrl**+**L**> to **L**og it. The QSO **end** time is recorded automatically at that point.

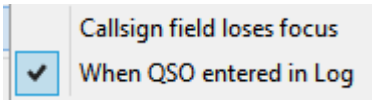
However, at what timepoint do we feel a QSO has *started*?

Hmmmmmmm.

Dwell on that thought for a moment before reading on …



Copyright © 2026 Robert C. Furzer K4CY



◀ This submenu of the log entry pane's right-click <**Setup>** menu gives us two options:

1. <**Callsign field loses focus**> records the time point at which we <**Tab**>, click or otherwise move the cursor away from the log entry pane's callsign field after entering a DX callsign. Typically, that is because we are going to check or enter another log entry field such as the received report … however, if we find ourselves patiently listening to or calling some exotic DX (FOØL for example), we may move away from the callsign field to check or pencil-in other information (such as her name) well before we have started an actual QSO. Given a callsign that exotic/rare and the associated pileup, we may be calling the DX for a significant amount of time before we get through. So when do we think FOØL will log our QSO: when we first moved away from the callsign field, or when she copied and called us?[103]
2. <**When QSO entered in Log**> is (for me at least) almost always shortly after I have completed a QSO. Having been calling for ages, when I finally break through the pileup and triumphantly work FOØL, I am likely to hit <**Enter**> to log the DX QSO soon after … which is hopefully about the same time that FOØL logs the QSO. We agree on that point, at least! Hopefully the QSO information we both logged will be a close match.

Either way, although our QSO **start** times may differ, we should both have logged more or less the same QSO **end** times – generally the time point at which we saved the QSO details to our logs. However, the QSO end time is *also* subject to differences of habit and opinion: in a net situation, for example, I may have hit return to log each of the net participants as they checked-in, *ages* before I bid them farewell and leave the net.

Hinson tip: if these discrepancies matter, while working a DX station we have been calling for ages, we can press <**Ctrl**+**S**> to set the QSO **s**tart time manually at that point, overriding the usual automated start time determination. Alternatively, if we display the start and end times for QSOs in the logbook, we can always click and edit the times recorded for logged QSOs.

Hinson tip: develop your own strategy. For most run-of-the-mill QSOs, the logged times are of little consequence but, for rare DX QSOs, timing discrepancies *could* result in the QSO details not matching, hence QSOs not being confirmed. For a keen DXer, after all the effort necessary to work a new one, that would be a disaster! Absolute precision is unnecessary but we should try to ensure that the QSO start and end times (both, ideally) record when we were actually in contact with each station in the log – not just calling in the pileup, tuning the linear, telling other ops to *QSY UP UP UP!* or doing a victory dance and lap around the shack.

[103] This is important for confirmations, especially if you work a 'new one' and *need* the QSL for awards. The ADIF standard allows both QSO START TIME and QSO END TIME values to be specified. Online logs, QSO matching and QSLing systems can use either, both or some other value to determine when a QSO took place. Most allow some leeway (*e.g.* 20-30 minutes).

Copyright © 2026 Robert C. Furzer K4CY

6.2.6 QTH field|Gridsquares in proper case

"Proper case" means that, by convention, the initial letter of each word or words comprising proper names (such as "London" or "Kuala Lumpur") is capitalized, with the remaining letters in lower case and a space between words.

For grid squares (Maidenhead locators), the convention is to capitalize only the first pair of letters. The second pair of letters (if given) use lower case *e.g.* RF80hx.

These configuration settings tell Logger32 to adjust the case accordingly, regardless of how the data are typed-in.

Hinson tip: a few place names do not follow convention, for example "DeKalb" is the official name for a city in Illinois. If we enable "QTH field in proper case", Logger32 will automatically adjust the city name to "Dekalb" as we type it into the log entry pane. Try as we might, we can't even correct the capitalization *after* it is logged with this setting enabled … unless if we temporarily disable the setting to enter or edit DeKalb in the log, and thankfully Logger32 will not 'correct' it in the log when we subsequently re-enable the setting. Phew! Case crisis averted![104]

6.2.7 Autolog Distance, SFI, K & A index

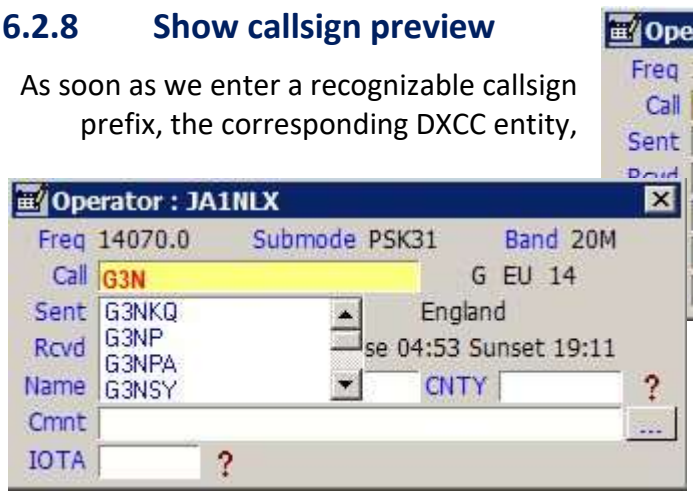
Right-click any field in the log entry pane, then click **Setup** and enable (tick) <**Autolog Distance, SFI, K & A Index>** to have Logger32 automatically calculate and record the short path distance from your station QTH to the contacted DX station, plus the latest solar and geomagnetic activity data – *provided* that:

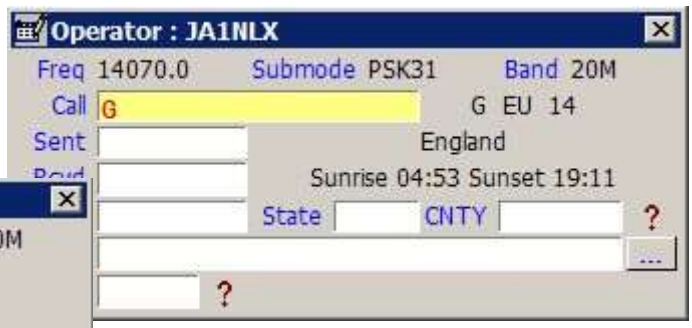
- You have correctly defined your QTH location for the current operator, from which the great circle distances are calculated.
- The WWV/WVY solar and geomagnetic data have been received through the DX cluster network since Logger32 was started.

Hinson tip: to ensure the solar and geomagnetic information is available to record as soon as you run Logger32 and start logging QSOs, set up a cluster login script telling the node to send the latest available data now, rather than just waiting patiently for the next scheduled circulation of the data on the DX cluster network, possibly a few hours later.

6.2.8 Show callsign preview

As soon as we enter a recognizable callsign prefix, the corresponding DXCC entity,





sunrise and sunset times, short- and long-path beam headings and short path

[104] Being a professional author and insufferable pedant, that's *A Big Deal* to me. You may not give two hoots however. Possibly none.

Copyright © 2026 Robert C. Furzer K4CY

distance *may* be displayed in the log entry pane *if* there is room ▶

◀ As we continue typing the callsign, any matching callsigns already in the log are listed by the **callsign preview** function. As we type each new character, the list of matching callsigns shortens accordingly. If the particular callsign we are working is listed in blue, we can stop typing and click it to pop it into the Call field.

Hinson tip: this is reminiscent of the **S**uper **C**heck **P**artial function beloved of contesters in N1MM+ and other contest loggers, only simpler, not quite so super. It searches for, and lists, previously-logged callsigns containing all the characters entered, as a unit. It finds some callsign variants (*e.g.* with /P, or with location modifiers such as /JD1 or SM/) but not all (*e.g.* typing the partial callsign TXF does not list G3TXF for me in blue, although I have worked and logged Nigel in many exotic places). It only searches in the open log, not in some database of known active hams.

6.2.9 Clear callsign on QSY

With this option on the log entry pane's right-click <**Setup>** menu ticked, data currently in the log entry pane will be cleared automatically if you tune the radio's VFO by more than +/- 1 kHz, ***unless*** the UDP BandMap is open[105].

The QSY frequency range or threshold can be changed by adding the following parameter to the [Globals] section of *C:\Logger32\Logger32.INI:*

[Globals]

Allowable QSY Frequency=3.5 … or whatever range you like, in kHz.

Any callsign lookup windows are closed when the Call edit box is cleared following a QSY, since there is no longer a callsign to look up.

Hinson tip: with this option enabled, if we neglect to log a QSO before tuning around looking for another, the log entry pane will be instantly cleared as the VFO spins past the threshold frequency. Provided we are paying attention to the screen we *might* notice it clearing and realize that the QSO wasn't logged … but will we still be able to recall the details in order to re-enter and log that QSO? Not if *your* short-term memory is as bad as … as … errr, umm, where was I?

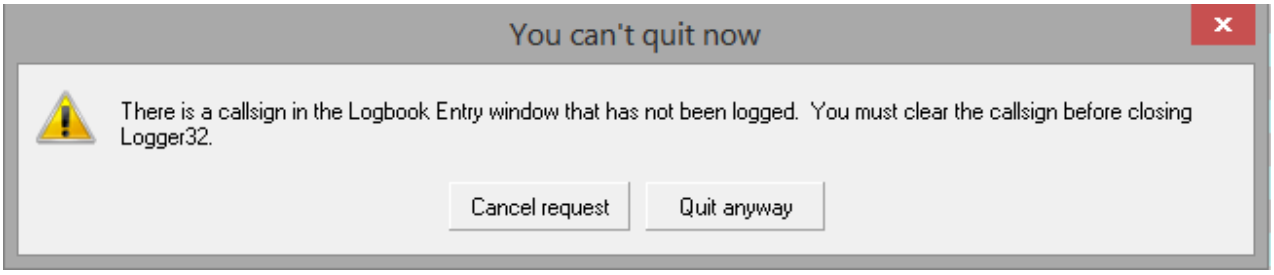
6.2.10 Allow program close with populated callsign

With this option on the log entry pane's right-click <**Setup>** menu selected, Logger32 *can* be closed immediately, even if you were apparently logging a QSO at the time. The QSO you are entering will *not* be logged, and will *not* reappear magically the next time you re-start Logger32.

[105] Split operating on quiet TX frequencies is the norm on digimodes such as FT#. The software can automatically QSY your TX in order to generate middling audio tones, reducing the probability of audio harmonics and wrong audio levels. You probably don't want Logger32 to wipe the current QSO just because of that!

Copyright © 2026 Robert C. Furzer K4CY

Do *not* select this option if you fear absentmindedly losing such QSOs. If unselected, Logger32 gives you a warning and a last chance to <**Cancel request**> (stop closing Logger32) in order to save the pending QSO to the log ▼



<**Quit anyway**> tells Logger32 to close *without* saving the pending QSO.

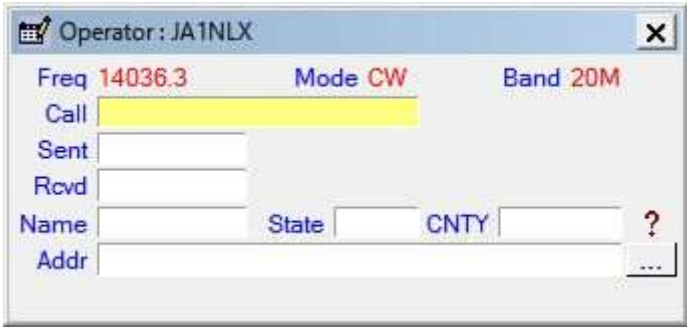
6.2.11 Use numeric pad +/- keys for BandMap navigation

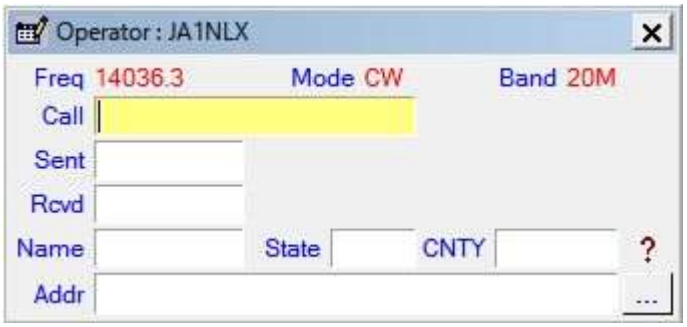
Enable this option in order to QSY to the next higher or lower frequency callsign spotted on the current band's BandMap with each tap on the + or – keys on the numeric keypad, respectively.

Hinson tip: if the prospect of working systematically through the spotted frequencies one at a time is not attractive, we can still click any spot on any BandMap, or in the DX cluster or DX Spots panes, to go directly to it; or enter a VFO frequency into the log entry pane's Call field; or even (yes!) stretch laboriously across the desk to twiddle the big knob and band switches on the radio, assuming the parts haven't long since seized-up through extended disuse.

6.2.12 Allow automatic window height adjustment

With user fields 6 and 7 *not* shown, and <**Allow automatic window height adjustment**> *not* selected, your log entry pane generally looks something like this, with a blank gray bar across the bottom wasting valuable screen space ▶





◀ If you enable the automatic adjustment option, the displayed log entry fields expand or contract both vertically *and* horizontally to fill the pane. They are magically re-sized whenever you adjust the size or shape of the log entry pane.

6.2.13 My QTH Lat/Long

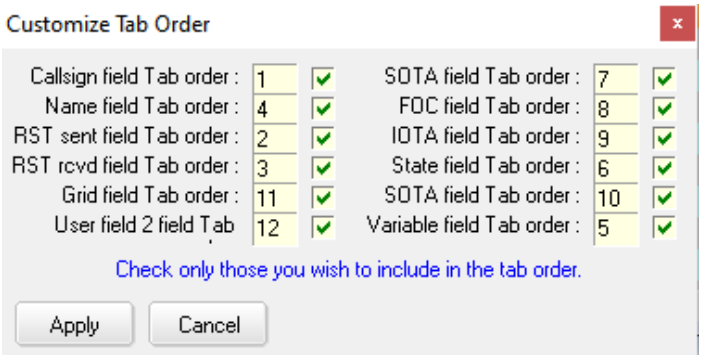
Read all about this important function in the installing Logger32 chapter. But, hey, you knew that already, having studied and followed the installation instructions to the letter, right? ☺

Copyright © 2026 Robert C. Furzer K4CY

6.2.14 Log entry pane field selector and tab order

After entering a callsign, we can <**Tab**> through the remaining log entry fields in a reconfigurable sequence: so, what sequence would we like?

Right-click the log entry pane, then click **Setup** ⇨ **Setup tab order** to configure the sequence.

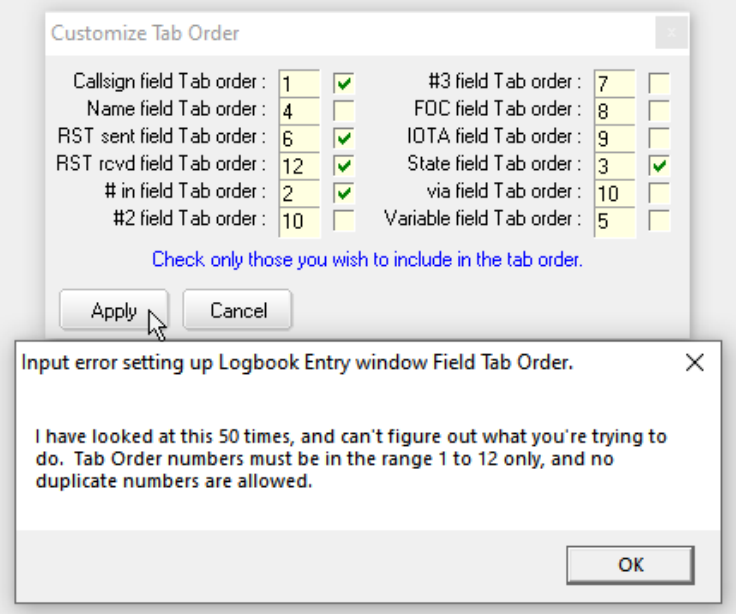


◀ In the rectangular boxes, enter the sequence number for each field. Check (*i.e.* tick) the fields to be included in the tab sequence: if un-ticked we can still access them by a mouse click.

Here, I can tab between all fields [if shown [106]], including my User fields for SOTA, FOC and IOTA references.

Having completed the selection, click <**Apply**> to have Logger32 check the information.

If we accidentally mess up the numbering (*e.g.* I set two fields to tab order 10), Logger32 expresses its confusion ▶



When we <**OK**> or close the error message, the tab order numbering changes magically to resolve the conflicts, but the resulting tab sequence may not be what we wanted, so check it out and adjust as necessary.

The Bands & Modes table (bandplan) may specify default sent and/or received reports for the mode in use, automatically filling the empty report fields as we simply <**Tab**> through them or hit <**Enter**> or <**Ctrl**+**L**> to **L**og the QSO. We can always overwrite the defaults with genuine reports.

Other fields may also be filled with information carried forward from previous QSOs with the same station, or from callsign database lookups, or with "Preset text" defined for the user fields. **These pre-entered/default values will be logged** as shown unless we edit them *before* hitting <**Enter**> or <**Ctrl**+**L**>, or edit the details later in the logbook.

Hinson tip: normally when DXing, I tab down each of the left fields in my log entry pane, systematically, then tab down each of the right panes. I find this sequence intuitive … except when casual contesting, when I prefer to tab from the callsign to the serial number received (ADIF SRX) field. *That* tab sequence, along with the relevant contest fields in the log entry pane, are saved in my "CONTEST" configuration.

[106] Whether these fields appear in the log entry pane, or not, is a separate issue.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: fields that are *excluded* from the tab sequence (un-ticked) can still be accessed, entered and edited with a mouse click. However, since we can tab without lifting a hand from the keyboard (and tab twice to skip past the next field), including them in the tab sequence is more ergonomic.

"I just wanted to say thanks for a great program. I just finished getting it configured to accept digital logs through my existing configuration. I like the fact it's so customizable. At first that seemed daunting but bouncing between the manual I was able to piece it all together. Back to configuring. Thanks for this great program."

73, Darin NY7H

6.2.15 QSO mask (carry forward information from previous QSOs)

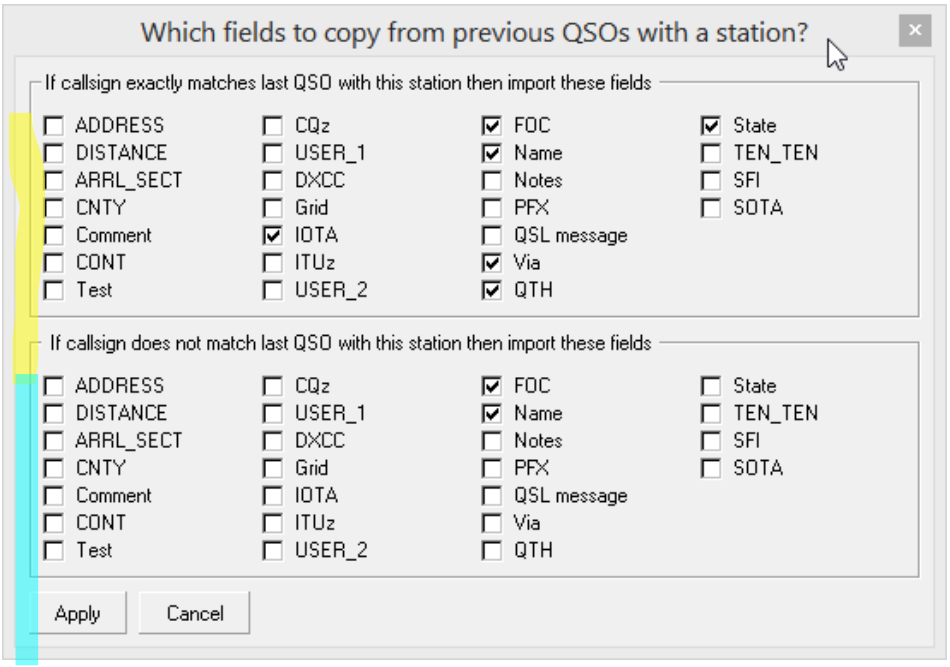
Logger32 can automatically look up and retrieve selected information from any *previously* logged QSO/s we had with someone, displaying it in the log entry pane when we *next* work them, making it easy to log it again this time around.

For example, if I had worked K4CY previously and logged his name in the name field of the logged QSO, then "Bob" will be retrieved and put in the name field of the log entry pane automatically the *next* time I am fortunate enough to be logging him.

If Bob decides he would rather be called Robert or Rob now, I can of course amend the content of the name field and log it accordingly this time, in which case his new name will be offered the *next* time I am logging K4CY. It's up to me whether to go back through my log and edit the names recorded for our previous QSOs, or not. Logger32 does not automatically apply such changes retrospectively: I retain control of my log.

Copyright © 2026 Robert C. Furzer K4CY

From the log entry pane's right-click <**Setup**> menu, open the <**Setup QSO Mask**> function then tick whichever fields we wish to be carried forward ▶



Notice there are two sections to this form:

- The upper section applies only if the station is using *exactly* the same callsign as before. The presumption is that the personal and locational information is probably going to remain the same. As mentioned above, we can always modify it when logging the QSO (including any carried-forward data), and the modified version will be logged. Or edit it later, no worries.
- The lower section applies if the station is using a *variant* of the callsign logged previously, for example 6Y5/K4CY or K4CY/M. Although the personal information is *probably* the same, the variant prefix/suffix suggests a different location … so we probably *don't* want to carry forward any locational information from the previous QSO.

Carried-forward information can be edited while or after logging the QSO. By default, though, the carried-forward information is simply saved with the new QSO when it is logged.

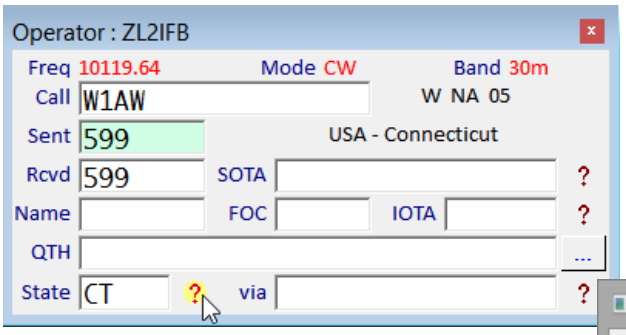
6.3 The ? (query or help) buttons

Clicking any ? (query or help) buttons in the log entry pane adjacent to some fields displays additional information relating to those fields.

Both the user fields and the ? buttons are optional so none are shown if none are configured.

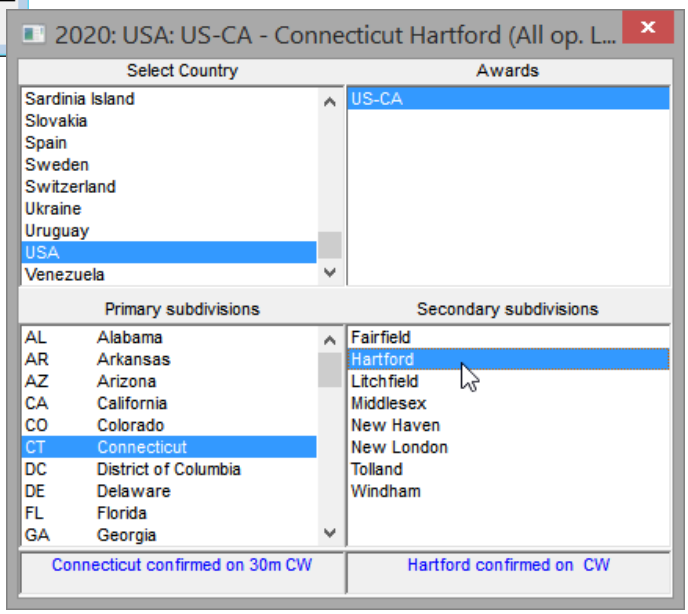
Copyright © 2026 Robert C. Furzer K4CY

6.3.1 State query



◀ While logging a QSO, the <**State**> field may be filled for us (following an online callsign-lookup or carried forward from a previously-logged QSO with that station), or we may type in the state abbreviation if the person we are working tells us which state they are in[107].

To enter the person's county, click the adjacent ? to open a form on which to select the primary (State) and secondary (County) subdivisions ▶

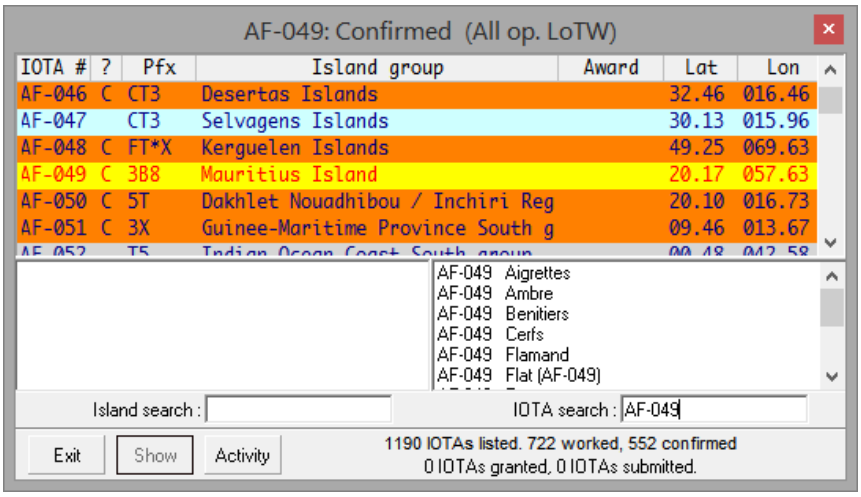


The caption identifies relevant awards for which this QSO may qualify (the USA Counties Award in this example).

At the bottom of the form, Logger32 tells us if this is a new one or not, once we have selected the subdivision.

Alternatively, *right*-clicking the ? opens a report showing QSOs with that state from the open logbook, but only those logged in the current year *if* the year is shown in the top left corner of the worked/confirmed table, rather than "ALL".

6.3.2 IOTA query

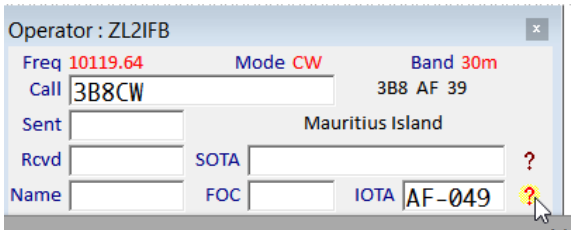


◀ A similar database form is shown if you click the ? next to the IOTA field (if both are shown on your log entry pane).

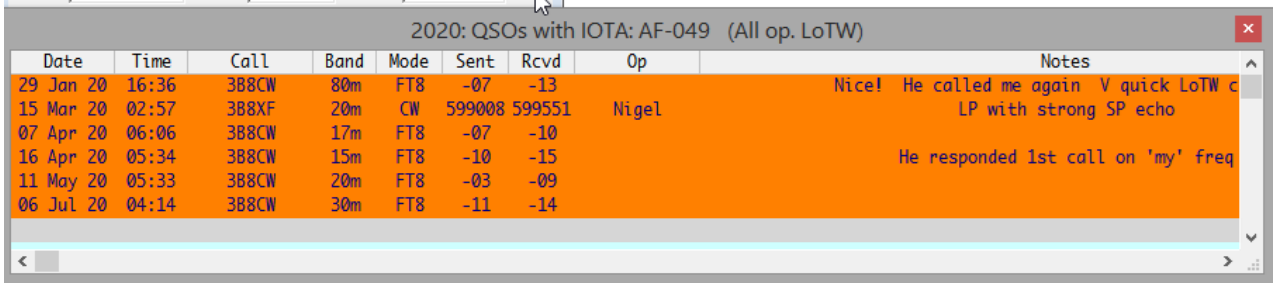
[107] "Disarray" or "Confusion" are my usual states. There are no states within New Zealand.

Copyright © 2026 Robert C. Furzer K4CY

In that example, I was logging a QSO with Clive 3B8CW and had put AF-049 in the IOTA field on my log entry pane. Having clicked the ?, the IOTA table opened at Mauritius Island displaying information about all islands within the AF-049 group, plus a summary of my IOTA awards status at the bottom.



Right-clicking the ? next to IOTA opens a report showing QSOs with that IOTA island reference from your log ▼



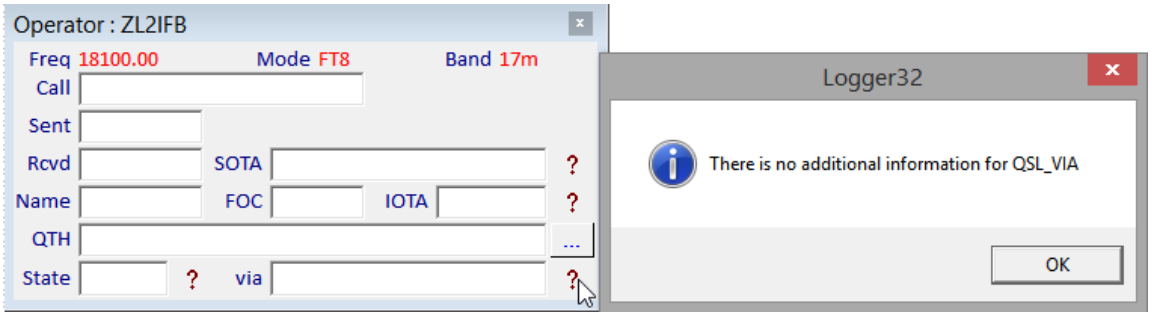
6.3.3 Grid queries

With the gridsquare entry field showing in the log entry pane, clicking the query button lets us check for QSOs with any 4 or 6 character locator, either in this year only or all-time ▶



6.3.4 Other queries

Log entry pane fields 2, 3, 5, 6 and 7 have the setup option to show a ? query/help button … but that is pointless if there are no additional information for Logger32 to look up internally for those fields, QSL_VIA for instance ▼



Hinson tip: using the user field setup function, hide the ? buttons except for the applicable fields where they work. It reduces clutter and confusion.

Copyright © 2026 Robert C. Furzer K4CY

6.3.5 Logging satellite QSOs

Satellite users can quickly select a different SAT_NAME when logging satellite QSOs:

- Setup the SAT_NAME field of the Log entry pane to show the **?** query/help button.
- Put your favorite satellite as the preset text and tick <**Show preset text**>.
- Now, with a text editor, make a list of the satellites you operate and save it as *C:\Logger32\SAT_NAME Picklist.txt.*
- Close and re-start Logger32.
- Thanks to the preset text, the Logbook entry pane shows your favorite satellite name in the SAT_NAME field.
- To change to another satellite, click the **?** query/help button. A new pane opens showing your list of satellites. Choose and click one. The Logbook entry pane's SAT_NAME changes accordingly. Next time you start Logger32, that will be the new default SAT_NAME … unless you change it again.
- If you have been using satellites and decide to operate HF for a while, click the**?** query/help button by the SAT_NAME field then click the empty top to clear the SAT_NAME and PROP_MODE fields in the Logbook entry pane, ready for non-satellite logging.
- Once you have mastered the basics, feel free to add radio macros to your *SAT_NAME Picklist.txt* file. For example, for a satellite named 'SAT 10', you want to send the macro $**command hop;skip;and;jump;**$ when you start using it, so:
  - Save the macro $**command hop;skip;and;jump;**$ somewhere on the Radio Control Panel. Make a note of its button number[108].
  - Say your macro is on button 36 at the righthand end of row 3. Modify the SAT 10 entry in your *C:\Logger32\SAT_NAME Picklist.txt file* to SAT 10 , 36 (spaces around the comma are optional). Now the next time you select SAT 10, the radio macro will run automatically[109].
- Under the Config menu of the Pick SAT_NAME window there are two options.
- Use <Enable radio setup macros> to enable/disable the radio hop, skip and jump when selecting SAT 10 (from the example above).
- Use <Send changes to the Tracking window> to change to the selected satellite in the Tracking window (if it is open).

6.4 Logbook informational entries (pseudo-QSOs)

Information other than genuine QSOs can be logged as pseudo-contacts by prepending or appending "=" (the equals sign, *without* the quotes) to the callsign in the Call field of the log entry pane or the logbook *e.g.* =K4CY, CQING=, =MARS, D1B=, =WAR, ZL6B= or =PWR_TEST.

Such informational entries become part of your log almost like real QSOs. They can be deleted and edited, and they can optionally be included or excluded if you export your log. However, since they are not actual QSOs, the awards statistics are *not* updated and they are *not* flagged to send any form of QSL or upload to online logs such as QRZ and Club Log. They needn't even be actual

[108] Macro buttons are numbered sequentially left to right by rows *i.e*. 1-12, 13-24, 25-36 and 37-48.

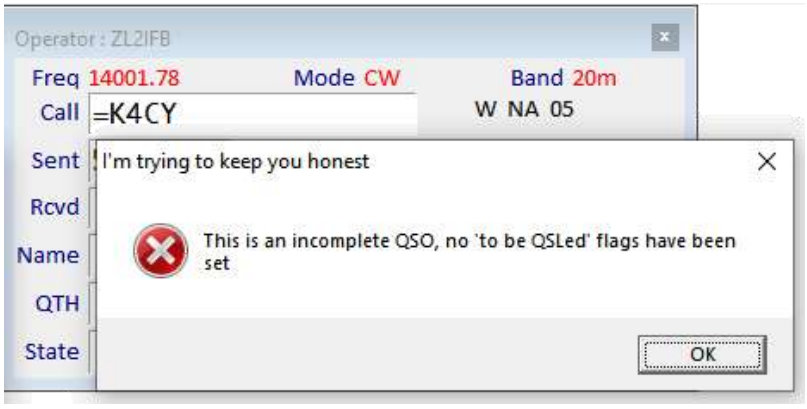
[109] Unless you are using FT2/4/8 where the digimode software – not Logger32 – has CAT control of the radio.

Copyright © 2026 Robert C. Furzer K4CY

callsigns, although they must be callsign-like to be accepted into the field *i.e.* contiguous character strings, all CAPITALS without spaces (UNDERSCORES_ARE_OK).

In conjunction with the previous QSOs pane, informational entries can be useful *e.g.*:

- You can record long periods of calling CQ, for instance to keep track of when you were on the air, perhaps to deflect spurious TVI complaints from your neighbors. Simply enter "CQ=" in the Callsign field of the log entry pane, along with your frequency, output power, antenna used, time on and time off, or any other information you wish to retain in other fields.
- You may want to record a contact with a station you suspect to have been a pirate (unlicensed). If you're proved wrong and eventually a valid confirmation comes through, so much the better: simply edit the record in your log, removing the equals sign to leave a regular QSO. In the meantime, your award statistics will not reflect the QSO, so if it would have been a new one for you, Logger32 will continue highlighting any other new ones that come along, including spots for the same DX station on the same band/mode.
- To log an incomplete QSO simply as a reminder that you tried but failed to contact him/her.
- To record information about a station before you make a QSO. It is not uncommon, especially on the low bands, to try for many nights to work a DXpedition before finally making it. Keen DXers hope to improve their chances by recording the exact times of openings, frequencies used etc. Conversely, casual operators may prefer not to "waste their time" calling the DX at the same time and band in future if they were unable to get through.
- Logging beacons or other interesting DX stations heard but not worked.
- Keeping records of changes to your station or antenna farm. Simply enter your own callsign followed by the =, and add notes to the comments or other fields as you wish. Alternatively, use entries similar to ANT= or STN= or QTH= to save relevant notes to your log, with the advantage that you can bring up all the notes of each type in the Previous QSOs pane when you type the "callsign" into the log entry pane's Call field.
- If you are unsure of the callsign of someone you have worked (perhaps because of QRM or QSB, really bad CW, poor modulation or a thick accent), or if you are not sure a QSO was completed, you can record the *possible* QSO as an informational entry pending some form of confirmation (*e.g.* if a DX station shows your QSO in an online log[110]), then remove the equals sign.
- When you hit <**Enter**> to log an informational QSO, Logger32 reminds you that it is 'incomplete' and so will *not* be flagged for QSLing ▶



[110] You won't get a match on LoTW though, unless/until you sign and upload a genuine QSO *without* the equals sign. Likewise with other online logs. Ethically speaking, you *should not* do that unless/until you honestly believe you have completed a legitimate QSO.

Copyright © 2026 Robert C. Furzer K4CY

Informational entries are *different* to any <**Not accepted for DXCC**> QSOs in your log. The latter are completed QSOs that, for some reason, are or would not be accepted for DXCC or other awards administered by the ARRL DXCC desk *e.g.* maritime mobiles, pirates or DX operators who are unable or have chosen not to submit adequate evidence of their licenses and permits to ARRL[111].

Hinson tip: since informational entries/pseudo-QSOs are unique to Logger32, other logging programs and services (such as Club Log and LoTW) may either ignore them, convert them to regular QSOs or flag errors if they are included in ADIF or QSL file outputs from Logger32 … which is why you have the *option* to skip them on the export routines. If you are simply backing-up your log, or plan on transferring it to a logging program or service that does not flag an error or implode on finding '=' within a callsign field, I recommend *not* skipping them on export to reduce the risk of losing them forever.

Because the = can be typed with one finger, tapping the key adjacent to the <**Backspace**> on most keyboards, it is easy to alternate the contents of the log entry pane Call field between, say, "FT5XO" and "FT5XO=". Doing this causes the contents of the previous QSOs pane to flip instantly back and forth between real, completed QSOs with FT5XO and information entries for attempted but incomplete or failed QSOs with FT5XO.



To illustrate how you might use this facility in practice, here is a section of W2RU's log for the last week of the FT5XO DXpedition ►

Bud has typed FT5XO into the Call field of the log entry pane whereupon the previous QSOs pane lists all the contacts W2RU made with FT5XO through March 26th. Notice that the drop-down list beneath the Callsign field in the log entry pane has "FT5XO=", indicating that W2RU's logbook contains one or more information entries for FT5XO …



◄ Now Bud added = after the callsign in the log entry pane's Call field.

The previous QSOs pane now shows *only* informational entries *i.e.* incomplete QSOs from W2RU's log. Bud can see what hours FT5XO was audible on topband at his QTH on previous days and which receiving antennas worked best.

[111] These QSOs *may* be eligible for *other* awards tracked by Logger32, and you might prefer to keep them in your log out of interest. I have worked two EZ stations, for instance, that I believe were both in Turkmenistan but not officially licensed, so I still *need* EZ for DXCC.

Copyright © 2026 Robert C. Furzer K4CY

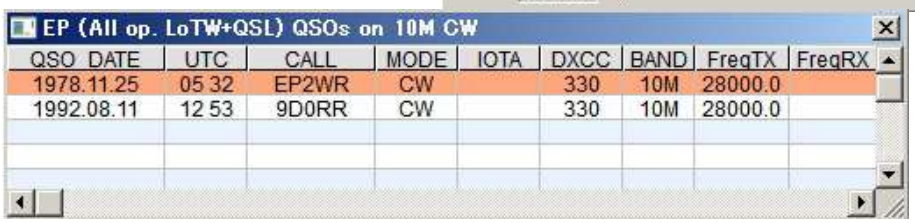
6.5 Database maintenance

6.5.1 Worked/confirmed table examples

Here Aki is logging a QSO with EP6T in the log entry pane, with the worked/confirmed table displayed alongside ▶

When Aki clicks, say, the 10m/CW cell of the worked/confirmed table …

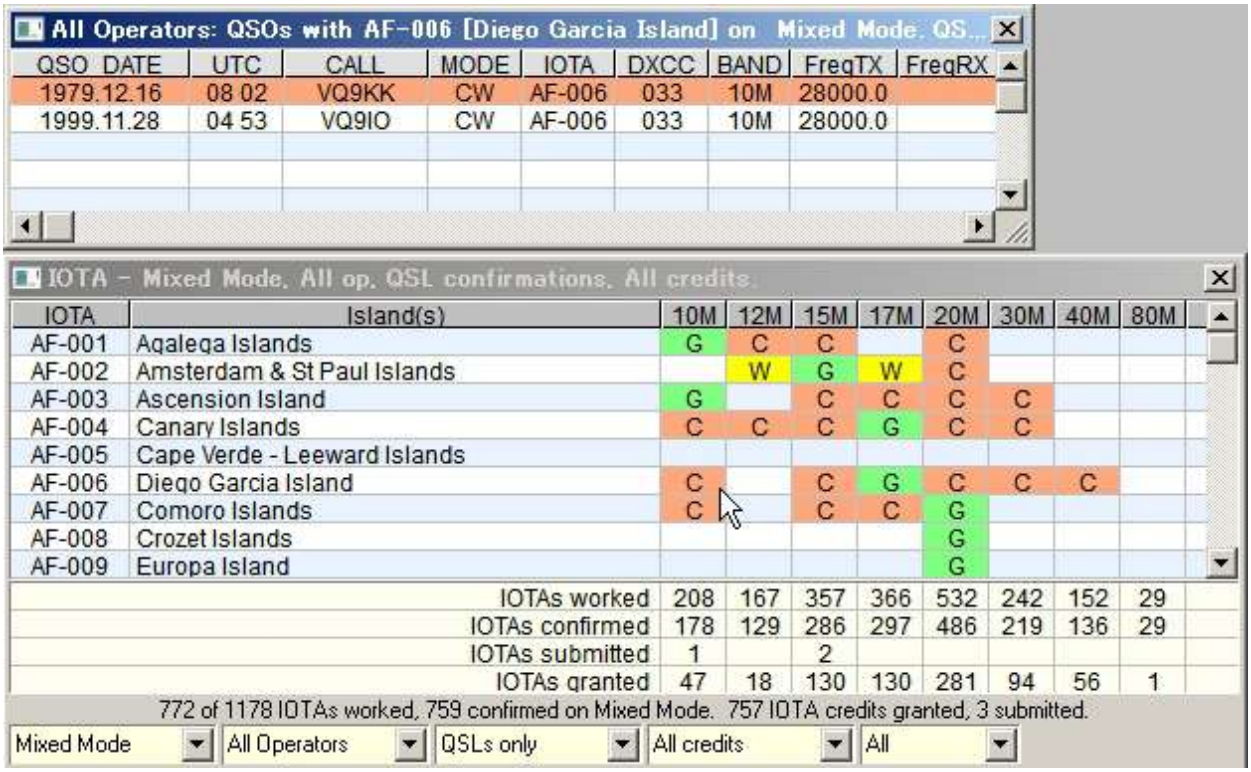




◀ *all*[112] his logged QSOs with EP on 10m/CW (not just EP6T) are shown in the previous QSOs pane.

6.5.2 Award tables examples

This example shows what is displayed in the generic QSOs pane when Aki clicked the AF-006/10m cell in the IOTA Awards table ▼



Aki saw all his QSOs with IOTA AF-006 on the 10m band.

[112] Correction: he sees all the relevant QSOs only if the top left corner cell of the worked/confirmed pane currently showed "ALL". If instead it showed the current year (*e.g.* "2025"), Aki would only see EP QSOs logged so far this UTC calendar year … and shouldn't be too surprised if there are none. Yet.

Copyright © 2026 Robert C. Furzer K4CY

6.6 Callsign and country information floater

The 'floating callsign' replicates the callsign currently in the log entry pane's Call field, and (optionally) the country name, local time and headings for the station being worked. Once popped-up, the little window can be positioned anywhere on the screen, wherever you would like it to appear the next time it appears *e.g.* near the top of the Sound card data window or CW Machine for quick reference.

Hinson tip: the callsign floater can be especially useful if you have multiple monitors, perhaps running digimode software on a side screen separate from the main screen with the log entry pane.

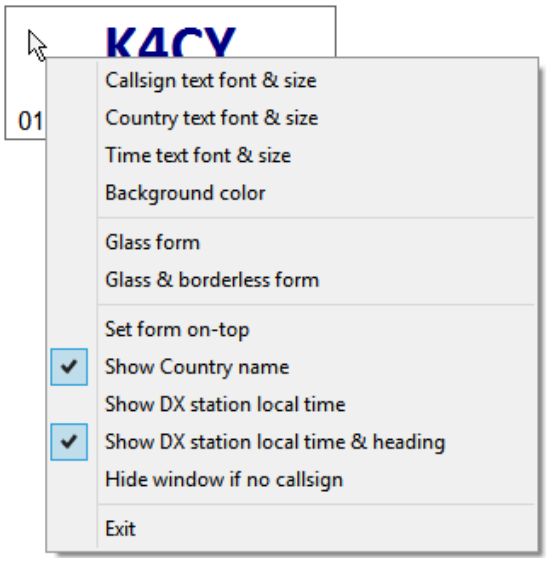
To open the **F**loating callsign window, click to focus anywhere in the log entry pane and hit <**Ctrl**+**F**>, or right-click the pane then click <**Floating callsign field**>.[113] Here it is configured to display all the available fields while I am logging a rare QSO with K4CY ▶



The fields shown in the example are:

- The callsign of the DX station I am in the process of logging.
- Bob's DXCC entity and primary administrative subdivision (state) where applicable and known.
- The date and time at the DX end, as shown on Bob's shack clock and calendar.
- The short path and complementary long path azimuth directions for me to beam at him.

Right-click the floating callsign window[114] for the configuration options ▶



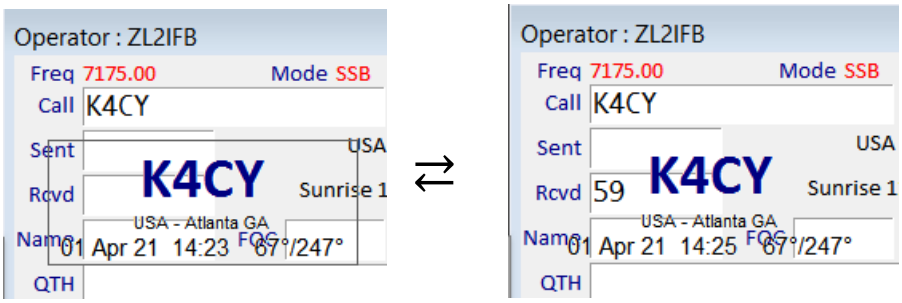
- **Callsign|Country|Time text font & size**: how would you like the text to be shown? The floating window's height is adjusted automatically according to the chosen font size. Its width can be adjusted by dragging the left or right sides with the mouse.
- **Background color**: select the desired color of the floating callsign box in the normal display mode (not glass …).

[113] Depending on various options, if there is no callsign in the log entry pane's Call field, the floating callsign will probably be mostly blank, may be completely transparent/invisible as a borderless glass window, and may not be shown at all if it is outside the screen boundary. Type a callsign into the Call field and hit <**Ctrl+F**> once or twice to **F**orce it to display. If it *still* isn't visible somewhere on your displays, use the **View ⇨ Find lost windows** function to move it into view at the top left of the primary monitor.

[114] Since there is no background to click with the glass option selected, you need to right-click *the text* to bring up the configuration menu … hence there has to be some text displayed, so put any call into the log entry pane. You can also move the invisible box by clicking and dragging *the text* into position. This is easier if you choose a large **bold** font, which makes sense since the main purpose of this function is to make it **obvious** who you are working.

Copyright © 2026 Robert C. Furzer K4CY

- **Glass form** makes the window background transparent, displaying the text (if any) with a border but no background color ▼, floating over whatever is on the screen.



- **Glass & borderless form**: shows the text (if any) without outlining the window area ▲ To figure out where the floater is, put any callsign in the log entry pane Call field and look for it … and if you still can't find it, press <**Ctrl**+**F**> twice with the focus on the log entry pane to disable glass, giving the window a visible background.
- **Set form on-top** stops other windows obscuring the information.
- **Show Country name|DX station local time|DX station local time & heading** let you choose to display the additional information besides the callsign, or not.
- **Hide window if no callsign** makes the floating callsign window disappear when the Call field in the log entry pane is cleared e.g. after you log a QSO or hit <**Alt+W**>.
- **Exit** closes the floating callsign window … since it doesn't have the usual corner X to click.

6.7 Notification messages

Logger32 displays notifications briefly in a yellow box at the bottom right corner of the screen when:

- Your system checks for Logger32 program updates, whether automatically (soon after Logger32 is started up) or manually triggered. It tells you if an update is found.
- Your system checks Club Log for updates to the prefix and callsign exceptions. It displays the progress if an update is found and downloaded ▶



- Your system clock is synchronized to an atomic clock automatically using the <**Sync clock when Logger32 starts**> and/or <**Sync clock when opening the UDP BandMap**> options in Get atomic clock time.
- A just-logged QSO is signed and sent to LoTW using <**Real time upload to LoTW**> in the L32 LogSync utility ▶



- You recalculate your statistics ▼

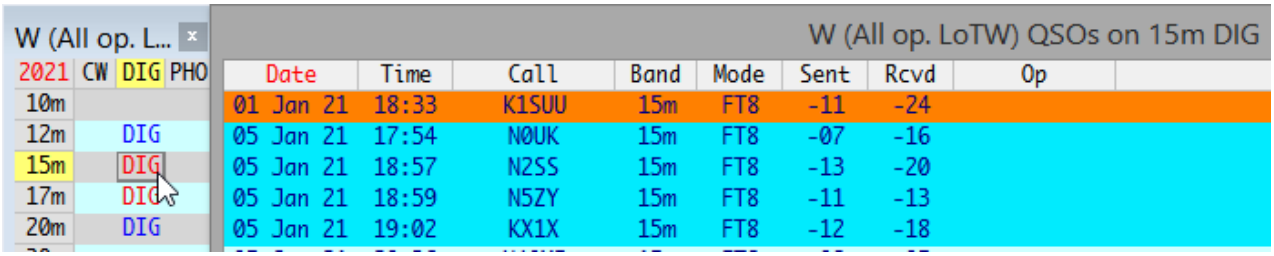


- You <**Load the LoTW users file**> or <**Load the LoTW users file (from Club Log)**> with information about callsigns known to have used LoTW.

Copyright © 2026 Robert C. Furzer K4CY

6.8 Generic QSO pane

The generic QSO pane is not available from the <**View**> menu or toolbar icons but magically appears when you click any cell in the worked/confirmed table or any of the award windows, showing the applicable QSOs ▼



'Applicable' means you are shown a *filtered* set of QSOs in the current logbook that qualify for whichever cell, row or column you clicked:

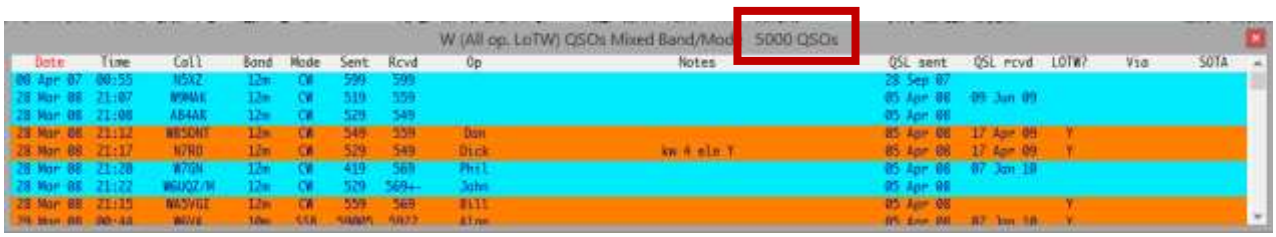
- If you clicked a populated cell, Logger32 only shows QSOs that qualify for the same cell (e.g. the same band and mode).
- If you clicked an empty cell, Logger32 shows all your QSOs logged with that entity, regardless of the bands and modes.
- It either shows QSOs logged at any time, or only those logged during the current calendar year if the top left corner cell of the worked/confirmed table shows the year (click to toggle between ALL ⇄ 2026).

The window's caption tells you the filtering in effect to select the specific QSOs shown.

In the example above, with a US callsign selected from my logbook and the current year in the top left corner of the worked/confirmed table, I clicked the red "DIG" entry at the intersection of the DIG column and 15m row, so the generic QSOs window listed my W (mainland USA) digimode QSOs on 15m so far this year. The orange background told me the top QSO on the list has been confirmed, which is why that "DIG" cell in my worked/confirmed table shows red text.

The window caption says "(All op. LoTW)" meaning that it lists all applicable QSOs in the current logbook regardless of the Operator (my station callsign), and any QSO confirmed on **L**ogbook **o**f **T**he **W**orld (specifically) are highlighted.

If there are more applicable QSOs to show than there are rows available given the current height of the window, the total QSO count (up to 5,000!) is stated in the caption, as with the previous QSOs pane ▼



6.8.1 Setup generic QSO pane

As with the logbook, there are three ways to configure the columns and text …

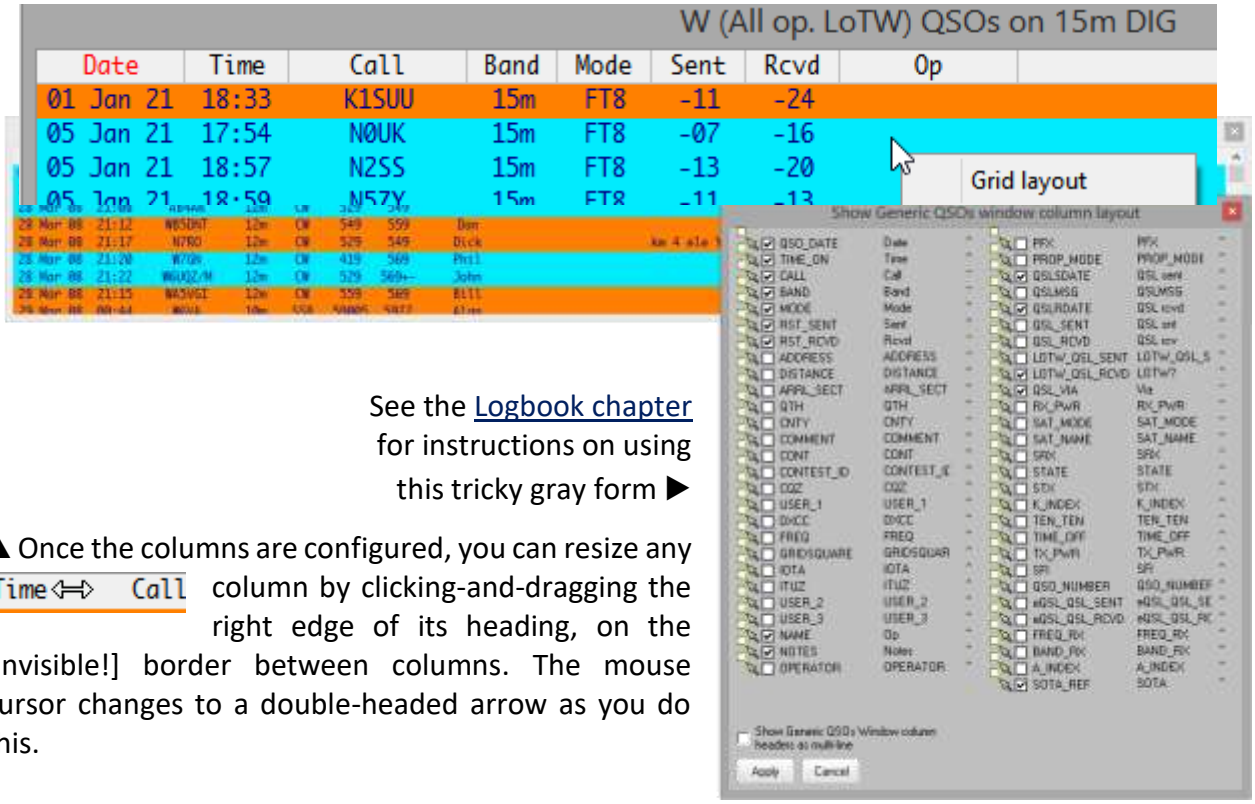
1. Right-click any **column heading** to access the configuration options for *that* column ▼



   After selecting the ADIF field you wish to show[115] in *that* column, editing its heading text (optionally making the headings multi-line rather than single-line) and choosing whether to left, center or right-justify text in the column, click <**Apply changes to this column**> to save any changes and update the displayed window. See how it looks before moving on. There are buttons to select adjacent columns if you want to update them too, or to shift the current column to the left or right.

   Having applied any changes, <**Exit**> the form when you are happy with your layout[116].

2. Right-click anywhere in the **main body** of the generic QSOs pane, then click <**Grid layout**> to configure the layout options ▼



See the Logbook chapter for instructions on using this tricky gray form ▶

▲ Once the columns are configured, you can resize any

column by clicking-and-dragging the right edge of its heading, on the [invisible!] border between columns. The mouse cursor changes to a double-headed arrow as you do this.

[115] Or hide by clicking to un-tick <**Column is visible**>.

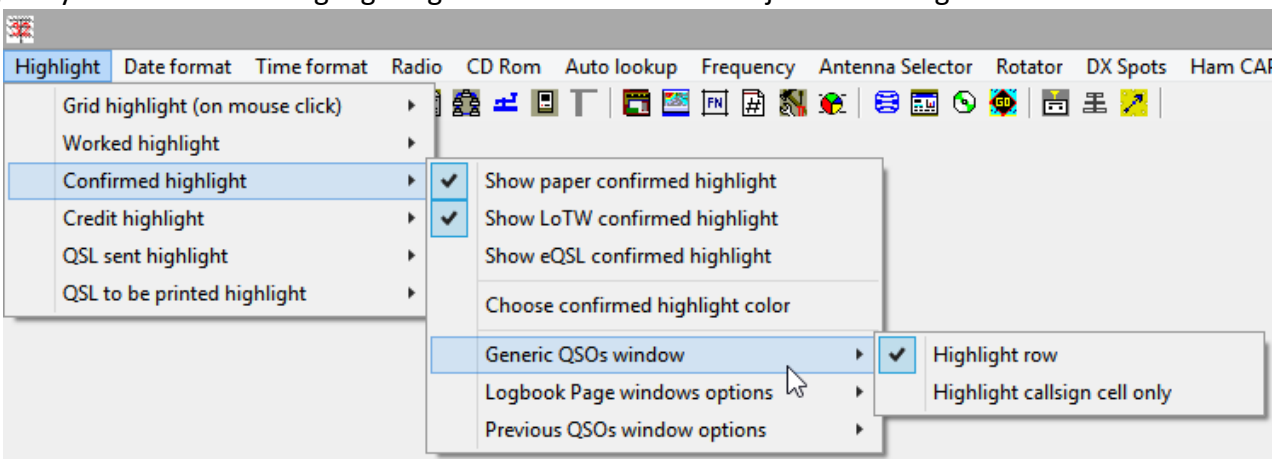
[116] If you <**Exit**> the form *without* clicking the <**Apply…**> button, any changes you just made are summarily and callously discarded.

Copyright © 2026 Robert C. Furzer K4CY

Click any heading to sort the table by that column, turning the heading text red.

Hinson tip: showing and sorting by the QSL_VIA column lets you see a selection of QSOs with DX stations using a given QSL manager. To save postage, you might prepare a batch of QSL cards for several DX stations to be sent to the same QSL manager.

To avoid confusion, generic QSOs have the same background color to highlight confirmed QSOs as elsewhere in Logger32[117]. However, **Setup ⇨ Highlight ⇨ Confirmed highlight ⇨ Generic QSOs** gives you the choice of highlighting either the whole row or just the callsign field ▼

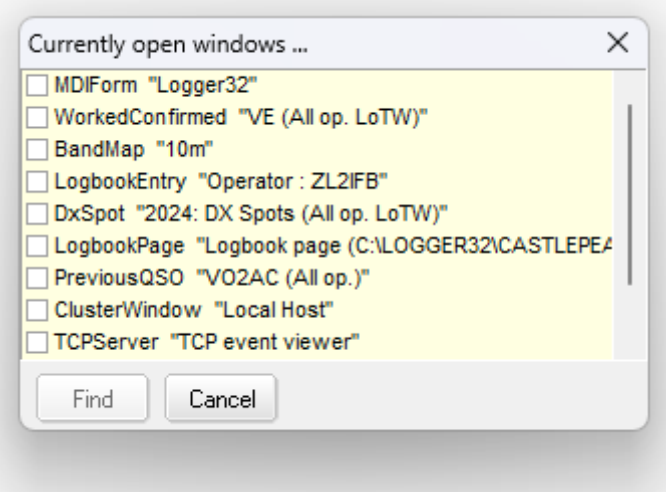


3. **Generic QSO text appearance**: to avoid utter confusion, generic QSOs are shown in the same text font, color and size as the logbook and previous QSOs. If you want to change any of them, reconfigure *all three* simultaneously.

6.9 Find lost windows

Sometimes, Logger32's windows[118] get 'lost', mysteriously disappearing partially or wholly even though they may be open according to the <**View**> menu. They may be hidden beneath other windows, totally transparent, or perhaps (for some spooky reason) they have slid completely off-screen, beyond your monitor's edges.

View ⇨ Find lost windows shows a ***scrollable*** list[119] of Logger32's open windows ▶



Click to tick any invisible one (!), then click <**Find**>, whereupon it magically reappears on top up in the top left-hand corner of the (primary) display, ready to be dragged

[117] You might like bright, even garish colors: personally, I prefer subtler, muted, pastel shades more in keeping with the Art Deco city of Napier where I live.

[118] 'Find lost windows' is technically a misnomer: it mostly looks for lost panes, that is lost *child* windows. Some non-child windows (such as the CW Machine) are not listed.

[119] The list is sorted alphabetically. Scroll down if you are looking for something later in the alphabet.

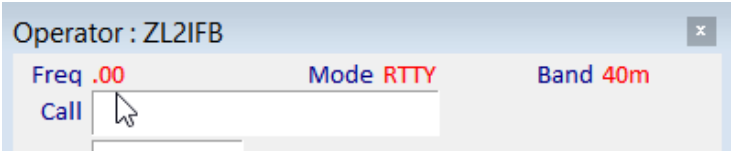
Copyright © 2026 Robert C. Furzer K4CY

screaming and kicking wherever you want it to be. Repeat as necessary for any other open-but-invisible Logger32 windows.

Hinson tip: **View ⇨ Find lost windows** *only* lists Logger32 windows that are currently open. Don't bother trying to find a window that is not currently open: it doesn't exist at that point.

6.10 Log entry pane FAQs

Q. What happened to my VFO frequency?? ▶



A. The log entry pane's frequency display reverts to .00 in red if, for some reason, Logger32 loses or drops its CAT connection to your radio.

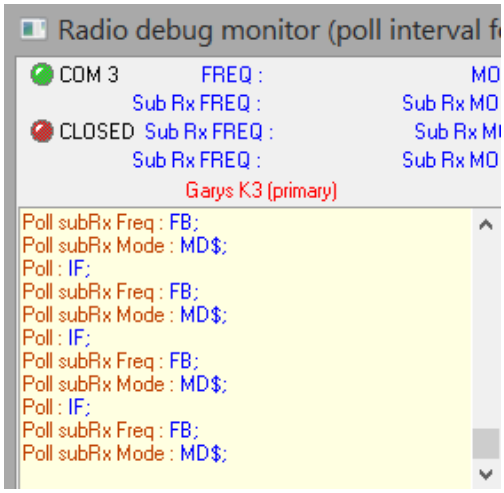
First confirm that your radio is powered on and working (yes, really!).

Next, check the <**Radio 1|2**> panel at the bottom of Logger32's main screen. If the text is red, right-click the panel and click <**Open port [port name]**> start polling the radio for its status information. That should turn the <**Radio 1|2**> panel text blue *and* show the VFO frequency in the log entry pane. Conversely if the panel was blue already, right-click it, close the port and then open it again – basically restart the CAT connection.

If the CAT port is not available, possibly other radio software – such as JTDX, WSJT-X or N1MM+ – has control of the CAT port, preventing Logger32 from opening it[120]. If so, once you have closed that other software or told it to release the port, *then* you can open it in Logger32.

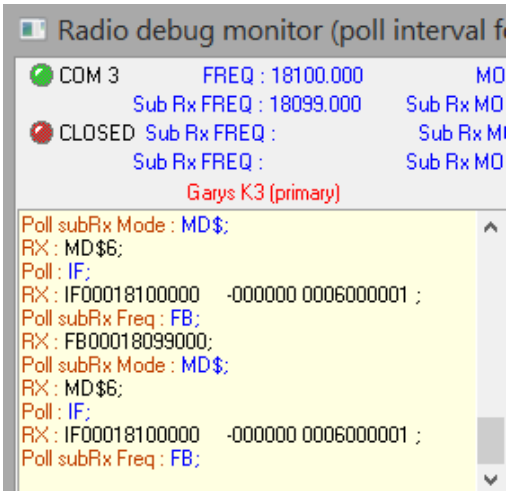
A loose USB-RS232 converter dangling from the radio's CAT port is the usual culprit in my case. When disconnected, the radio debug window shows only the red and blue text messages from Logger32 with no black text responses from the radio ▼

USB converter *dis*connected or *non*-functional



⇄

USB converter connected and working normally



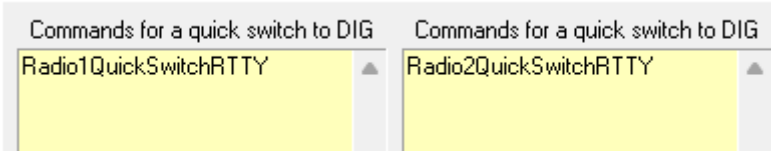
[120] PC ports are evidently tiny, such that only one ship may dock at a time.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why doesn't a log entry pane query/help button work?

A. Firstly, only user-defined fields #2, 3, 5, 6 & 7 have the screen space to show **?** buttons next to the data entry boxes. Secondly, only some user-defined field types have additional information to look up when you click the **?**. If, for example, you are defining a user field for IOTA, be sure to use one of the user-defined field numbers listed here *and* tick the associated <**Show Help Button**> option.

Q. What are "Radio1QuickSwitchRTTY" and "Radio2QuickSwitchRTTY"?



A. They are placeholders, showing where you can trigger macros when Quick-Switching to digital modes on radios 1 (the left yellow panel) or 2 (right panel).

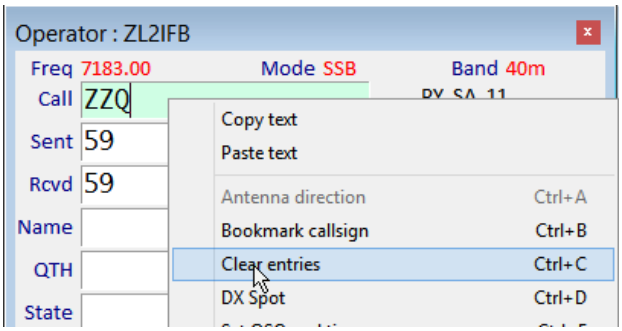
For example, with Quick Switch enabled, when you click a RTTY spot having typed **RTTY=RadioControlMacro2** in place of Radio1QuickSwithRTTY in the left-hand digital modes panel, Logger32 triggers macro button #2 on the **R**adio **C**ontrol **P**anel. Macro #2 then sends your choice of CAT commands to radio 1, setting it up ready to work RTTY *e.g.* $**command ABC;DEF;GHI;JKL;MNO;**$ where **ABC** is typically replaced by the CAT command to set the radio to DATA or RTTY mode, **DEF** disables the speech processor, **GHI** disables the front panel microphone input and enables the rear panel mic/data input, and the remaining commands set up filters, RF amplifiers *etc*. just the way you like[121].

Similarly, **PSK-RadioControlMacro3** would trigger RCP macro #3 when you click an interesting PSK DX spot.

Q. Is there a quick way to *clear* the log entry pane?

A. No. Not one, not two, not three, but *four*. Choose any of these four:

1. Click the left side of the log entry pane, where the field labels are (Call, Sent, Rcvd *etc*.).
2. Press <**Ctrl**+**C**> for **C**lear.
3. Press <**Alt**+**W**> for **W**ipe.
4. Right-click the log entry pane and choose <**Clear entries**> ▶



5. [Free bonus method] Press function key <**F11**>.

[121] If your radio needs hexadecimal CAT commands, replace $**command** with $**hexcommand** followed by a space then one or more hex commands each terminated with a semicolon.

Copyright © 2026 Robert C. Furzer K4CY

Q. I'm sure I've worked K4CY before. If I work him again, can't Logger32 remind me of his name and other info?

A. Yes. Take a look at the QSO mask function that can carry-forward various fields from previously-logged QSOs.

Q. Why does Logger32 warn me when I tune my VFO?

A. Try reading the warning message for starters. You have probably tuned your radio beyond the edge of one of the amateur bands defined in your Bands & Modes table. You could add a catch-all "GEN" (**GEN**eral coverage) entry at the bottom of your Band Plan covering the frequency range .0000 to 100.000 (or higher). For example:

GEN,USB,.000000,100.000000,,LSB,,N,,,

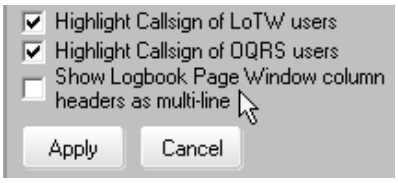
Alternatively, deselect <**Enable out of band messages**> on the <**View**> menu and Logger32 will let you tune around at will, unhindered by those annoying warnings. Whatever you do, though, do be careful not to transmit out-of-band and risk losing your license and/or liberty.

Q. Help! My latest QSO isn't shown at the bottom of the log, but I definitely logged it, and I can find it with a search. Where is it? Has it *gone*?

A. First of all, check that your logbook is sorted in date and time order. Click the date column header to turn it red.

If that doesn't solve your problem, it is likely that your current logbook layout and settings result in the bottom row being partially or completely hidden. Hopefully, one of these workarounds will fix it:

- Adjust the vertical size of the logbook by dragging the top or bottom edge of its window up or down just a little.
- Tick or un-tick **View ⇨ Lock child windows**.
- Tick or un-tick the multi-line header option at the bottom of the forbidding gray grid layout form ▶ or on the right side of the less-forbidding single logbook column selector form.



- Click toolbar icon #16 or **Tools ⇨ Snap windows to size**.
- Adjust the logbook row height a little under **View ⇨ Grid appearance ⇨ Row height**.

Failing all that, *try* persuading Bob K4CY to redesign the logbook display as a conventional Windows pane with conventional Windows functions such as up/down sorting on any column, a scroll slider that works reliably, arrow-key and page up/down key and mouse wheel scrolling *etc*. Good luck with your quest.

Q. Why can't I select, say, USER_3 for one of my log entry pane fields?

A. ADIF fields that are already in use in any of the log entry panes are 'taken' and so cannot be selected for additional data entry fields. If you really want to use them, you will have to de-select them from the current fields first.

Copyright © 2026 Robert C. Furzer K4CY

Q. How can I prevent Logger32 from assigning a given DXCC to a callsign?

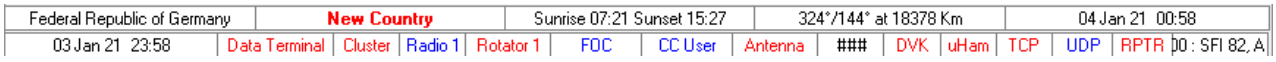
A. Update the DXCC field in the logbook. Since this is a key field, you can't simply overtype the entity name in case you mis-spell it, inadvertently creating a new [false] DXCC entity. Instead, right-click the QSO, click <**Edit country info**>, then select the correct entity from the list.

Q. I need to be on 40m right now for a sked. How can I QSY, *quickly*?

A. In the Call field of the log entry pane, type your intended frequency in the same format as the frequency shown at the top of the pane, then <**Tab**> or <**Enter**> to send your CAT-connected radio straight there. This is handy if someone you are working asks you to try another band or shift to avoid QRM, or if a rare DX station you've been calling in vain on 20m quietly announces "QSY 10110": type 10110 into the Call field and <**Tab**> to QSY instantly.

Q. If I add user fields to my log entry pane, they cover up the country, distance, bearing info *etc.* How can I still have this information *and* use the user fields?

A. If it's not already shown, turn on the DX info bar near the bottom of Logger32's main window using **View ⇨ Show DX information bar**. It displays all this sexy info *e.g.* ▼



Q. I *think* I made an FT8 QSO with P5DX, but due to QRM and QSB, I never received his 73 message. What now?

A. Log the questionable QSO as an informational message. Temporarily remove the equals sign, then sign and upload it to LoTW, and replace the equals sign. Check on the LoTW website from time to time to see if it is confirmed, or simply run the LoTW download and synch process as normal, routinely checking the *BAD.ADI* file for any 'QSO not found' errors. If you find the QSO in question listed, it has been matched and confirmed so go ahead and remove the equals sign for good. Congrats!

Use the equals sign so as not to count the dubious QSO in your statistics until/unless it is actually confirmed, whereupon you can remove the equals sign. Meanwhile, if any P5 station is spotted, you will hopefully be alerted in time to beat the rush and grab an 'insurance QSO'.

Q. Is there a possibility of using a wildcard character in the callsign search, wherever in the log window, with the right click in the QSO data entry window?

A. Try configuring the Previous QSOs pane to show QSOs with stations' home calls – see section 9.1 for instructions. That will find and list all your logged QSOs with the station currently in the log entry pane, wherever he was, *provided* you logged him under his home callsign with location variants such as P5/ZL2iFB. Logger32 needs the clue about who you mean. It is useful, though, to find all your QSOs with serial expeditioners such as G3TXF and SP5APW, in order to QSL the most interesting ones.

Copyright © 2026 Robert C. Furzer K4CY

Q. Ooops. Somehow I've lost the *main Logger32 window* … so I can't open its <View> menu to get it back. What now, Brains?

A. Occasionally due to obscure bugs hiding deep in the software, Windows programs lose their user interfaces (their windows!), leaving us no way to see and interact with them. According to the Windows task bar and task manager, the programs are still running uselessly and invisibly in the background. It's as if they have dissolved *through* the monitor and gone out the back, like Alice through the looking glass. Or something.



Work systematically through these possible solutions:

1. Close and re-start Logger32: you may be able to close it from the preview display that appears when you mouseover the Logger32 icon on the Windows task bar along the bottom edge of the screen, by clicking the tiny corner X on the little screen preview ▶

   Otherwise, use <**Win**+**X**> **T** to open the Windows **T**ask Manager, then right-click and <**End task**> to terminate the *Logger32.exe* process (with extreme prejudice!).

   Now re-start Logger32, with your fingers crossed that the main Logger32 screen appears.
2. If it is still hidden, minimize or close all the other open windows, one by one, hoping that Logger32 will be revealed at some point. Maybe it was hidden beneath them. It is quite shy.
3. Click ▼ the Logger32 task bar icon at the bottom of the screen, hoping that the focus is on the [invisible] log entry pane's Call field. Now type **FIND** <**Enter**> as if you were entering that callsign. Logger32 moves the main window to the front in the top left corner of the display.



4. Click harder, muttering expletives.
5. Close your eyes and count to ten.
6. Try a reboot ▶
7. Replace *C:/Logger32/Logger32.ini* with a recent copy of the file from the backups.
8. Close Logger32, then delete *C:/Logger32/Logger32.ini* then reconfigure Logger32 from scratch. Sorry. We tried.



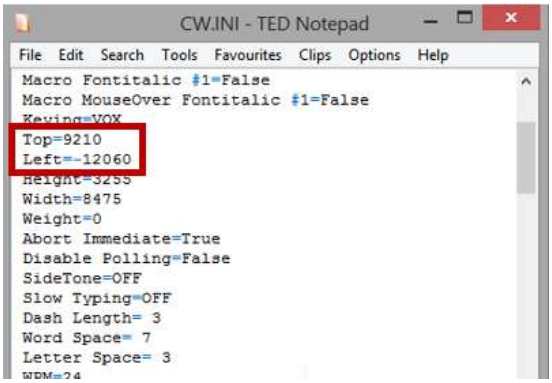
Copyright © 2026 Robert C. Furzer K4CY

Q. I've lost the CW Machine. Where should I look?

A. The CW Machine is an example of a non-child (grownup?) window that is not listed under **View ⇨ Find lost windows**. You'll have to reset its screen location manually.

Close Logger32, then edit *C:\Logger32\CW.INI* in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad.

Change both the "Top" and "Left" parameters to zeroes ▶



Save the edited *CW.INI* file then re-open Logger32. Now, when you open the CW Machine, it will start top left of the screen: move it wherever you want it, and it *should* behave itself in future. Give it a good talking-to if that makes you feel better.

Q. Why isn't the particular window *I* want to see listed under the <View> menu?

A. The <**View**> menu only lists Logger32's child windows *i.e.* subsidiary windows that are constrained within the area of the main Logger32 window. Non-child windows could be hiding anywhere on any screen and can be trickier to find.

The simplest and quickest thing to try is to close and re-start Logger32.

If the missing window still doesn't appear, maybe it isn't open *e.g.* the CW Machine may have been closed, so try clicking the toolbar blue Morse key icon #13 to open it.

If it *still* stubbornly refuses to show itself, you may need to zero the Top and Left parameters in the *.INI* file as noted above.

Q. The floating callsign window must have sunk. Where has it gone?

A. Find it using **View ⇨ Find lost windows**. When the 'currently open' form is displayed, tick <**Floating**> then <**Find**> to disable *glass* mode and move the floater to the top left corner of the screen.

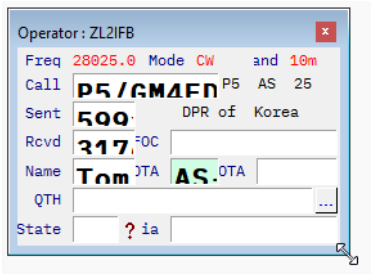
Alternatively, press <**Ctrl**+**F**> twice in the log entry pane, CW Machine, or Sound card data window to disable the *glass* option, making the **f**loater less transparent, opaquer.

Copyright © 2026 Robert C. Furzer K4CY

Q. Getting on a bit now and eyesight is getting a bit dim. How can I enlarge the text in the log entry pane?

A. There's lots to adjust here, including the type of font and font size.

If you choose a much larger font, you may want to make the pane bigger to avoid truncating the fields. Tick allow automatic window height adjustment on the log entry pane right-click <**Setup**> menu to scale the data entry fields in proportion to the pane. Make the log entry pane taller and all the fields become taller. Make the log entry pane wider, and all the entry fields become wider. Put fake text into the pane to try it out ▼▶





Q. There's something wrong with *my* system, but nobody else seems to suffer the same problem. Am I "special"?

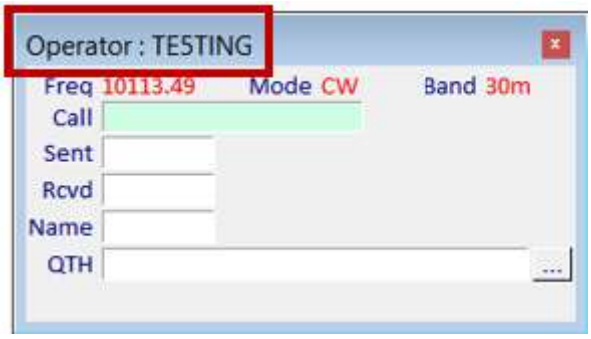
A. Yes, you are. Everyone who uses Logger32 is special (no quotes). The fact is that we each have unique ways of configuring and using the software, exploiting the myriad configuration options available to us. The beta test team is literally unable to test every possible permutation of configuration options but, along with other Logger32 users, we exercise all the common configurations and functions frequently, and the more obscure ones occasionally. *Maybe* you have discovered a very obscure design flaw or bug that nobody else has experienced, perhaps something caused by your specific configuration and/or the way you use the system.

You can explore this further for yourself by temporarily reverting to the default Logger32 configuration and finding out whether your problem still occurs:

1. Before you start faultfinding, take a mental note of the problem you are experiencing. What does it affect? When does it occur? What are the symptoms? Ideally, figure out the precise circumstances, sequence of activities or data that cause the problem to occur. This will help immensely at step 6.
2. Export your whole log as an ADIF file, as a just-in-case log backup. Keep that safe.
3. Close Logger32.
4. In your Logger32 folder (usually *C:\Logger32*), rename *Logger32.INI* to something like *Logger32suspect.INI*
5. Run Logger32. It will generate a new *Logger32.INI* with the default settings. Your screen layout will look awful but don't worry: this is just for testing purposes.

Copyright © 2026 Robert C. Furzer K4CY

6. When prompted for your callsign, enter a fake operator callsign that is 'clearly' just for testing *e.g.* TE5TING. That way, provided you are observant, you are less likely to mistake the testing setup for your normal setup with the fake callsign emblazoned across the log entry pane's caption ▶



7. Optionally, select *nasty* highlighting colors to make it even more obvious that you are using the testing configuration.
8. Do whatever you need to do to trigger the problem. Depending on what it was, you may need to do some limited configuration first *e.g.*:
   - If the problem was something to do with searching your log, you could log a few fake QSOs in your TE5TING log in order to be able to search. You can also import the ADIF log backup to recreate your normal log … but don't forget you are just testing at this stage.
   - If the problem involved Logger32's communications with your radio, configure your radio and open its CAT connection in order to check it out.
   - If you were having trouble with DX cluster spots, you'll need to get a DX cluster connection working so that you can play with it.
9. If you simply cannot trigger the problem in the test setup, despite your very best efforts to reproduce it, there was most likely something wrong with your original configuration. You now have choices. You can:
   - Manually reconfigure Logger32 to your liking, using the test setup. Don't forget the current operator is the fake callsign you entered earlier. Use **File ⇨ Change operator** to revert to your real callsign, and **File ⇨ Change Logbook** to open your real log.
   - Using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad, hunt through *Logger32suspect.INI* for any gross issues such as a truncated file or gibberish content.
   - Take a closer look at the *.INI* file section relating to the problem (*e.g.* the DX cluster section): is there anything obviously amiss? If not, try deleting the whole section, then save the file as *C:\Logger32\Logger32.INI* and re-start Logger32. You will have to reconfigure that function but the remainder of your configuration should be OK.
10. If the problem persists in the testing setup, having eliminated your original configuration as the cause, something else is evidently playing up – probably Logger32, possibly Windows or some other software running on your system, your radio, a device driver, your Internet connection, your PC or … whatever. Good luck finding and solving the problem. We'll do our best to help you via the Logger32 reflector, especially if that's the root cause. In this case:
    - Close Logger32.
    - In *C:\Logger32* folder, rename *Logger32.INI* to *TESTING.INI* so that, later on, you can resume testing by opening Logger32 with this configuration.
    - In *C:\Logger32* folder, rename *Logger32suspect.INI* to *Logger32.INI*
    - Run Logger32 as usual. You are now back where you started, a little older and wiser.

Copyright © 2026 Robert C. Furzer K4CY

Q. If I type a callsign into the log entry pane and either click/tab into a report field or simply hit enter to log the QSO, the report is pre-filled with 579 (on CW), not 599 as I expected. Why is that?

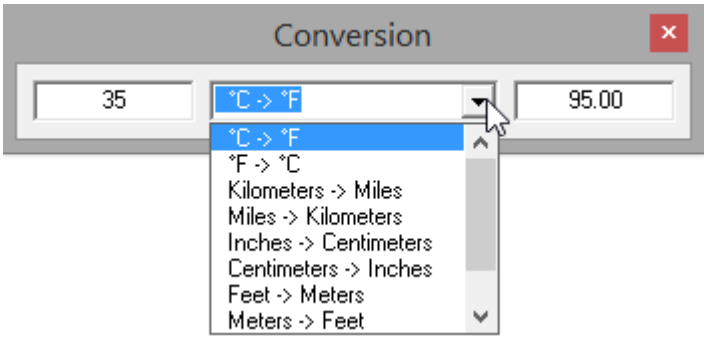
A. You have presumably changed the default report. Check the <**Report**> column of your Bands & Modes table. Check/change the default reports for each band+mode segment in the table, or leave them empty if you would rather not log standard reports automatically, preferring only to log reports that you have sent/received on-air *and* entered by hand.

Q. Just how hot *is* 35°C?

A. A handy little utility can convert values between common units of measure (*e.g.* kilometers and miles, degrees Celsius and degrees Fahrenheit). If someone tells you their beam is 167 feet above the ground, it's easy to work out just how many metres that is. Yes, it's a lot.

Click the curious 'dead cat peering out of a coal hole' #20 Conversion icon on the toolbar ▶





◀ Enter the value to be converted into the left-hand field (just the number, no units) then select the particular conversion required from the drop-down list in the middle, and the answer appears on the right.

Hinson tip: the conversion utility can't handle currency conversions (try the FT instead), nor is it any use for the curious non-SI units of measure much favored by journalists (London buses, Olympic swimming pools, Empire State Buildings, elephants, human hairs, distance to the moon …) or the signal reports often heard on-air ("You are five and nine but I need your callsign and name again") so until Bob codes the associated functions, you might just have to refer the person you are working to the Plain English Campaign.

Copyright © 2026 Robert C. Furzer K4CY

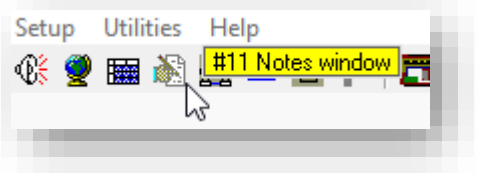
7 Scratchpad and notes

> "Writing is thinking on paper"
>
> *William Zinsser*

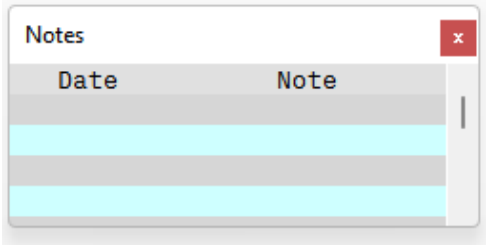
Use the notes or scratchpad functions in Logger32 to make little electronic notes about QSOs in progress or spotted DX, just as if you were scribbling notes on a paper pad next to the radio.

Clicking icon #11 on the toolbar ▶
or <**Ctrl**+**N**> opens the **N**otes window.

<**Ctrl**+**Z**> opens the scratchpad.



7.1 Notes



Clicking icon #11 or pressing <**Ctrl**+**N**> opens a little pane ▶ in which you can type **N**otes as if you were jotting things down against QSOs in your physical logbook.

Notes are saved in your logbook with logged QSOs[122]. The Notes pane simply displays the notes recorded against previous QSOs with a given station using the callsign currently in the log entry pane for a QSO in progress, one you are about to log. This is a basic text jotter function, whereas the scratchpad is far sexier …

7.2 Scratchpad

The scratchpad lets you temporarily store and edit QSO information without actually logging it, not yet anyway.



Open the scratchpad by pressing <**Ctrl**+**Z**> while the focus is on the log entry pane. That copies the callsign and other QSO info from the log entry pane (if any) to the Scratchpad ▸

The scratchpad can hold multiple QSO lines. It is as if you

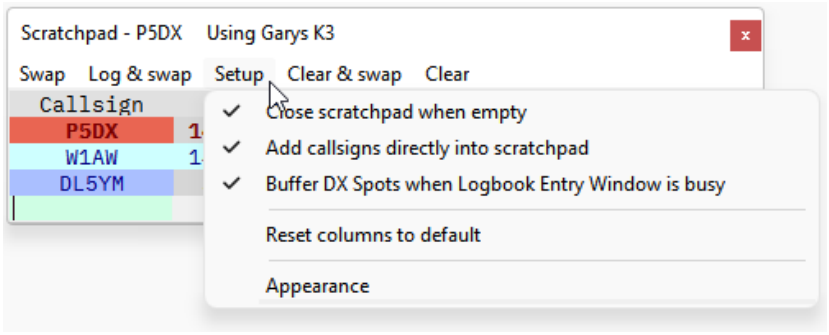
[122] If you have the Notes field displayed in your logbook or previous QSOs panes, you can type or edit notes directly in those fields without using the Notes function at all.

Copyright © 2026 Robert C. Furzer K4CY

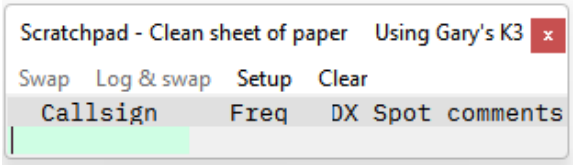
are building a personal, editable list of interesting DX spots with the usual visual highlighting of any new ones. Clicking a QSO line in the scratchpad (or clicking <**Swap**> or pressing <**Ctrl**+**X**>) is much the same as clicking a D**X** spot on the DX Spots pane or BandMaps, sending the QSO information from the scratchpad to the log entry pane and QSYing your radio, ready for you to complete and log the QSO.

7.2.1 Scratchpad setup

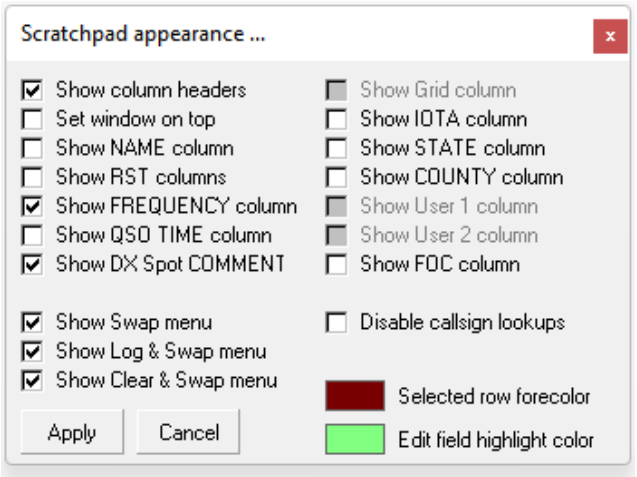
To use the scratchpad, press <**Ctrl**+**Z**> to open it, then configure it using <**Setup**> ▼



- **Close scratchpad when empty**: the scratchpad can automatically close and disappear after the last entry is either deleted or transferred to the log entry pane[123]. If not ticked, clicking the X corner will close and disappear it.
- **Add callsigns directly into scratchpad:** tick this to be able to type a callsign and other data directly into a new blank row in the scratchpad – for example, jotting down notes about a weak DX station you've heard or someone new joining a net. If un-ticked, new scratchpad rows can only be added by transferring entries from the log entry pane using <**Ctrl**+**Z**> or <**Ctrl**+**X**>.
- **Buffer DX Spots when Logbook Entry Window is busy**: rather than immediately QSYing the radio as normally happens when you click a DX spot, if there is information in the log entry pane at the time (*e.g.* while you are in QSO with someone else), the spotted information is saved to the scratchpad so you can check it out later, at your leisure. This also opens and displays the scratchpad if it was not already shown.
- **Reset columns to default**: resets the scratchpad layout if 'someone' has somehow made a complete mess of it and you want to start over. *Note: this also clears the scratchpad.*
- **Appearance:** opens a scratchpad display customization submenu ►



  - ▲ <**Show column headers**> labels each column in the scratchpad. Handy if you forget what they are.



[123] With one exception: even if the scratchpad is empty, <**Ctrl+Z**> from the log entry pane opens the empty scratchpad … and it stays open so we can configure it.

Copyright © 2026 Robert C. Furzer K4CY

- <**Set window on top**> prevents the scratchpad from being covered up by other windows *e.g.* if you are reading your email or browser while patiently waiting your chance to call someone in a DX net. When you want to use it, click the scratchpad to bring Logger32 fully to the front.
- Tick to show whichever columns and menu options you want on the scratchpad[124].

> Hinson **tip**: it is safer *not* to show **Clear & swap** if there is a chance you might click it by mistake, accidentally wiping whatever happened to be in the log entry pane at the time, perhaps losing details of a QSO in progress. The <**Ctrl**+**C**> hotkey still works though.

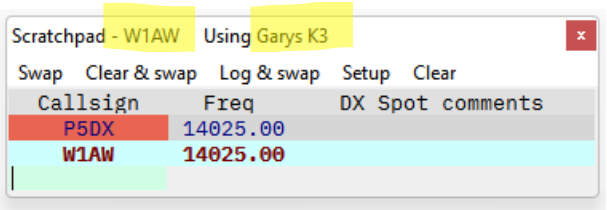
- <**Disable callsign lookups**> avoids the processing delay while looking up a station's DXCC info, distance *etc*. when saved to the scratchpad, if you don't need that additional information. This setting only affects lookups for lines on the scratchpad: the log entry pane lookups work as normal.
- Click the colored rectangles to choose the forecolor (text/font color) for the selected row, and the highlight (background shading) color for the row being edited. The callsign fields use the same highlighting colors as your DX spots and band maps for new ones (new all time, new this year, new this band or mode *etc*.).
- Click <**Apply**> to finish and save your custom configuration, or <**Cancel**> or the corner X if you chicken out.

7.3 Using the scratchpad

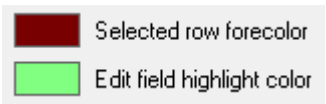
Having opened the scratchpad using <**Ctrl**+**Z**> with the focus on the log entry pane, here's what you can do with it …

7.3.1 Caption

The top line of the window shows the callsign from the selected scratchpad QSO line, and the active radio[125] ▶



7.3.2 Navigation and hotkeys



Select any scratchpad row directly by clicking it with the mouse. The selected row's text goes bold in the color chosen under the <**Setup**> ◀ menu.

The following navigation and hotkeys are then active:

- **Up/down arrow** selects the scratchpad row above or below the current one (if any).
- <**Ctrl**+**up/down arrow**> moves the current row up or down in the list.

---

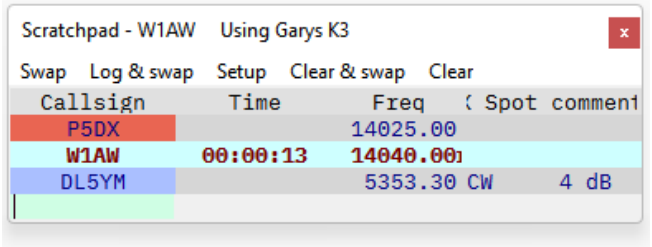
[124] Newly-selected columns may not appear instantly if the scratchpad is too narrow: simply draft the right border to the right to see them, then reposition/resize the columns and resize the scratchpad.

[125] The radio is named under **Setup ⇨ Radio ⇨ Radio 1|2 setup**. "Garys K3" is mine. If we change the name, the new name is picked up when we next open the scratchpad.

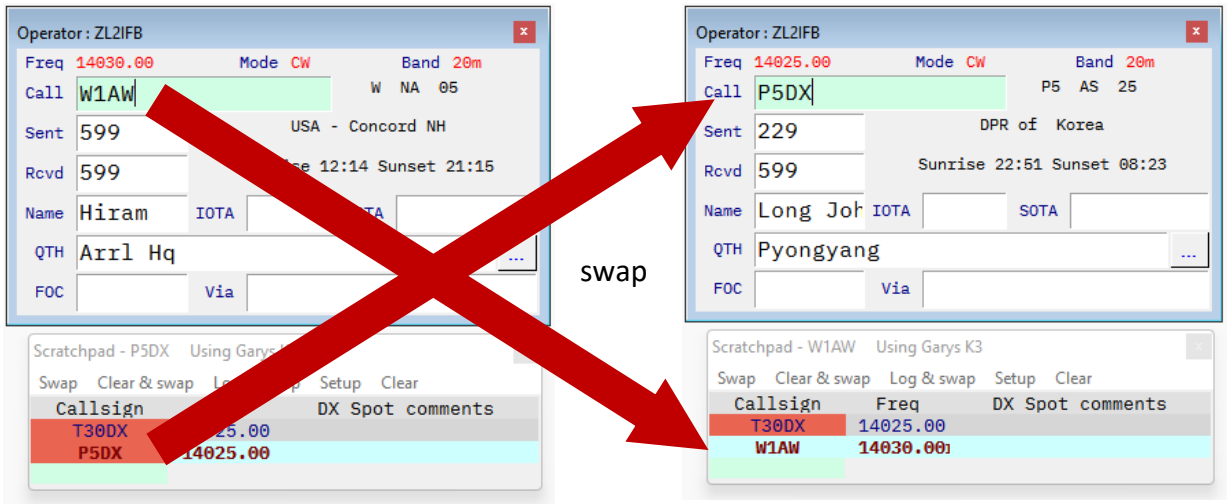
Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: net controllers and list operators can use this facility to re-sequence the callsigns of checked-in stations in the scratchpad, for example cycling systematically through the list of callsigns. Change the order to give priority to DX, mobile or portable stations over fixed/home stations, or give someone who needs to leave the net soon their chance to sign out gracefully.

- **<Ctrl+C>** moves the current row from the scratchpad to the log entry pane, **C**learing anything already there as if you had clicked the Clear & swap menu item. Take care with this!
- <**Ctrl+D**> **D**eletes the current row from the scratchpad.
- <**Ctrl+E**> lets you **E**dit any editable field in the current row of the scratchpad.
- <**Ctrl+K**> **K**opies (copies!) the selected row from the scratchpad to the log entry pane, as if you had clicked the <Swap> menu item. The focus moves to the log entry pane, ready for you to update the information and log the QSO when you are ready.
- <**Ctrl+L**> **L**ogs the QSO currently in the log entry pane and moves the selected row from the scratchpad to the log entry pane as if you had clicked the <Log & swap> menu item.
- <**Ctrl+R**> **R**esets the scratchpad columns to their default sequence and widths. Use this if you have messed up the scratchpad layout and decide to have another go at the configuration.
- <**Ctrl+T**> sets the QSO start **T**ime (TIME_ON) for the selected row, as if you had used the QSO start time menu item[126]. If the QSO Time column is shown on the scratchpad, it then displays a timer ▸ counting up the hours, minutes and seconds from that point.



- **<Ctrl+X>** swaps the current row from the scratchpad with whatever is in the log entry pane[127] as if you had clicked <**Swap**> on the menu ▼



- **<Esc>** lets you **Esc**ape from the scratchpad, returning focus to the log entry pane.

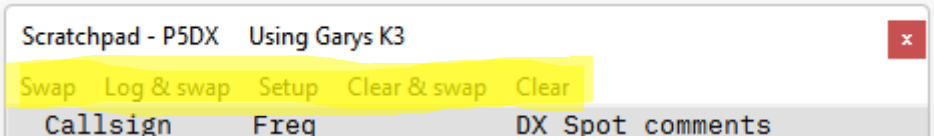
[126] If you have configured <**Ctrl+T**> globally to toggle radios in an SO2R setup, that takes precedence *unless* the scratchpad has focus.

[127] Think of control+X as "controlled crossing", if that means anything to you. It does to me. Sort of.

Copyright © 2026 Robert C. Furzer K4CY

7.3.3 Scratchpad menu

The scratchpad's menu ▼ can show *up to* 5 functions[128]

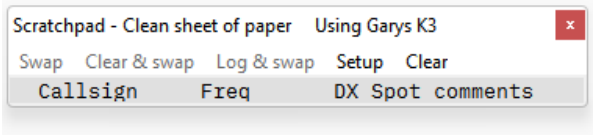


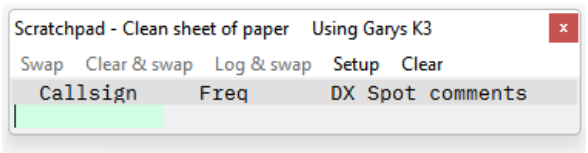
1. **Swap**: if there is more than one row populated in the scratchpad, click to select one (turning the text bold, and putting the callsign into the scratchpad caption), then click <**Swap**> or use <**Ctrl**+**X**> to send that row to the log entry pane, QSY the radio *etc*. almost as if you had just clicked a DX spot, *and* save the log entry pane information (if any) as a new row in the scratchpad (which does *not* happen when you click a DX spot). Click <**Swap**> again to swap them back. Knock yourself out: <**Swap**><**Swap**><**Swap**><**Swap**> … as often as you like, flicking back and forth between two pileups maybe, or to keep checking on a pileup while you are in QSO with a local, waiting for your chance to make your excuses and dash away to bag the DX.
2. **Log & swap**: logs the QSO *currently* in the log entry pane and clears it, then moves the selected scratchpad row into the log entry pane ready for your *next* QSO. As usual, depending on your configuration, the radio QSYs, the DXCC entity (plus the short path distance, heading and the station's local time) are all looked up, the worked/confirmed table is populated, the wanted status is checked, and an online callsign lookup is performed.

> **Hinson tip**: despite my dislike of list operations, <**Log & swap**> could be handy to work my way through a little list of callers whose callsigns I have entered onto the scratchpad when multiple people responded to my CQ, or when someone rudely called me *during* an ongoing QSO. Simply complete the current QSO in the log entry pane then click <**Log & swap**> or <**Ctrl**+**L**> to log it and line up for the next.

3. **Setup**: see the previous section.
4. **Clear & swap**: *wipes* anything currently in the log entry pane, replacing it with the currently selected scratchpad row. Depending on your configuration, the radio QSYs, the DXCC entity plus the short path distance, heading and the station's local time are determined, the worked/confirmed table is populated, the wanted status is checked, and an online callsign lookup is performed … just as if you have clicked a DX spot. Take care!
5. **Clear**: this simply wipes the entire scratchpad contents, like ripping off a used page from your paper note pad … only without the ripping noise and paper cuts. Take care! The erased information disappears in a puff of logic and cannot be retrieved unless you can remember it.

Note: if <**Add callsigns directly into scratchpad**> is *not* ticked in <**Setup**>, the cleared scratchpad looks odd with no data rows, only the caption and menu, plus the column headers *if* <**Show column headers**> is ticked in **Setup** ⇨ **Appearance** ▶





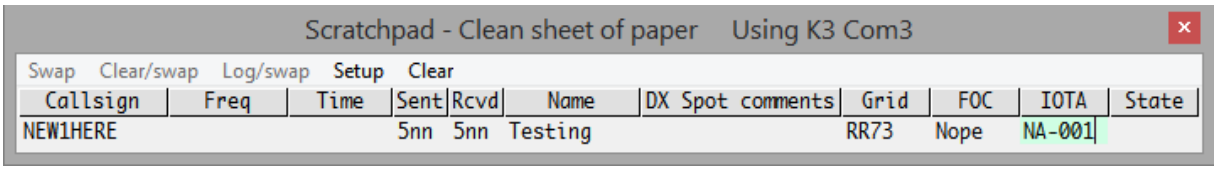
◀ However, if <**Add callsigns directly into scratchpad**> *is* enabled in <**Setup**>, clearing the

[128] You can hide <**Swap**>, <**Log & swap**> and/or <**Clear & swap**> from the menu by unticking them from **Setup** ⇨ **Appearance**, using their hotkeys or the right-click menu instead if you want to invoke them.

Copyright © 2026 Robert C. Furzer K4CY

scratchpad leaves one vacant data row ready for you to enter information.

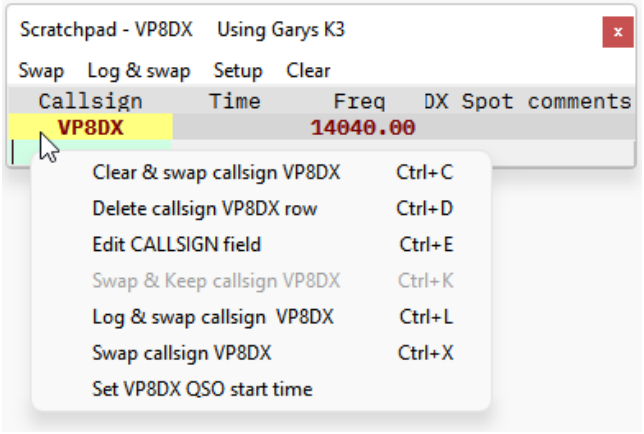
Enter information into the vacant row similarly to how you enter it in the log entry pane: type data into a field in the correct format, depending on the field[129] (*e.g.* grid squares and IOTA references) and <**Tab**> or click to the next editable field ▼



Then, when you are happy with the line, press <**Enter**> to stop editing and save the line in the scratchpad. When you hit <**Enter**>, the DXCC and other info is looked up (unless disabled in **Setup** ⇨ **Appearance**), the text on that line goes bold and the line can be used *e.g.* swapped into the log entry pane. However, if you forget to press <**Enter**>, the line cannot be used. It's not broken, just in limbo, still being edited.

7.3.4 Editing the scratchpad content

With the scratchpad open, right-click any row to open a menu of editing functions for that row ▶



Hinson tip: notice the Ctrl+letters. Those remind you of the hotkeys to trigger the functions directly, without having to pop open the right-click menu first.

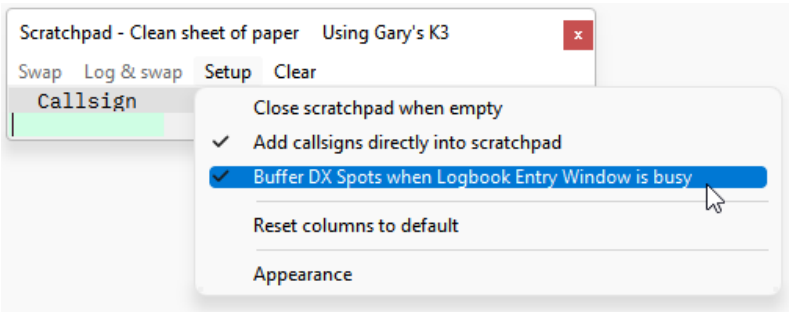
1. **Clear & swap callsign [**callsign**]**: works the same as clicking <**Clear & swap**> on the menu.
2. **Delete callsign [**callsign**] row**: removes the row from the scratchpad. If it is the only row, removing it empties the scratchpad … which then automatically closes itself *if* you have ticked **Setup** ⇨ **Close scratchpad when empty**.
3. **Edit [**NAMED**] field**[130]: this *may* let you edit whichever field you right-clicked … but the Freq, Time and DX Spot comment fields are read-only fields, not user-editable, so the edit field functions are grayed-out if you right-clicked them. Also, you cannot edit a county *unless* a state is specified, in which case Logger32 only allows you to select a county within the specified state (these are linked database fields).
4. **Swap & Keep callsign [**callsign**]**:
5. **Log & swap callsign [**callsign**]**: does the same as <**Log & swap**> on the menu.
6. **Swap callsign [**callsign**]**: does the same as <**Swap**> on the menu.
7. **Set [callsign] QSO start time**: puts the current UTC time into the QSO Time field. You will notice the seconds counting up. When you swap or move the entry to the log entry pane to make the QSO, the TIME_ON value is set at that point.

[129] The <**Frequency**> field automatically shows the current VFO frequency from your CAT-connected radio when you manually enter a new line in the scratchpad, but you can always edit it in the same format.

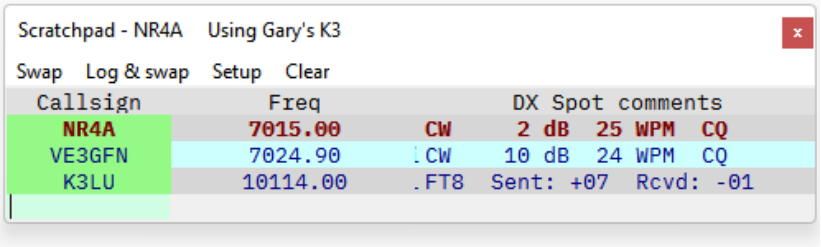
[130] I presume it SHOUTS the name of the field at us because ADIF field names are DEFINED IN CAPITALS.

Copyright © 2026 Robert C. Furzer K4CY

7.3.5 Scribble interesting DX spots to the scratchpad

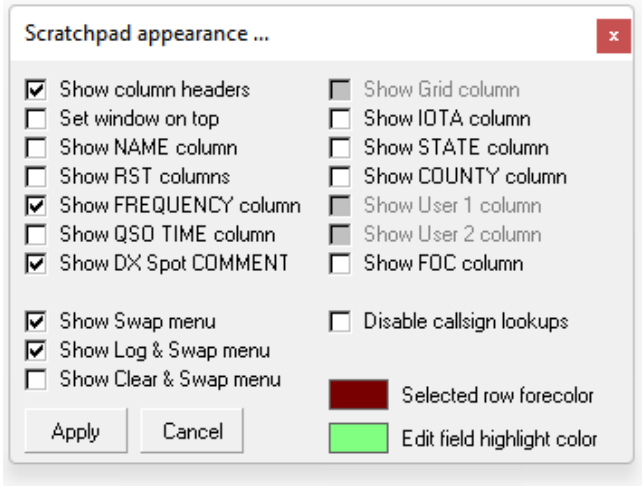


With **Setup ⇨ Buffer DX Spots when Logbook Entry Window is busy** ticked ▲ you can save interesting DX spots to the scratchpad *while* you are busy making and logging QSOs *i.e*. with QSO info in the log entry pane. Simply click a spot in the DX Spots pane or BandMaps to save the DX callsign, spotted frequency and spot comment (which may specify the mode and split) to your scratchpad, rather than to the log entry pane as usual. No frantic scribbling required!

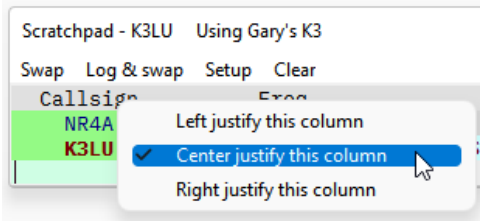


◀ Here I have clicked to save 3 interesting spots that appeared while I was logging a QSO. Once the log entry pane is empty (*e.g.* having logged the QSO in progress), I can click any row from the scratchpad to send it to the log entry pane and QSY the radio to find out if the spotted DX station is still there, removing it from the scratchpad.

Using the scratchpad in this way as a kind of DX spot memo, I don't need to show all the scratchpad columns available under **Setup ⇨ Appearance** ▶



To resemble the DX Spots pane, I have aligned the columns by right-clicking their headers and selecting the justification ▼

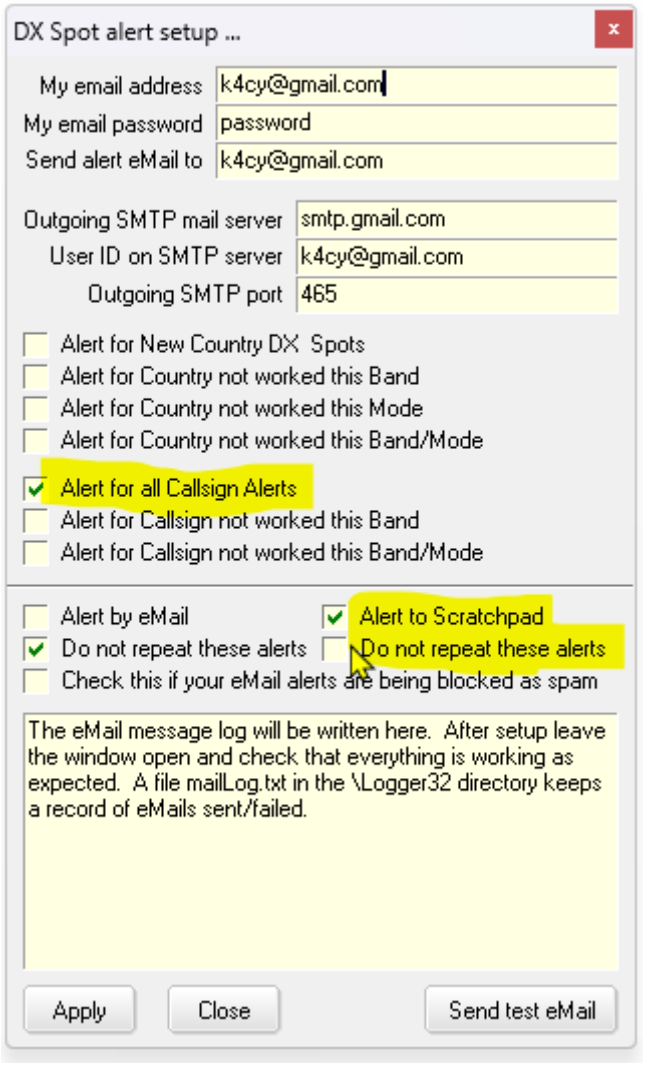


Copyright © 2026 Robert C. Furzer K4CY

7.3.6 Automatically send interesting DX spots to the scratchpad

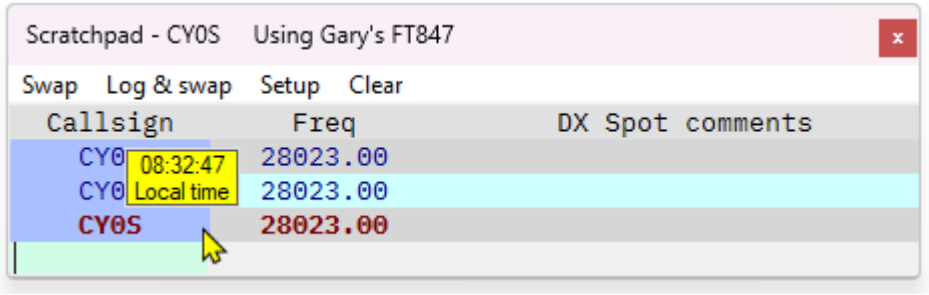
Bryan N3ST has found a creative way to compile a list of interesting DX spots for stations he 'needs', even while he is away from the shack. He uses the DX spot alerting function.

To set this up, he right-clicked the DX Spots pane, then clicked **Setup ⇨ Setup DX spot alerts (eMail/SMS/Scratchpad)** ▶



Having previously configured the alerts, ticking both <**Alert for all Callsign Alerts**> *and* <**Alert to Scratchpad**> tells Logger32 to send alerted DX spots to Bryan's scratchpad ▶

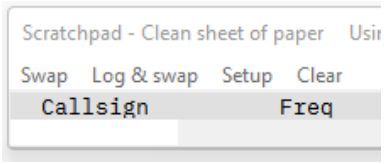
Not ticking the <**Do not repeat these alerts**> on the right means repeated spots for the same interesting DX station are all sent to the scratchpad, building a chronological record of the spotted station's activity over a period. When he returns to the shack, Bryan can simply point the mouse at the spotted callsigns in the scratchpad to discover when they arrived (in his local time) ▼



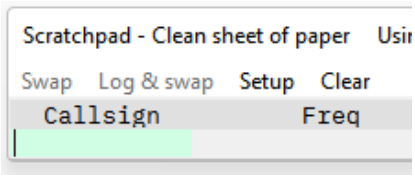
The time of day when an interesting DX station was spotted gives Bryan a big hint as to whether they might still be active, and (if he's lucky) perhaps a clue to their future schedule and favorite frequencies if they are habitual DXers.

Copyright © 2026 Robert C. Furzer K4CY

7.3.7 Field background colors in scratchpad



The data entry field's background color is the same as the log entry pane, in my case ◀ white for *Data field background* and green for *Edit field highlight color* ▶



7.4 Log entry pane ⇄ scratchpad

There are several ways to copy or move QSO information from the scratchpad into the log entry pane, to move QSO information from the log entry pane into the scratchpad, or do both at once *i.e*. swap over (e**X**change) the QSO information between the two:

1. **Move QSO info *from* the log entry pane *to* the scratchpad**: move the QSO currently in the log entry pane to the scratchpad using <**Ctrl**+**Z**>[131], for instance if a friend calls in for a quick report but you plan to resume your QSO with the station you were originally logging afterwards, so move the original QSO temporarily to the scratchpad while you work and log your pal.
2. **Move an entry *from* the scratchpad *to* the empty log entry pane**: on the scratchpad, simply click the row you want to move. This is the default action. However, if the log entry pane is *not* empty, the click merely selects the scratchpad row but doesn't move it[132].
3. **Move an entry *from* the scratchpad *to* the log entry pane, erasing anything already there**: if you don't want to log or keep the information currently in the log entry pane, but merely replace it with the currently-selected row in the scratchpad, click <**Log & clear**> if shown on the scratchpad menu, or right-click the relevant row and click <**Clear & swap callsign [**callsign**]**>, or press <**Ctrl**+**C**> to move the currently-selected row from the scratchpad to the log entry pane, having **C**leared (wiped) it.
4. ***Exchange* the QSO info between log entry pane and scratchpad:** with the focus on the log entry pane, press <**Ctrl**+**X**> to e**X**change them. Alternatively, click <**Swap**> on the scratchpad menu.
5. **Log the QSO in the log entry pane *and* Get the next one from the scratchpad:** with the focus on the log entry pane, press <**Ctrl**+**G**>. Alternatively, click <**Log & swap**> on the scratchpad menu (if shown), or right-click 'the next one' on the scratchpad and select <**Log & swap callsign [**callsign**]**> or press <**Ctrl**+**L**>. This function may be useful if you are running a small pileup with a few people patiently waiting to work you, or if you find a few interesting stations that you want to call: simply save their details in the scratchpad and **G**et them when you are ready.

[131] The shape of the letter Z is a visual reminder of this action: if the top bar of the Z is the QSO currently in the log entry pane, clicking <**Ctrl+Z**> notionally moves it down to the bottom bar *i.e*. the scratchpad.

[132] If the default action was <**Swap**> rather than <**Move**>, it would work whether the log entry pane was filled or empty. If filled, the QSO information would then be available in the scratchpad, not lost. I have requested a program change … and I'm very patient!

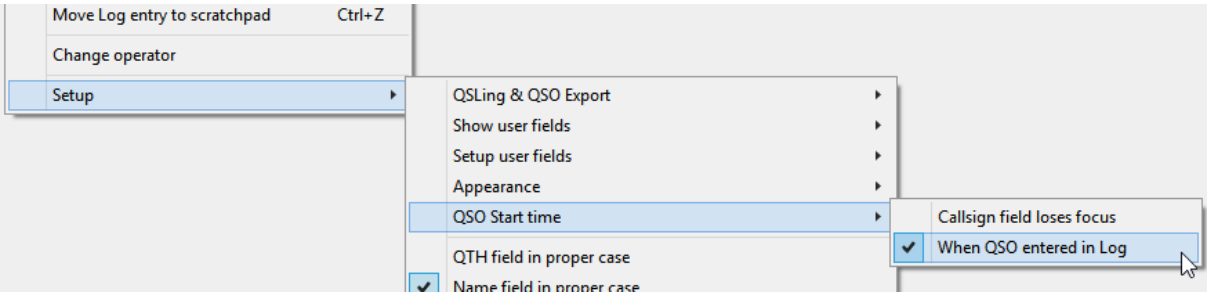
Copyright © 2026 Robert C. Furzer K4CY

7.5 Scratchpad and notes FAQs

Q. Imagine I had been calling P5DX for *hours* before finally making it. At what time should I log the QSO?

A. Determining when QSOs took place is trickier than you might think. There are two time points (QSO start and end times), both imprecisely defined. Naturally, Logger32 gives you options.

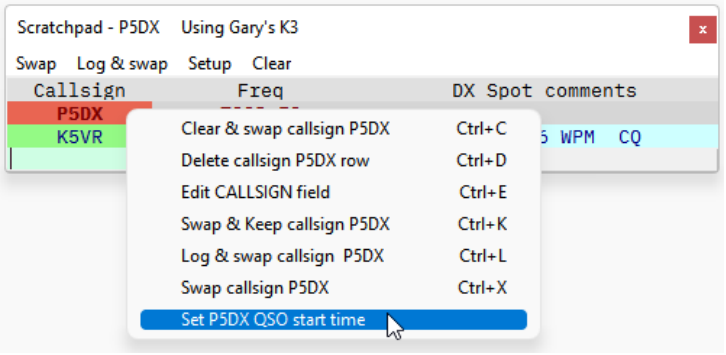
Right-click the log entry pane, then **Setup ⇨ QSO Start time ▼**



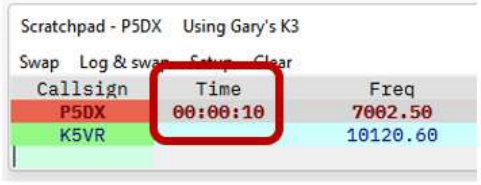
We typically *start* logging a DX QSO some time before QSO is established (if ever!) *e.g.* at the point we originally clicked the DX spot or came across the DX station and typed his callsign into the log entry pane. It is not unusual for determined DXers with modest stations to be calling a truly rare DX station literally for *hours* in a massive pileup … so that initial time point does not reliably indicate when the QSO occurred. Normally, however, we don't hit <**Enter**> to log the QSO until/unless we complete the QSO, and that generally happens during or shortly after the QSO is made, hence the timepoint <**When QSO entered in log**> is a better indication of when the QSO took place: logging a QSO sets *both* the TIME_ON *and* TIME_OFF values for the QSO to that same point.

The other QSO start time option, <**Callsign field loses focus**>, assumes that you don't move to or edit any other fields in the log entry pane *except* while you are in QSO with the station. If you are passively monitoring the DX with his callsign in the log entry pane waiting for your opportunity to call and make QSO, that's fine … but if you <**Tab**> to record, say, his name, IOTA reference or QSL details that he has given to someone else, that action sets the timepoint.

With <**Callsign field loses focus**> selected, however, you have another option to set the QSO start time in the scratchpad. Right-click any row in the scratchpad then click <**Set [**callsign**] QSO start time**> ▶



The elapsed time since the start appears in the scratchpad QSO time column, if shown, counting up the hours, minute and seconds ▼



Then, if you move that row to the log entry pane and log it, the QSO start time is logged as TIME_ON and TIME_OFF is recorded when you log the QSO.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why can't I close Logger32 with the scratchpad still open?

A. To avoid inadvertently losing the information you have stashed temporarily on the scratchpad when you close Logger32, Logger32 politely warns you to empty the scratchpad first ▼



Hit <**Cancel request**> to continue using Logger32 (*e.g.* to log a QSO that is sitting in your scratchpad) or <**Quit anyway**> to abandon the scratchpad info and close Logger32.

Q. Can I change the scratchpad column widths and sequence?

A. Yes. The process is the same as for the logbook, worked/confirmed table and other 'grid' tables in Logger32:

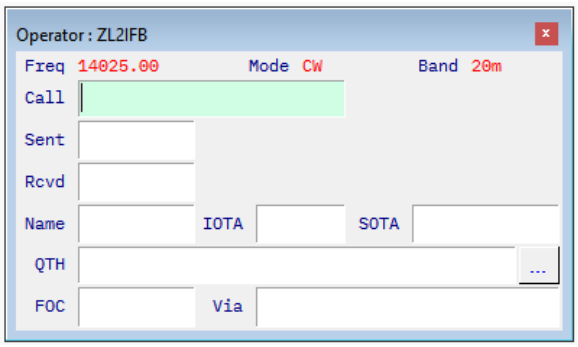
- Display the column headings using **Setup ⇨ Appearance ⇨ Show column headers**.
- Position the mouse pointer at the [invisible] border *between* column headers until it changes to a double-headed arrow, then click and drag the border left or right to narrow or widen the column to the left of the border.
- Click and drag a column header left or right towards another column border to change the column sequence.
- To resize the rightmost column, you can't simply stretch it to the right since it is blocked by the right edge of the window[133]. Instead use a combination approach, a workaround: drag the column to the left, resize it, and then drag it back to the right. Alternatively, using **Setup ⇨ Appearance**, tick to display another temporary column (which gets added on the right), resize what is now the right-but-one column, then use **Setup ⇨ Appearance** again to untick and hide the temporary column you just added.

[133] The usual double-ended resizing arrow appears if you click the righthand edge of the rightmost column header, but you'll find you can only narrow that column, not expand it.

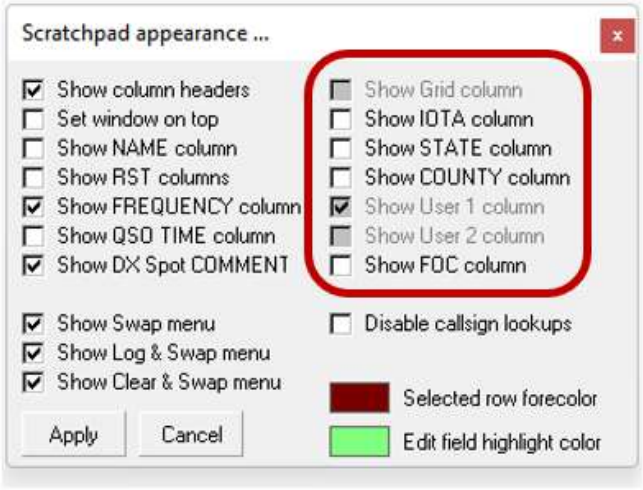
Copyright © 2026 Robert C. Furzer K4CY

Q. Why are some fields grayed out in the scratchpad Setup ⇨ Appearance menu?

A. Only those user fields that are selected in the log entry pane can be selected for the scratchpad.



▲ I am using log entry pane fields for IOTA reference, state, county and FOC number, but not grid, User 1 or User 2 ▶



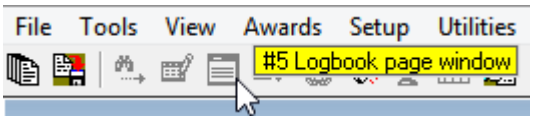
Your selection is probably different.

Copyright © 2026 Robert C. Furzer K4CY

8 The logbook

> "The written word endures,
> the spoken word disappears"
>
> *Neil Postman*

The logbook is central to the way Logger32, or indeed any amateur station, works. It is a relatively formal record of the contacts that have been made from the station – who has been contacted, when, on what bands/frequencies and modes – and may include a wealth of related information such as the reports exchanged, the names and locations of the stations contacted, notes/comments, the QSL status, distances, type of propagation and more. Open it from <**View**> or toolbar ◀ icon #5.



Given the wide variety of users, use cases and hence requirements, and the complexities of electronic logging, radio control, DX chasing, QSLing *etc*., there are bound to difference between individual hams. **How you use *your* logbook is down to *you***, although you may be legally obliged to record certain things in a certain way by your license. Logger32 doesn't force you to do it one way but aims to cover all the possibilities with lots of configurable options and flexibility, combined with powerful functionality that far exceeds what you can reasonably achieve with a handwritten paper log book.

8.1 Logbooks and operators

8.1.1 One callsign + One operator = One logbook, right?

If you are the sole operator of your station, always using your one allocated callsign from your shack, a single logbook makes perfect sense, doesn't it? Well yes it does … but things may not always be quite so simple in practice, for example if you:

- Operate /M mobile or /P portable, particularly if you operate some way from the shack, maybe in a different grid square, county or state.
- Go on a holiday-style DXpedition to a different location, perhaps abroad (a different DXCC entity) using an overseas callsign or your own with the CEPT prefix[134].
- Operate QRP, signing /QRP or "slash QRP"[135].
- Move home – probably a different grid square, county, state and maybe country (or rather DXCC entity).

[134] I was ZL/G4IFB for a while until I got my ZL callsign. I choose *not* to log them in my G4IFB logbook.

[135] Whether or not to log /QRP and other variant callsigns is a contentious issue. Individuals have different preferences, and may be constrained by the license terms. Personally, I have a simple rule: I do my level best to log whatever callsign you send me. If you sign portable, I log portable, and QSL accordingly.

Copyright © 2026 Robert C. Furzer K4CY

- Let a licensed friend or visitor borrow your shack to work some DX, using his/her own callsign … or go to someone else's shack or the radio club to make a few QSOs there under your callsign.
- Upgrade your license or otherwise obtain a new personal, vanity or contest callsign.
- Operate from home on behalf of a club or special event using the club or special event callsign.
- Use a special prefix or suffix for some reason (*e.g.* VK hams may use the AX prefix on Australia Day, and hams reaching a significant anniversary sometimes sign /20, /30 or /whatever if their licenses permit).
- Join the Logger32 beta test crew to check out the callsign validation and lookup functions, inventing and logging a bunch of fake QSOs purely for test purposes.

Logger32 has the capability to log QSOs using different callsigns or callsign variants in the same or different logbooks. You have choices. To create a new logbook, see below.

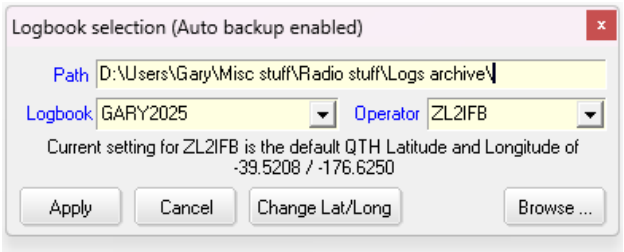
Some of us maintain consolidated 'master logbooks' for all our QSOs made with our base callsigns, portable variants and special event callsigns allocated personally to us: the point is that we personally made all those QSOs. The callsigns we used for individual QSOs are recorded in the log under the <**Operator**> field. We may import single-operator contest QSOs logged in dedicated contest logging software such as N1MM+ into the master log to update our award statistics and manage the QSLs. With a bit more work, we could filter our own QSOs away from those made by other club members in a club or contest log, then import those QSOs into our master logbook.

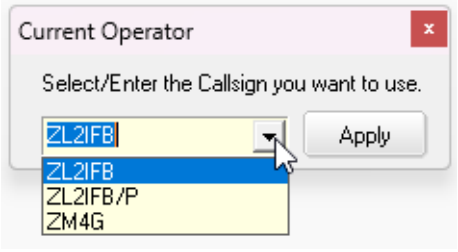
Logger32 can handle multiple logbooks for one callsign, for instance if you were a KP4 ham in Puerto Rico who retired to Florida, retaining and using the exact same KP4 callsign. For DXCC purposes, Puerto Rico and mainland USA are distinct entities, hence QSOs made from each place count for DXCC awards independently: maintaining Puerto Rico *and* Florida logbooks makes it easy to distinguish the QSOs and track your DXCC award status for both places. You can open the Puerto Rico logbook to check and respond to QSL requests for QSOs made before you retired to Florida, and with a few clicks revert to the Florida logbook to log new QSOs from your condominium.

A QSL manager will typically manage QSLs on behalf of numerous stations, plus his/her own logs: using separate logbooks for each managed callsign avoids getting into an almighty muddle, whereas (used with care) a consolidated master log makes it easier to search for and check the details on individual QSOs, particularly when QSLs arrive that are ambiguous or plain wrong.

8.1.2 Using multiple logbooks and station callsigns

You may be using several callsign variants on-air (*e.g.* home call, /P portable and /M mobile) and possibly several distinct callsigns (*e.g.* an everyday callsign, a short contest callsign and a special event callsign). According to the ADIF standard, the **OPERATOR** field contains the *callsign* of the person operating your station, making and logging contacts.





Swap between logbooks ▲ using **File ⇨ Change Logbook**, or to change operators ▲ using **File ⇨ Change operator** or <**Ctrl+U**> (**U** for **U**ser) from the log entry pane. However, it's also easy to mess this up! Using the drop-down arrow, select a callsign already on the list to start logging

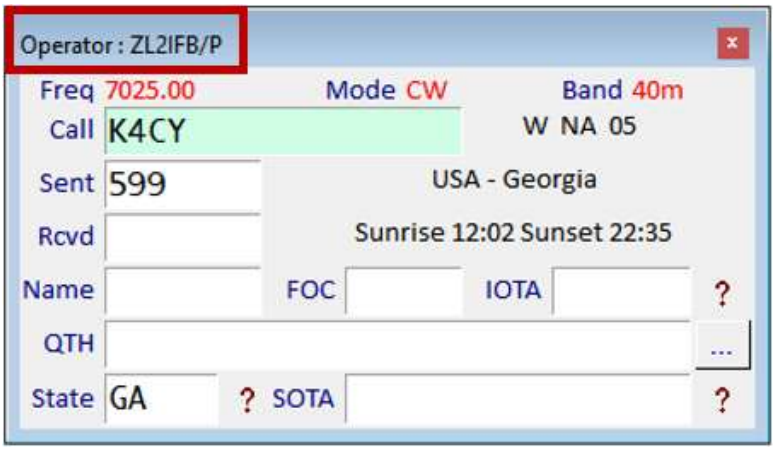
Copyright © 2026 Robert C. Furzer K4CY

QSOs using that station callsign, or type in a new callsign to start using that instead. Click <**Apply**> to save the change.

Hinson tip: operators are associated with station locations, so if you define and use separate operators for your home shack and other locations, be sure to define the station locations accordingly (select an operator and check/update the station QTH, then repeat for other operators). If you don't, there will be errors in the calculated (and logged) distances and bearings to the stations you are working … which you may or may not notice at the time.

Take *particular* care to use the correct "operator" while logging QSOs.

Hinson tip: don't accidentally log QSOs made with your contest or club callsign under your personal callsign, and *vice versa*. **Whenever you change logbook, operator or callsign, check the caption of the log entry pane ▼**



◀ Get in the habit of glancing up here for a ~~massive~~ tiny clue!

Hinson tip: if you make a mistake, you *can* correct the operator for those QSOs by editing the log … *provided* you know which QSOs to correct … so keep notes about your activities from different locations or using other callsigns, specifically the UTC dates (and times if necessary). Informational entries in your log are one way to do this, or a diary/calendar, or for Luddites like me a station notebook made of paper – a place to record radio-related stuff such as propagation, experiments, equipment and mode changes, visitors, awards, unplanned releases of magic smoke and other goings-on in the shack.

8.1.3 Multiple operators sharing a single logbook

If a single logbook is shared by multiple operators (*e.g.* a shared family or club shack), or if the single operator uses multiple callsigns, things get a little more complicated.

The choice of <**Operator**> filter for the award reports becomes important. With <**All Operators**> selected, the statistics relate to the entire log, whereas if a specified operator is selected, only that operator's QSOs are counted.

Copyright © 2026 Robert C. Furzer K4CY

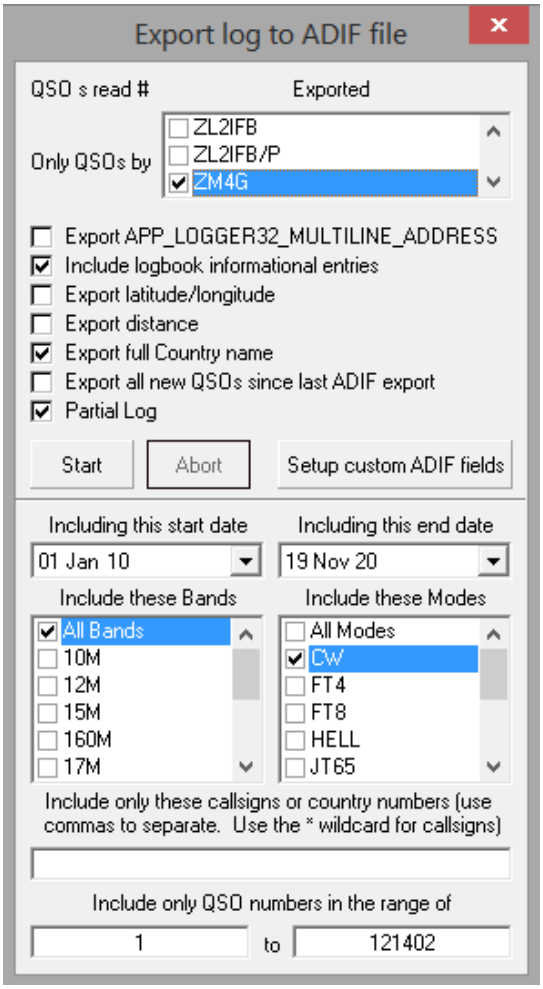
Hinson tip: read about configurations for ideas on how individual operators can set up Logger32 to their own preferences, without messing up anyone else's.

8.1.4 Annual/periodic logbooks

To determine statistics over a specified period such as a year, you *can* set up logbooks for each time period giving each an appropriate name such as "K4CY 2022". Your statistics will only reflect whichever log is loaded: fine if you want to know how many DXCC entities you have worked so far in the year, no good for your all-time DXCC statistics though. For that purpose, you will *also* need to generate and maintain (meaning periodically update) a consolidated log containing *all* your QSOs from a given DXCC entity …

8.1.5 Maintain a master log

If you maintain a 'master' (whole station, all time) log by aggregating all your individual logs, you can still (if you wish) extract subsidiary logs from it containing just the QSOs made by specified **operators** and/or in a specified **period**, **bands** or **modes**, using the filtering options on the **File ⇨ Export logs ⇨ ADIF (.adi file)** function.



In this example, with my master log open, I am generating a partial/subsidiary log containing *just* my CW contest QSOs (logged with ZM4G, my contest callsign, as operator) from 1st January 2010 to 19th November 2020 ▶

Having then changed logbook to a new empty log into which I imported the subsidiary ADIF file, my statistics show how many DXCC entities I had worked as ZM4G and confirmed in CW contests[136] during that period ▼

DXCC_MIXED - All op, QSL & LoTW confirmations, All credits.

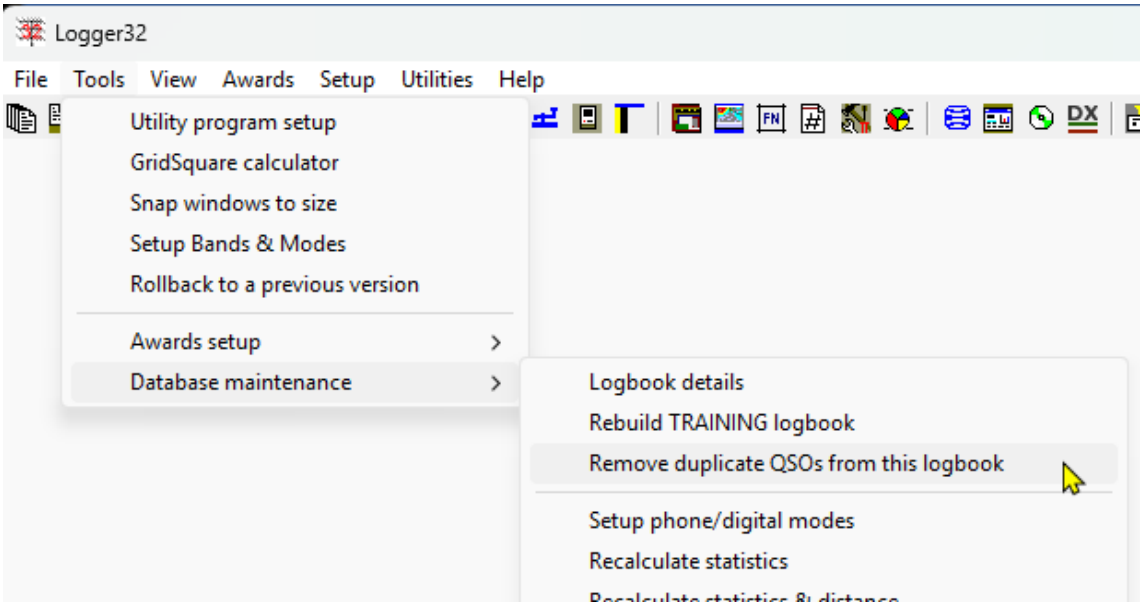
Country	CQZ	ITUZ	160m	80m	40m	30m	20m	17m	15m	12m	10m
All time Countries Worked				95	158	2	170	2	155		112
All time Countries Confirmed				85	137	1	143	1	134		99

[136] Do you notice I appear to have logged some QSOs on 30m and 17m with my contest call, despite (by general agreement) there being no contests on the WARC bands? Those were DX QSOs I made during contests on the other non-WARC bands, having seen juicy DX spots. I *could* have reverted to my everyday callsign but instead I chose to grab them … and quickly get back to the contest!

Copyright © 2026 Robert C. Furzer K4CY

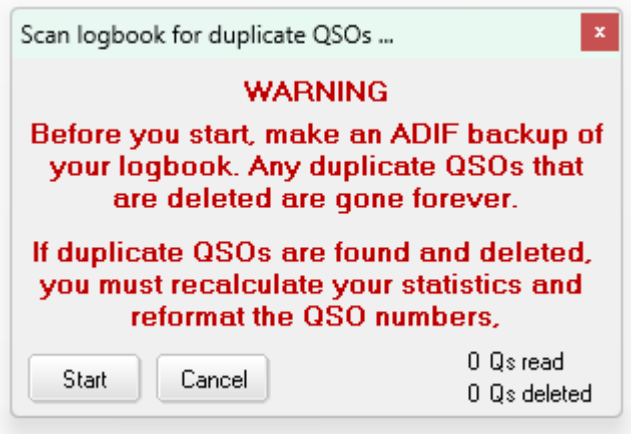
8.1.6 Deleting duplicate QSOs

Under **Tools ⇨ Database maintenance** is an option to remove duplicate QSOs from the open log▼



A duplicate QSO is defined as one adjacent to another in the log, less than 60 seconds apart, with the same callsign, band, mode, operator, state, county, comment, name and gridsquare (if present). If any of those details differ, or if another intervening QSO has been logged between a matching pair, the de-duper leaves them well alone.

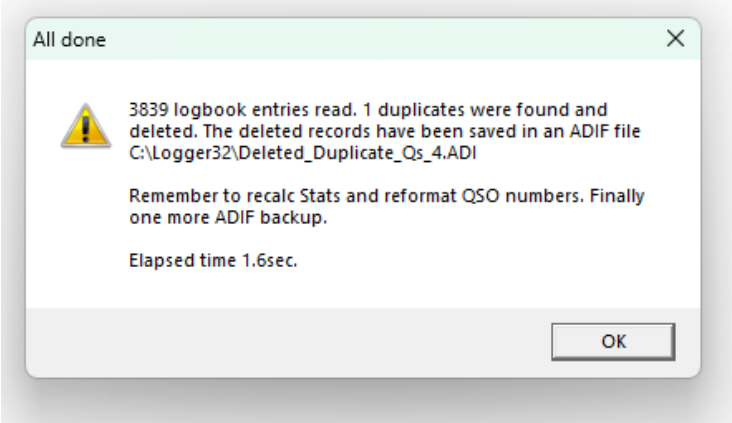
Hinson tip: the de-duper function makes a good job of resolving multiple imports of the exact same ADIF file *e.g.* if we accidentally import the same contest log from, say, N1MM twice or more. *Doh!*



◀ The de-duper will automatically identify and delete duplicate QSO records from the log – a potentially risky process, so the first thing it does is display a stark warning.

After being ***sure*** you have a current ADIF backup of your log, bravely click <**Start**>.

The de-duper runs quickly, ending with a summary and reminder message ▶



Recalc and renumber instructions.

Copyright © 2026 Robert C. Furzer K4CY

8.1.7 Deleting logbooks

If you have any old or unwanted logbooks that you wish to delete[137], you should first:

- Run Logger32.
- Open a logbook you wish to keep (*not* the one you are about to delete), using **File ⇨ Change Logbook**.
- Close Logger32.

That may *seem* pointless but Logger32 saves the name of the logbook in use when it was closed, and opens the same logbook the next time you run Logger32. If you simply close Logger32 while using the logbook that you wish to delete, then delete the logbook files, Logger32 will automatically re-create a blank logbook (containing no QSOs) with the same name when you next start the program. The deleted log files will seemingly reappear, although empty of data.

Next, in File **E**xplorer, browse to the *C:\Logger32* folder and delete the four types of file associated with the logbook you wish to delete, with the extensions: *.isd, .isf, .isl* and *.ism*. Be careful: you may have several logbooks, each with those same four file types. The name of the logbook is the main part of the file name.

Hinson tip: use distinctive names for your logs, and check the file dates and sizes for further clues.

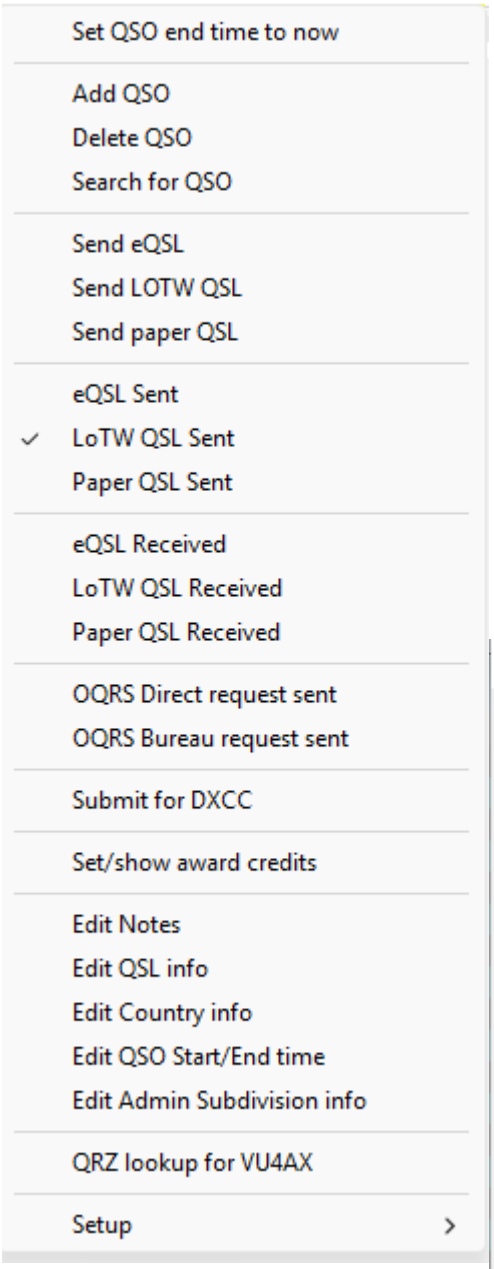
8.2 Logbook right-click menus

As always in Logger32, a *wealth* of relevant functions and options is offered by right-clicking the logbook – right-click either a logbook *column header* or a *QSO* data line (they each have their own right-click menus).

[137] If there is even a remote possibility you *might* perhaps need the logbook later, export it to an ADIF file first and save that somewhere safe, *before* deleting it from Logger32. Once it's gone, it's gone.

Copyright © 2026 Robert C. Furzer K4CY

8.2.1 Logbook *QSO* right-click menu

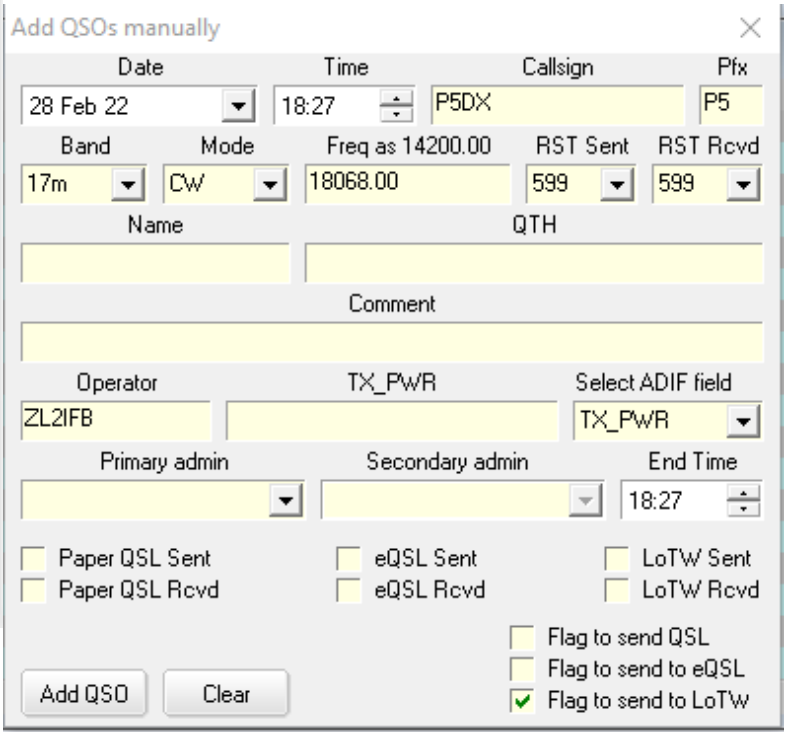


◀ We'll take you through every item on the *QSO* right-click menu first:

- **Set QSO end time to now**: updates the ADIF TIME_OFF field for the highlighted QSO to the current UTC date and time.
- **Add QSO**: logging a *current* QSO is simply a matter of completing the log entry pane in the normal way, of course. However, if you want to log a QSO made previously (maybe scribbled on a scrap of paper while at the radio club or going somewhere in your car), you can either use the log entry pane and then edit the logged QSO, or complete the **Add QSO** form from scratch.

Hinson tip: with the focus on the log entry pane, you can also open the same **Add QSO** form using the shortcut <**Ctrl**+**M**> (that's **M** for **M**anual logging).

Adding an *historical* QSO [138] involves completing this form ▶

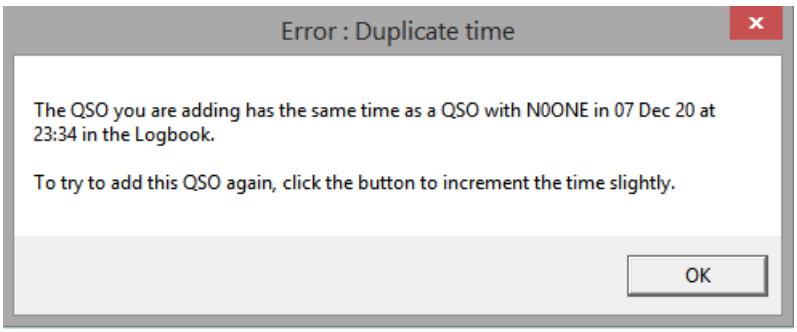


Hinson tip: the DXCC entity for a callsign is normally determined as we are logging a QSO by looking up the prefix in the ARRL DXCC list. For a truly historic QSO, the correct DXCC entity is determined by the DXCC list that was in effect way back when the QSO was made: you may have worked someone in a DXCC entity that was subsequently deleted from the list. Therefore, it pays to check/update the DXCC entity number shown on the form having selected the <**DXCC**> ADIF field.

[138] If you have more than just a handful of QSOs to enter, you may prefer Fast Log Entry by DF3CB to generate an ADIF file that then can be imported into Logger32. It is very efficient and quite easy to use.

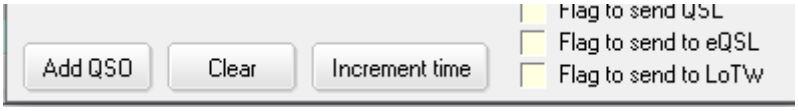
Copyright © 2026 Robert C. Furzer K4CY

Select <**Flag to send to QSL|eQSL|LoTW**> to include the new QSO in the next QSL file exports. These options and most fields on the Add QSOs form are sticky – they are retained for the next QSO after you log one. If you change them, the new values stick.

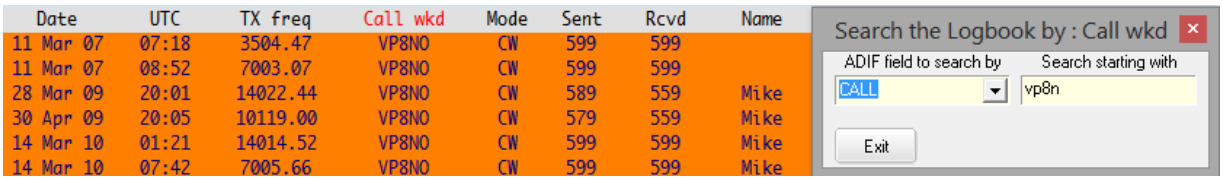


◀ If you attempt to log multiple QSOs at *identical* times, Logger32 objects with this error message.

After <**OK**>ing the error, you may click the <**Increment time**> button that appears ▶ at the bottom of the form in order to nudge the time value one second forward *and* log the QSO. Alternatively, manually adjust the time so that is not exactly the same as a QSO already logged, and then click <**Add QSO**> in the normal fashion.



- **Delete QSO**: just in case you clicked this accidentally, Logger32 asks if you *really* meant to delete the right-clicked QSO from your log. Click <**Yes**> to go ahead and send it to the bit-bucket or <**No**> to keep it after all. You won't be asked if you really *really* meant it. After deleting a QSO, the log display scrolls down to the end, showing your most recent QSOs.
- **Search for QSO**: with the log search form open, choose which ADIF field to search from the drop-down list (presented in the same sequence and with the same titles as the columns in your logbook) and start typing your search term. The search form caption shows the column title you have chosen for this field in your logbook ▼



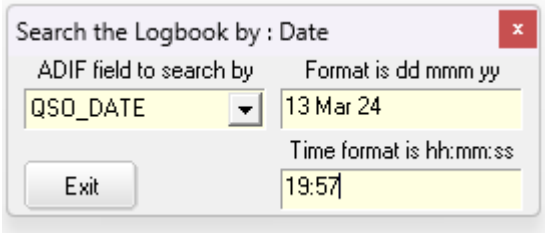
Watch the logbook as you compose your search. The logbook is sorted and filtered as each piece of information is entered into the search form. For example:

  - When searching for the callsign of someone you have worked[139], selecting the CALL field instantly sorts the whole logbook alphanumerically by callsign (the column heading goes red), then the logbook display zooms-in on QSOs with calls starting with the characters you have typed into the search term field.

> **Hinson tip:** if you receive a QSL card from a station whose callsign is not in your log, it is worth checking your log around the date and time stated for a busted callsign.

[139] The previous QSOs pane shows the result of this search as you are logging a new QSO with the same station, so that's an even easier way to find your past QSOs.

Copyright © 2026 Robert C. Furzer K4CY



- The QSO_DATE ADIF field makes it easy to search for a QSO at or after the UTC date and time specified, for instance when checking a QSL card received ▶
- Searching for a well-known QSL manager (such as UA4HWX) in the QSL_VIA field shows all your QSOs with DX stations that use that manager, provided you have recorded the manager's callsign for all applicable QSOs in your log.

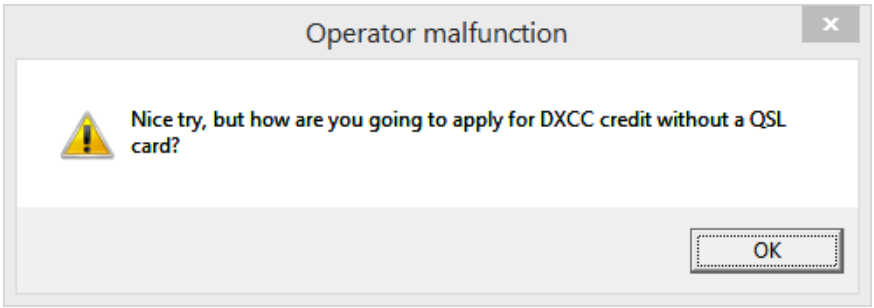
Hinson tip: you can QSL them individually … but why not prepare a batch of QSL requests to the manager for *several* managed DX calls to share the postage costs? Any extra greenstamps will, I'm sure, be welcomed by the hardworking QSL manager.

- If you search on the IOTA field starting with NA-001, the log will sort the log by IOTA references, then position itself at NA-001. **Note:** only QSOs with data in the IOTA field are shown *(you are no longer looking at the complete logbook!* It's all there, just part-hidden*)*.

Once you have entered the search criteria, you can move the search dialog aside to work with the log. Clicking a header to re-sort the logbook on a different field changes the ADIF field on the search form.

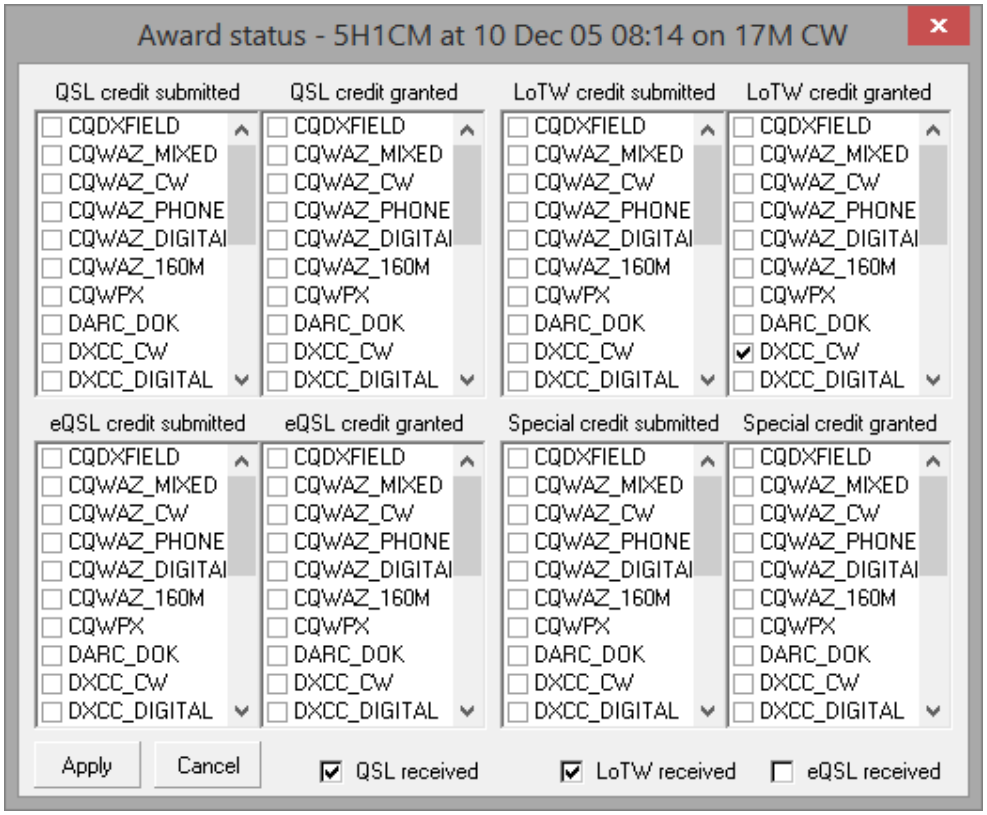
When you finish searching and exit the search form (or click the corner X), or if you edit and update a QSO, the logbook reverts to how it was before you started searching: the sorting and filtering are removed as the search form disappears. All your logged QSOs can be seen again.

- **Send eQSL|LoTW|paper QSL**: click to flag (tick) this QSO to be included in the QSL file exports.
- **eQSL|LoTW|Paper QSL Received**: click to record receipt of a confirmation for this QSO, and hence for this DXCC entity on this band and mode. C for Confirmed appears in the relevant DXCC award reports but remember that only LoTW and paper QSLs are valid for DXCC.
- **OQRS Direct|Bureau request sent**: if you ask for a direct or bureau QSL via an online QSL service (such as Club Log's **O**nline **Q**SL **R**equest **S**ervice), you can simply flag that request here. See also <**Edit QSL information**> further down the same menu for additional options.
- **Submit for DXCC**: sets a flag in your log indicating that you intend to submit this QSO for credit towards a DXCC award. If the QSO is unconfirmed, Logger32 warns of an 'Operator malfunction' and poses a rhetorical question ▼



Copyright © 2026 Robert C. Furzer K4CY

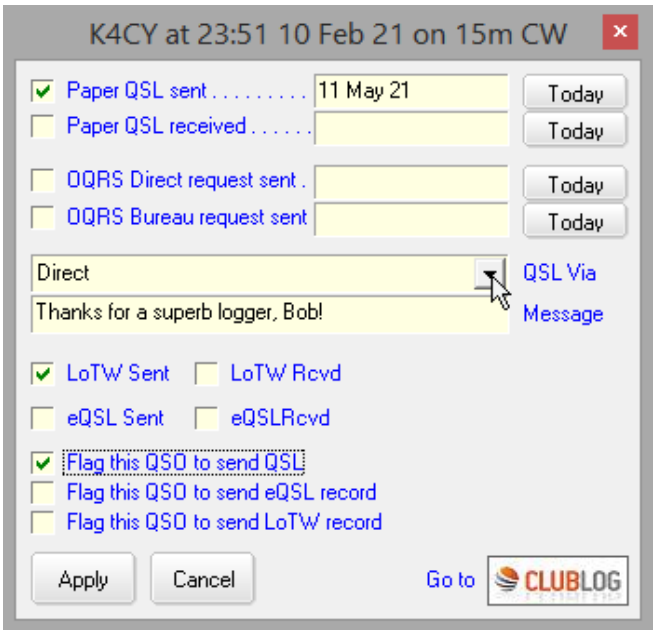
- **Set/show award credits**: shows whether you have received confirmation/s for the QSO, and whether it has been submitted or credited towards DXCC *etc*. ▼



The award status table is interactive: click to select or deselect any item to update the status, then click <**Apply**> to save your changes (or if you change your mind, click <**Cancel**> to scrap the changes) and close the form.

- **Edit Notes**: any existing notes associated with the right-clicked callsign are displayed along with the QSO date. To enter or edit a note for a particular QSO, type the information into the text box then click <**Apply**> to save it.
- **Edit QSL info**: enter or update the QSL details for the right-clicked QSO ▶

  When clicked and ticked, the top four selector boxes automatically fill in the empty date boxes with today's UTC date[140] but you can edit them … and revert to today's UTC date again by clicking the <**Today**> button if you change your mind.



[140] The 'sent' dates are useful because It may take a busy DX operator or QSL manager days, weeks, months, even *years* to process and send a QSL. If the anticipated QSL hasn't arrived within, say, six months to a year of your making the request, it is not unreasonable to follow up by email or try again in case your original request got lost – especially if it is a 'new one'. Please be considerate though: you are not the only ham!

Copyright © 2026 Robert C. Furzer K4CY

The <**QSL Via**> drop-down list offers the following options:

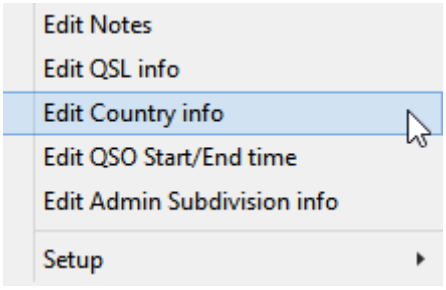
- **Bureau** – send your card through the QSL bureau system, and wait very patiently …
- **Direct** - post your QSL card to the station, preferably with sufficient greenstamps (cash) for them to buy the stamps needed for a reply.
- **OQRS direct request sent** – select this after you have requested and paid for a direct QSL through an online QSLing system such as Club Log's **O**nline **Q**SL **R**equest **S**ervice supporting numerous DX stations.
- **OQRS bureau request sent** – if you have requested a free bureau QSL card via an online QSLing system.
- **OQRS direct request received** – if the station has used an online QSLing system to request a direct QSL card from you.
- **OQRS bureau request received** – if the station has asked for your QSL card via the bureau using an online QSLing system.
- **Manager direct** - post your card to the station's QSL manager, preferably with sufficient greenstamps for the manager to buy the stamps and reply.
- **Manager by bureau** – send your QSL card to the DX station's manager via the QSL bureau system and wait very patiently in the hope of getting a card back.
- **QSL service** – if you use some third party QSO confirmation service.
- **email request** – you email the QSO information to the DX station or manager and perhaps send some money via PayPal, requesting a confirmation.
- **Auto QSL** – some DXpeditions and special event stations declare they will "QSL 100%", often adding "No need to send us your card".

You can either manually enter data in the field, choose an option from the drop-down list, or combine the two *e.g.* select <**Manager direct -** > from the drop-down box, click after the hyphen, then type the QSL manager's callsign to complete the <**QSL Via**> field entry.

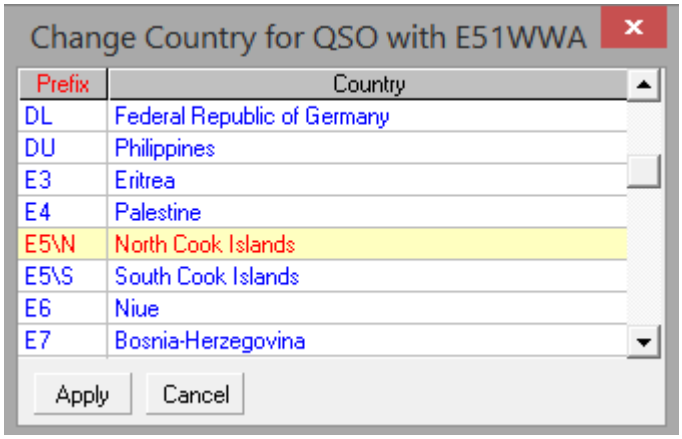
Logger32's auto-updater attempts to provide easier and smoother access to Club Log's **O**nline **Q**SL **R**equest **S**ervice. The Edit QSL info form has four OQRS-related options under the QSL_VIA drop down list. Clicking the Club Log logo takes you directly to the Club Log search page for the station you selected in the logbook. There you can query your QSOs with the station and, with the click of a button, go to the corresponding OQRS page if available. If the station does not offer an online log and OQRS at Club Log, you will be taken instead to the Club Log page that lists all the available DXpedition logs.

Copyright © 2026 Robert C. Furzer K4CY

- **Edit Country info**: ▼

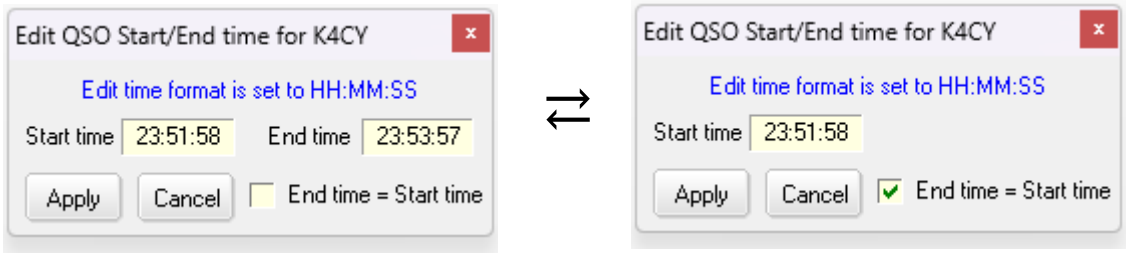


Click to select a different DXCC entity (country) for the QSO[141] then click <**Apply**> to commit the change ▶



- **Edit QSO Start/End time**:

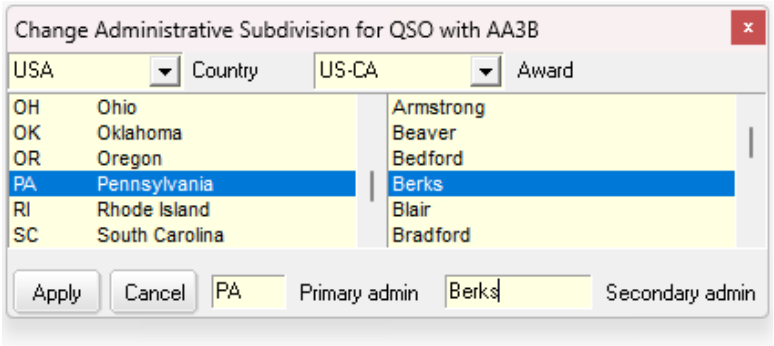
▼ Use this form to adjust the QSO start and/or end times[142].



Both times are set identically if you tick <**End time = Start time**> ▲

- **Edit Admin Subdivision info**:

This form lets you choose a different primary or secondary administrative subdivision (*e.g.* state, oblast, county …) for the QSO ▶



Hinson tip: the same form pops up if you click the state or county field for a QSO in the logbook.

- **QRZ lookup for** [callsign]: displays the QRZ.com entry for this callsign (if any) in your browser.
- **Copy QSO to parallel logbook**: sends the right-clicked QSO to the other system if, for some reason, it didn't find its own way there.

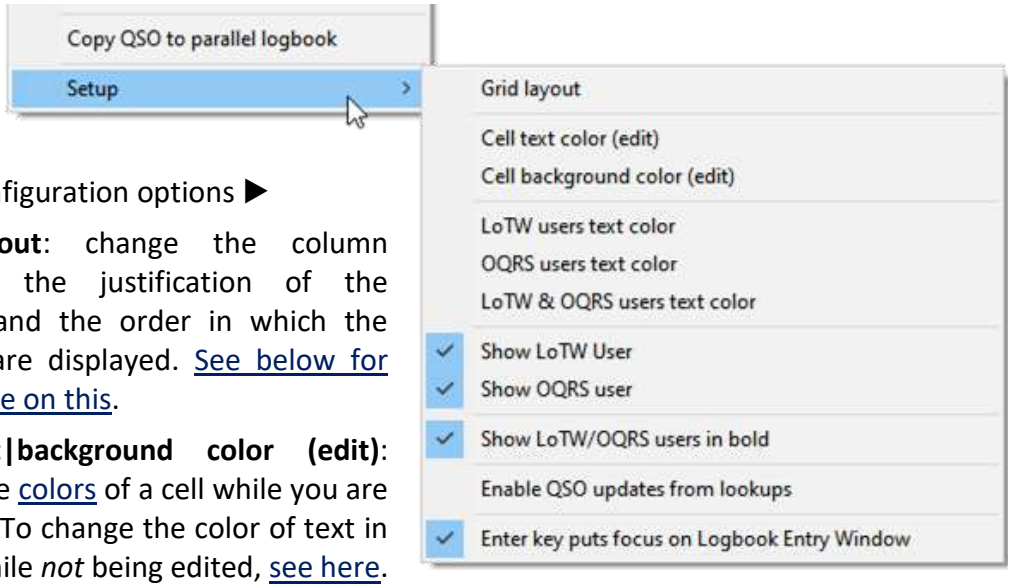
Hinson tip: this menu option only appears if you are currently using the parallel logging function.

[141] This is useful if you work someone whose prefix suggests a different location to where they actually are. For example, you may have worked someone using a KH2 prefix (implying Guam) actually operating from Florida (mainland USA), or an E51 station in the remote North Cook Islands rather than one of the more accessible beauty spots in the South Cooks.

[142] The seconds value *must* be set, even if you choose not to show seconds in your logbook.

Copyright © 2026 Robert C. Furzer K4CY

- **Setup**: at the bottom of the logbook's right-click menu, <**Setup**> opens a submenu of configuration options ▶

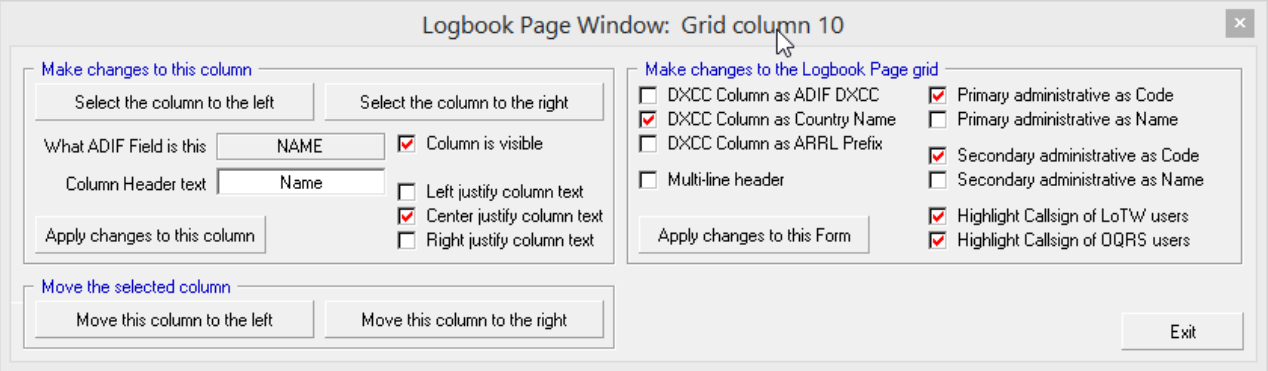


  - **Grid layout**: change the column headings, the justification of the columns and the order in which the columns are displayed. See below for much more on this.
  - **Cell text|background color (edit)**: change the colors of a cell while you are editing it. To change the color of text in the log while *not* being edited, see here.
  - **LoTW users text color**: choose the text color for logged stations that [only] use LoTW.
  - **OQRS users text color**: choose the text color for logged stations that [only] use Club Log's **O**nline **Q**SL **R**equest **S**ervice.
  - **LoTW & OQRS users text color**: sets the text color for stations that use *both* LoTW *and* Club Log's **O**nline **Q**SL **R**equest **S**ervice.
  - **Show LoTW|OQRS user**: these options apply the selected text colors to either or both types of callsign in the log.
  - **Show LoTW/OQRS users in bold**: as well as the configurable text color, LoTW and/or Club Log's **O**nline **Q**SL **R**equest **S**ervice users' callsigns can be shown in **bold**.
  - **Enable QSO updates from lookups** saves information from callsign lookups when QSOs are logged – including any arriving via UDP from JTDX|WSJT-X. See the Callsign lookups chapter for more.
  - **Enter key puts focus on Logbook Entry Window**: having finished editing a QSO record in your log, when you press enter to commit the change, the focus can either move to the log entry pane to log your next QSO (with this option ticked), or to the next QSO in your log to review/change another logged QSO (unticked).

Copyright © 2026 Robert C. Furzer K4CY

8.2.2 Logbook *header* right-click menu

As a simpler alternative to the forbidding and awkward gray QSO grid layout configuration form, right-click a column heading in the logbook, previous QSOs or additional information windows to open a configuration form ▼



The left side of this form concerns the column whose heading you right-clicked (specifically column 10 in the example above, as indicated in the window caption). You can:

- Change the column heading.
- Make the column visible or hidden.
- Justify the text to the left side, center or right side of the column.

Click <**Apply changes to this column**> to put the changes immediately into effect: the logbook is instantly updated, making it easy to check whether the changes are working as you intended.

You also have buttons to:

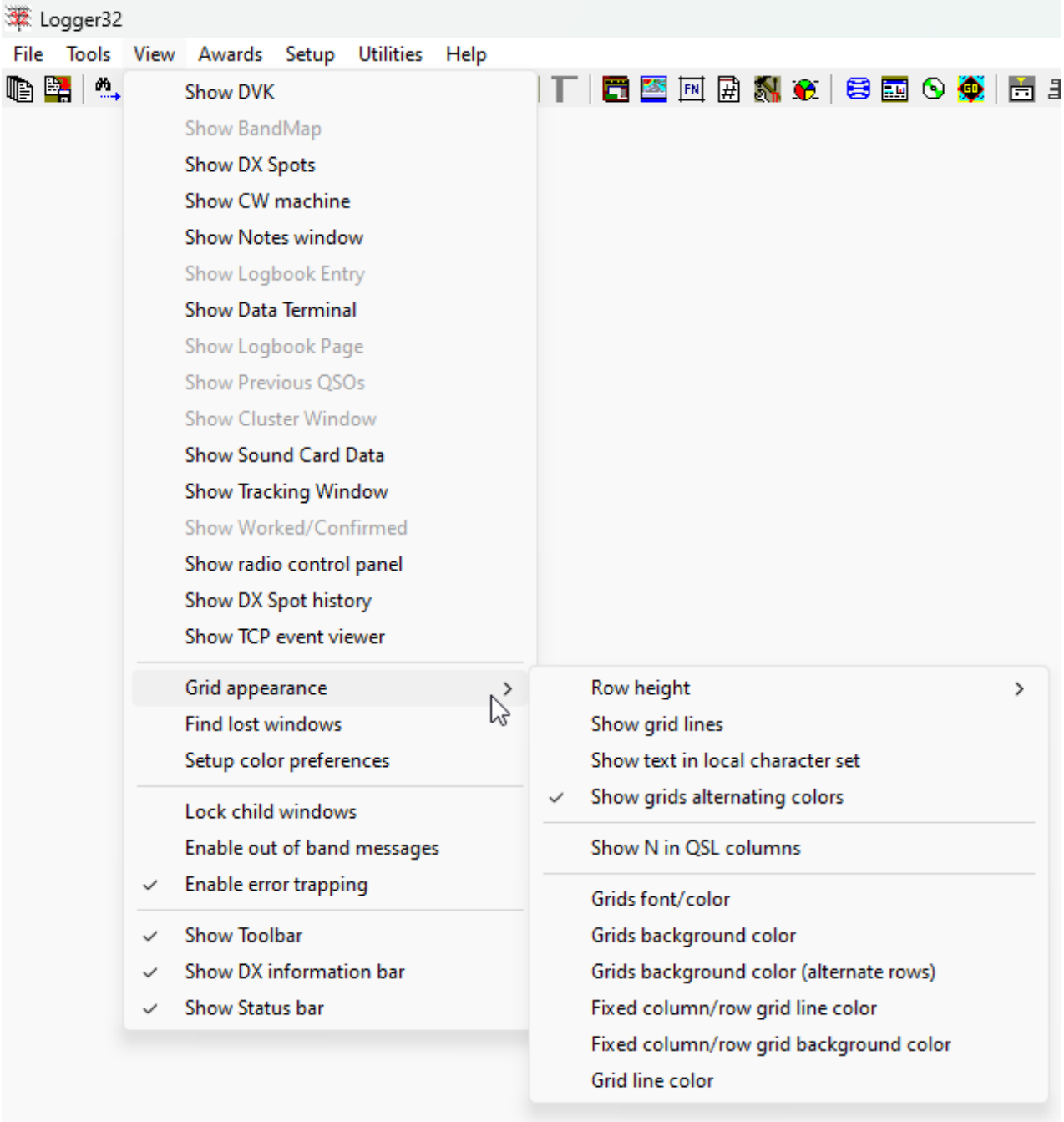
- **Select the column to the left|right** relative to the currently-selected one. This lets us configure the entire logbook grid, column-by-column. Note that although some columns may not be set to 'visible', they can still be configured here: to see the results in your logbook, tick <**Column is visible**>.
- **Move this column left|right** – a change that takes immediate effect so, again, if you are looking at the logbook, you can see whether you are making things better or worse.

On the right side of the form are further layout and appearance options. Having changed any of these, click <**Apply changes to this Form**> to see the effects.

Copyright © 2026 Robert C. Furzer K4CY

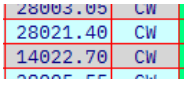
8.3 Configure QSO tables (logbook and other 'grids')

Several panes in Logger32 (the logbook, previous QSOs, generic QSOs, worked/confirmed table, notes and NCDXF beacons) share a common 'grid' (tabular) layout that can be customized – *not* through a right-click <**Appearance**> option as an experienced Logger32 user might naturally expect but instead using **View ⇨ Grid appearance** ▼



The settings here affect all the tabular panes in much the same way:

- **Row height**: on my display, a row height of 255 twips spaces out the rows nicely. Experiment with the value on your system to find a setting that suits you, sir.
- **Show grid lines:** Logger32 can draw lines separating data rows and columns ▶ The line color is configurable using the bottom menu item – see below.



- **Show text in local character set**: use accented characters and hanzi/kanji if you will.

Copyright © 2026 Robert C. Furzer K4CY

- **Show grids alternating colors**: contrasting row colors (zebra stripes) make it easier to read along each row.
- **Show N in QSL columns**: do you prefer to see an "N" if there is no QSL, or just a blank space?
- **Grids font/color**: choose the font, size and color for the QSO data displayed in the tables.

> **Hinson tip**: to increase the readability and hence reduce the chances of mistakes within callsigns, browse fonts.google.com for a *huge* selection of free fonts, including some with slashed or dotted zeroes *e.g*. IBM Plex Mono – a dotted zero *sans serif* monospaced font that (on my screen, through my eyes) is clearly readable in 10 point size.

- **Grids background color**: if you don't use alternating colors, this sets the background color of every data row in your tables. If you do, it sets the background color for alternating rows …
- **Grids background color (alternate rows)**: … while this sets the color of the intervening data rows.
- **Fixed column/row grid line color**: this sets the color outlining the top row non-scrolling column headers and (if applicable – *e.g*. in the worked/confirmed table) the lefthand side row labels as well. The fixed column/row lines (bright green here) are always shown ▼
- **Fixed column/row grid background color**: sets the background color for the non-scrolling column/row headers.

Date	Start	TX fr
25 May 24	04:38	28045
25 May 24	04:52	28032

- **Grid line color**: colors the lines within the bodies of the tables if shown *e.g*. in red here ▲

> **Hinson tip**: the logbook and worked/confirmed panes normally show the effects of color changes instantly as soon as you select a new color and click the <**OK**> button[143]. It may take a few tries and a while using Logger32 to find the perfect combination of colors for your screen and eyes. Various shades of gray work for me. The settings are saved automatically in *C:\Logger32\Logger32.INI*. Once you have found your sweet spot, and ideally *before* you start fiddling with the settings, save a copy of *C:\Logger32\Logger32.INI* somewhere safe so that you can revert the settings if your changes don't work out and you decide to start over.

8.4 QSO grid columns and layout

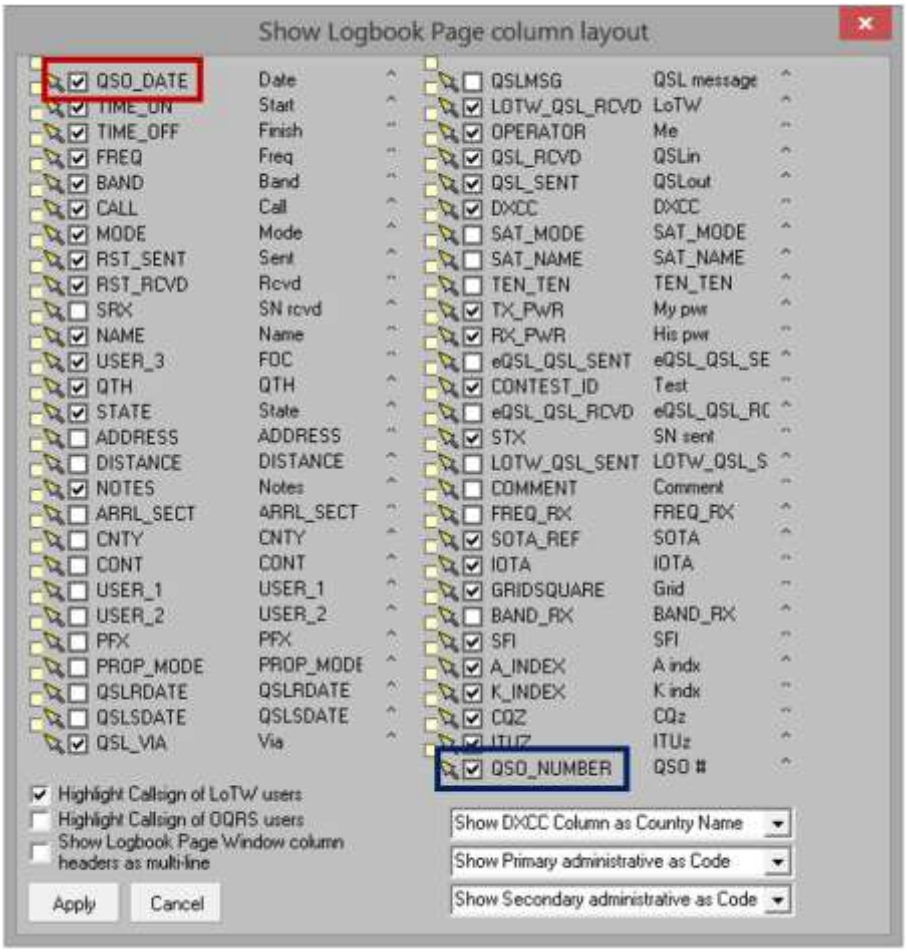
The screen layout of the logbook, previous QSOs and additional information windows can be customized by right-clicking any QSO in the window, then clicking **Setup ⇨ Grid Layout**.

[143] In a beta test, changing the color settings had no apparent effect at first – until I stopped and re-started Logger32, whereupon everything worked normally. The classic turnitoffandonagain did the trick for me.

Copyright © 2026 Robert C. Furzer K4CY

8.4.1 Displaying fields

The forbidding gray form shows all available logbook data fields, plus a few display options and buttons at the bottom ▶



The sequence runs top to bottom in the left column, then the right.
The top left field (QSO_DATE in my case, outlined in red) corresponds to the left-most column on my logbook display, while the bottom right field (QSO_NUMBER boxed in blue) is for the right-most logbook column.

Hinson tip: if you find this gray form *too* forbidding, there is another way to configure the logbook.

Each column has:

- **A 'move field' target and pointer:** the small yellow boxes and arrows ▶ 
- **A 'display field' checkbox**: the white boxes, some of which are ticked ▶
  Simply click the checkbox to toggle the display of that field on (shown) or off (hidden).
- **ADIF field name**: *e.g.* "QSO_DATE" and "QSO_NUMBER" (see below).
- **Column heading**: *e.g.* "Date" (see below).
- **Text alignment indicator**: "<" means the text in that column will be left-aligned; "^" means centered; ">" means right-aligned. To change the text alignment for a column, either right-click the symbol for that field and choose from the three options, or else click and then overtype the alignment symbol.

8.4.2 ADIF field names

In the example above, the text outlined in red ("QSO_DATE") is an ADIF field name as formally specified in the ADIF standard. All the 'official' ADIF fields are fixed[144] in ALL CAPITALS, using underscores rather than spaces between words LIKE_THIS. That convention makes them reasonably distinctive.

[144] To remain compliant, they cannot be changed … unless/until the ADIF standard itself is changed.

Copyright © 2026 Robert C. Furzer K4CY

When importing and exporting ADIF data, the ADIF field names allow Logger32 and other ADIF-compatible programs to identify different kinds of data within each QSO record. This is how the programs all 'know' that, for instance, the "CALL" field in an ADIF QSO record is always *supposed*[145] to contain the callsign of the station contacted, not the operator's callsign or some other information such as "CQ", "599" or "Fred".

8.4.3 Column headings

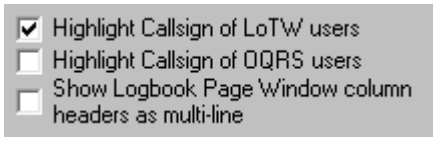
These are the titles shown at the top of each column of your logbook, as displayed. In the logbook layout example above, I have configured my logbook to display the heading "Date" in mixed case, for instance, rather than the default "QSO_DATE" all shouty in CAPITALS with_the_underscore.

It is not intuitively obvious that **you can edit the entries in this column of the configuration form**: whereas the ADIF field name is used by default, simply click the text on the gray form and change it as you wish[146].

Even though you can change the text of the column headers, the correct ADIF field name is always used in ADIF file exports. The three USER fields mentioned above will be exported in the ADIF file similar to <APP_LOGGER32_USER_1:n>[n characters of data go here].

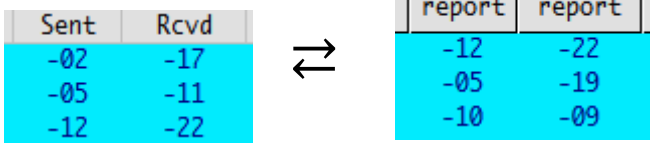
If you need to log custom data, it is better to use the Notes or Comments field, or for data that you would like to track in a Simple Award, use the USER_1, USER_2 and/or USER_3 fields.

8.4.4 Display options



◀ At the bottom left of the gray grid configuration form are these three options.

- **Highlight Callsign of LoTW|OQRS users**: logged callsigns are shown in bold if they are present in the LoTW or Club Log **O**nline **Q**SL **R**equest **S**ervice user lists, respectively.
- **Multi-line display**: the column heading row triples in height, and the column names word-wrap if they are wider than the column width ▼



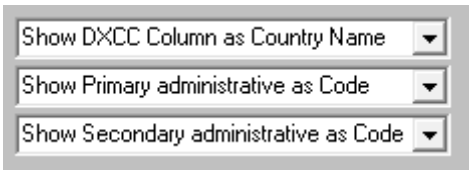
  However, column names containing underscores rather than spaces will not word-wrap, so edit the column name text accordingly.

[145] ADIF-like files generated by naughty non-ADIF-compliant programs may contain virtually anything.

[146] Don't get *too* carried away with this capability though. You *could*, for instance, change the "QSLMSG" column header to "FOC member #", then use that logbook column to record the membership numbers when working your FOC pals … but when you export your log, that field will be named "QSLMSG" in the ADIF data file, not "FOC member #". Thanks to the ADIF standard, programs reading-in that field label will interpret the data content as a QSL message, causing unpredictable and probably unintended results.

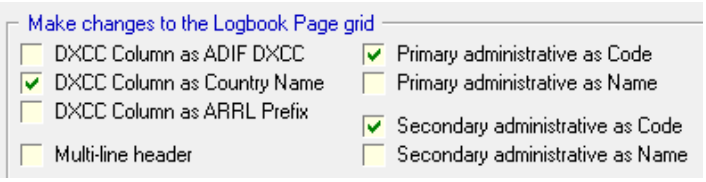
Copyright © 2026 Robert C. Furzer K4CY

8.4.5 Field display options



Three selector boxes at the bottom right of the gray grid ▶ configuration window determine how some data appear in your logbook:

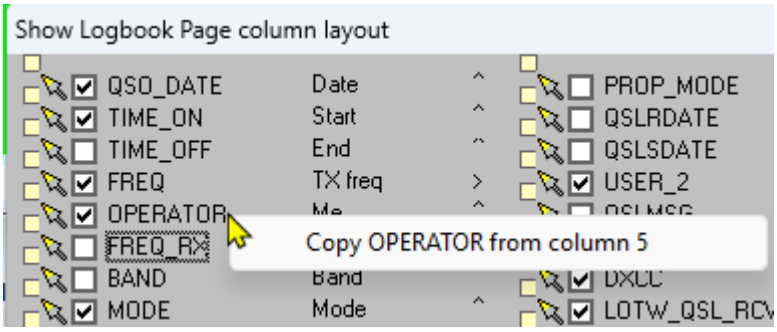
- The top box configures the DXCC column to show the DXCC entity number, country name or DXCC prefix.
- The middle and bottom boxes let you display the Primary and Secondary Administrative fields by name (*e.g.* California) or code (*i.e*. the abbreviation or number *e.g.* CA).



◀ The same options are laid out differently on the right side of the column-by-column configuration form.

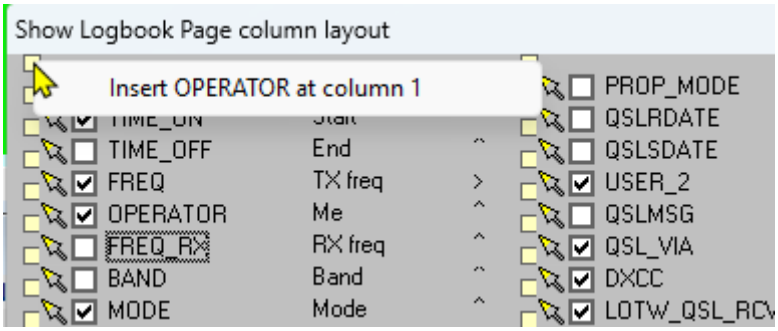
8.4.6 Rearranging logbook columns

Rearranging the order in which logbook columns are displayed is a little tricky using the gray grid configuration form. It requires good eyesight and hand-eye coordination. **To move a given column, drag the yellow arrow to the left of its field name towards the little yellow box between the columns where you want to place it.** For example, to move the MODE column to the left of CALL, point the mouse at the yellow arrow next to MODE, press and hold down the mouse button, then drag the arrow until it touches the box between BAND and CALL which drops it there, and finally release the mouse button.



◀ Alternatively, if clicking and dragging columns is awkward, you can right-click the name of a column you wish to move, then click the line <**Copy [ADIF field name] from column [number]**>.

Now right-click the little yellow box where you'd like to move the column, then click the line <**Insert [ADIF field name] at column [number]**> ▶



When you've made your changes and they look right, click <**Apply**> to save them. If you make a mistake or change your mind, click <**Cancel**> instead to discard the changes.[147]

[147] Bear in mind that if you make five changes and only then click <**Cancel**>, *all five* changes will be lost. If you struggle with the process, it may be better to make, check and <**Apply**> just one or two changes at a time.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: if your tremors or poor eyesight make it too difficult to point and click at those tiny yellow boxes, it is *possible* to meddle with the layout and titles of the logbook columns in the [wndLogbookPage Settings] section of *C:\Logger32\Logger32.INI* using a plain text editor, re-launching Logger32 to see the effect.▼

```
[wndLogbookPage Settings]
Logbook Page Index= 0
Logbook Page Index Column= 1
Column #A_INDEX=50,705,A indx,True,^,,,,,,
Column #ADDRESS=20,705,ADDRESS,False,^,,,,,,
Column #ARRL_SECT=22,705,ARRL_SECT,False,^,,,,,,
Column #BAND=6,705,Band,False,^,,,,,,
Column #BAND_RX=48,705,BAND_RX,False,^,,,,,,
Column #CALL=9,2055,Call wkd,True,^,,,,,,
Column #CNTY=21,1410,CNTY,True,^,,,,,,
Column #COMMENT=47,705,Comment,False,^,,,,,,
Column #CONT=23,705,CONT,False,^,,,,,,
Column #CONTEST_ID=43,1770,Test,True,^,,,,,,
Column #CQZ=52,510,CQz,True,^,,,,,,
Column #DISTANCE=8,1185,Distance,False,^,,,,,,
Column #DXCC=33,3105,DXCC,True,^,,,,,,
```

However, you'll need to figure out the parameters and take care when editing them. **Stash an unmeddled copy of *Logger32.INI* somewhere safe *before* you start meddling, just in case things go horribly wrong** and, while you're at it, stash away a copy of the finished masterpiece too in case you need to reinstall Logger32 at some point.

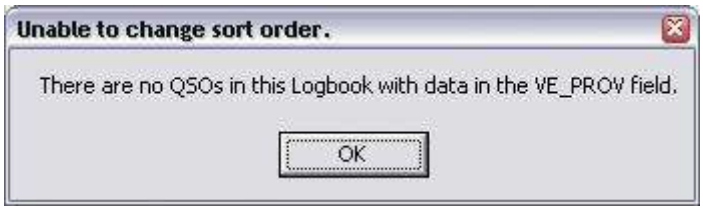
8.5 Sorting the logbook

Click a sortable logbook column heading to display your QSOs in ascending order, sorted on that column. When you select a column to sort by (the date column by default), that column's heading is displayed in red. However, not all logbook columns are sortable (*e.g.* QSO times are not: QSOs sorted by date are *also* sorted by time within each date – chronologically, that is).

Hinson tip: in an idle moment, click along each of the headings in your logbook to discover which columns are sortable and which aren't. Click the **date** column when your curiosity is sated.

When you click a column heading to sort the logbook, **Logger32 only displays QSOs with data in the sorted field**. For example, if you sort the log on the IOTA field, *only QSOs with IOTA references* are displayed. If you have logged 30,000 QSOs but only a precious few have their IOTA references, don't fret if the IOTA-sorted logbook only appears to contain those few QSOs. The remaining QSOs are still there, just not shown until you re-sort by some other field such as the QSO date (which *every* QSO has).

If you sort the logbook on a field that does not contain *any* data, Logger32 displays an error message. Click <**OK**> to cancel the sort and return to the logbook ▶



Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: if it *appears* you have lost most of the QSOs in your logbook, **don't panic!** Almost always, the reason is that you have sorted the logbook by a column for which many QSOs have no values logged. Simply click the **date** column heading to sort the logbook in date and time order as normal: since every QSO *must* have a date and time, every QSO is shown. Phew!

If the view has more rows than can be displayed in the logbook window, a vertical scrollbar appears on the right side. The sort order is maintained as you scroll through the listed QSOs. Similarly, if you choose to display more columns than can be displayed in the logbook window, a horizontal scrollbar appears at the bottom of the window.

Hinson tip: Logger32 automatically saves the current logbook sort field as it shuts down, so if you sort your log by a particular field for a specific reason and then close Logger32, the log will reappear in the *same* sorted order when you next launch Logger32. To avoid consternation and a great gnashing of teeth when your log appears to be all messed up (having forgotten that you had sorted it on some other field the last time you used Logger32), click the **Date** header to reset the log to the default sequence *before* closing Logger32.

8.6 Modifying/editing logged QSO information directly

You can change the data in almost [148] any logbook field simply by clicking an entry and editing/typing in the correct information. After you have made an edit, save it using:

- <**Tab**>: saves the change and moves to the next cell. Logger32 remains in edit mode
- <**Enter**>: saves the change, leaves edit mode and optionally puts the focus on the log entry pane if so configured (right-click the logbook, then look at the bottom of the setup menu).
- <**UpArrow**> or <**DownArrow**>: saves the change, then moves the editing cursor to the same field in the QSO above or below it in the logbook. This can be handy to edit a series of logged QSOs in the same way *e.g.* pasting-in notes such as "Non-contest DX worked during CQ WW DX contest".
- **Click away**: click another logbook field on the same or a different QSO to save the change and edit the other field/QSO.

Additional keys and actions in edit mode (note: these do *not* save any changes made already):

- <**Esc**>: cancels/discards the edit/s, moving the focus back to the log entry pane.
- <**PageUp**> or <**PageDown**>: the edit is canceled and the logbook moves up or down by one page. As you are no longer in edit mode, further page ups or downs have no effect.
- <**Ctrl**+**End**> and <**Ctrl**+**Home**>: jump to edit the rightmost or leftmost field for the same QSO.
- <**LeftArrow**> or <**RightArrow**>: edit an adjacent field in the direction of the arrow.

  Hinson tip: notice that, in contrast to <**UpArrow**> and <**DownArrow**>, the left and right moves *discard* any changes you have just made to the previous column.

- <**Home**> or <**End**>: move to the left or right edges of the present field.

When initially entering a QSO using the log entry pane or Add QSO, the mode *must* be one of those currently defined in your Bands & Modes table. Subsequently, however, you can change the mode

[148] For cells containing key fields such as DXCC entities and primary/secondary administrations, we can *only* select values from drop-down lists. This important integrity control prevents us creating new entries by mistake *e.g.* by typing "Algerian" instead of "Algeria", or "USA" instead of "United States of America".

Copyright © 2026 Robert C. Furzer K4CY

recorded for a logged QSO to any valid ADIF mode (as listed in *C:\Logger32\ADIFModes.txt*) by clicking the mode field in the logbook and editing the text.

If a QSO has been logged with a dubious gridsquare (*e.g*. a station sending a *krazy* grid in his FT# CQ messages, or someone who has chosen the wrong 'station location' and hence grid square in TQSL, confirming it to you through LoTW), you can simply delete the dubious grid or correct it (assuming you know the station's real grid). Either way, Logger32 recalculates and updates the logged distance from your home QTH to the updated grid or to the default location for that DXCC entity when the edited QSO is saved.

If you sort your log by the callsign or DXCC columns, the distances and grids should be *roughly* the same for worked stations in a given DXCC entity, making any truly *krazy* ones stand out *e.g*. ▼

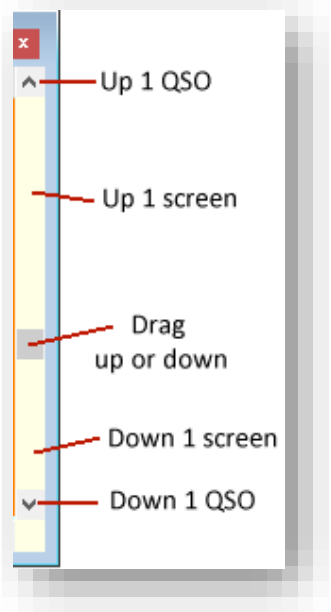
Date	Start	TX freq	Call wkd	Distance	Mode	Sent	Rcvd	Grid
13 Jul 14	09:52	3527.86	ZW2HQ	11663 Km	CW	60	99 labr	GG66
25 Nov 06	07:49	7030.80	ZW5B		CW	599	599	
27 Dec 14	20:28	28021.22	ZW5B		CW	599001	59911	
24 Oct 15	05:49	7074.30	ZW5B	11366 Km	SSB	5932	5911	GG54iq
14 May 20	05:47	10137.14	ZW5STAYHOME	3531 Km	FT8	-12	-25	PF85
07 Jun 20	06:28	3575.03	ZW5STAYHOME	11357 Km	FT8	-07	-15	GG54kl

My first QSO with Brazilian COVID-19 special event station ZW5STAYHOME has somehow picked up a completely wrong grid and hence distance. The second QSO looks about right.

Plotting your logged QSOs on a world map is another way to spot those crazy grids, such as European stations apparently located in Asia, or fixed (non /MM or /AM) stations apparently floating way offshore in an ocean. It is conceivable that what appears to be a *krazy* grid is, in fact, correct but you have somehow logged the wrong callsign and DXCC entity for the QSO (*e.g*. omitted the /MM).

Hinson tip: whenever you edit your log, *try* to avoid 'correcting' QSOs using the wrong values, making things even worse – *especially* if you have no recent log backups to restore (big hint!).

8.7 Navigating the logbook



Navigate through your logbook with the mouse:

- Click the arrows at the top or bottom of the scrollbar to move 1 QSO up or down.
- Click above or below the slider in the scroll bar to jump a screen up or down.
- Click, hold and drag the scrollbar slider up or down to move quickly to a different part of the logbook. Here, QSOs about two thirds of the way through my logbook are visible ▶
- Click a QSO to edit it, more specifically to edit the field you clicked.
- Right-click a QSO for a menu of options.
- Mouse wheel scroll through the logbook, QSO-by-QSO. Hold <**Shift**> while you roll the wheel to scroll faster.

Copyright © 2026 Robert C. Furzer K4CY

If you prefer to use the keyboard, with the focus on the logbook *and* the mouse pointing at the slider area, the <**Home**>, <**End**>, <**Page Up**> and <**Page Down**> work intuitively … but the arrow keys don't (too bad: blame Microsoft[149]).

8.7.1 Logbook fields

- **Date**: defaults to today's UTC date. Change the date by clicking the month, day or year, then pressing the **+** or **-** keys on the numeric pad or the <**Up-arrow**> or <**Down-arrow**> keys. You can also pick a UTC date from the calendar after clicking the arrow to the right of the date.
- **Time**: works much like date – UTC again, as per the ADIF standard.
- **Band and mode**: when you select a QSO mode, the frequency changes to the relevant section of that band[150]. Change it if you wish. If you enter a mode that has a submode (like SSTV), the submode field appears.
- **Address and select ADIF**: these fields operate together. When you change the Select ADIF field, the name on the field immediately to its left changes accordingly, giving you access to every field in your logbook. As you change fields, any data you entered in the previous field is retained and will be added to the QSO when it is saved.
- **Other fields**: several fields are "sticky" - they remember the last data entered, making it easier to add several QSOs in sequence on the same day, mode and band.
    - Select (tick) <**QSL Sent**> or <**QSL Rcvd**> to update the corresponding ADIF fields.
    - Select (tick) <**Flag QSO for QSL**>, <**Flag for eQSL**> and/or <**Flag for LoTW**> for the QSOs to be included in the respective uploads.
    - <**Clear**> empties the non-sticky fields.
    - <**Exit**> instantly closes the form: any data remaining in the fields is discarded without being logged.

8.8 Rebuild (validate and optimize) your logbook

Due to the way database systems manage their data storage internally, it may be worth running validation and optimization routines occasionally to find and tidy up loose ends, or to fix issues (such as 'Send QSL' flags not working or reported errors in the database):

- Spaces left behind after records have been deleted may be removed, and large records may be consolidated, saving space and reducing fragmentation.

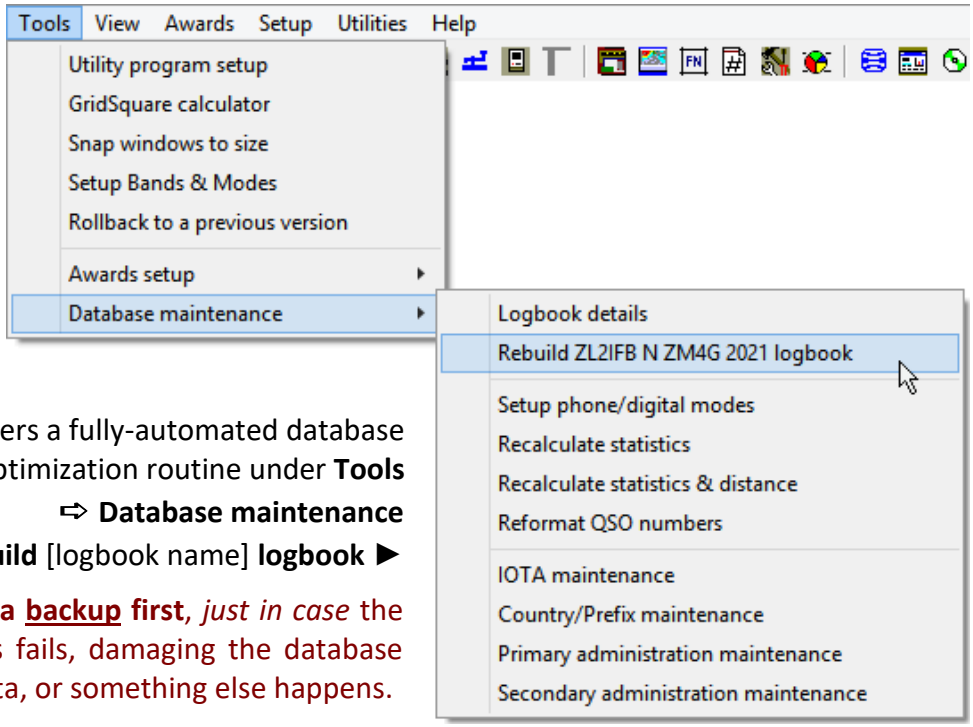
---

[149] The logbook slider is a K4CY special custom design because of a technical limitation with the version of Microsoft software used to develop Logger32, which does not support databases with more than 64k rows. As a result, its appearance differs a little from other Windows sliders. It is 'close enough for government work' though.

[150] When entering the mode, the drop-down list shows *all* the valid ADIF modes, not just those modes currently defined in your Bands & Modes table since you may have made QSOs on legacy modes that you no longer use, or on experimental modes you have not yet added to your Bands & Modes table. Only modes and submodes enumerated in the ADIF standard can be logged *unless* you cheat by adding others to your *C:\Logger32\ADIFmodes.txt* file – but having exported an ADIF file with custom modes, don't expect it to be accepted and understood by other programs, such as LoTW. Non-standard means you're on your own, flying solo, with no safety net.

Copyright © 2026 Robert C. Furzer K4CY

- Tables may be reorganized such that key fields (in particular) are more logically sequenced, speeding up indexed lookups and searches.
- Possibly other database housekeeping tasks, such as finding and dealing with minor data integrity issues, validation failures *etc*.

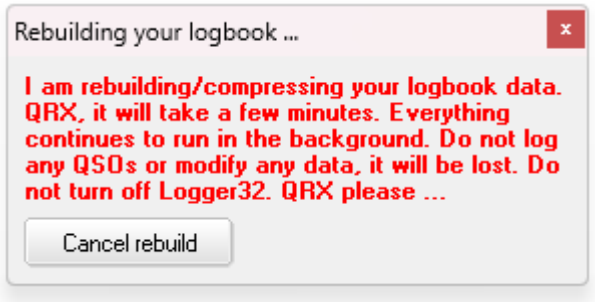


Logger32 offers a fully-automated database optimization routine under **Tools**

➪ **Database maintenance**

➪ **Rebuild** [logbook name] **logbook** ▶

… but … **make a backup first**, *just in case* the rebuild process fails, damaging the database structure or data, or something else happens.



◀ The logbook rebuilder function takes a few minutes to do its job. While the logbook is being rebuilt, Logger32 temporarily suspends the cherry-picker to conserve resources and complete the process as efficiently as possible. Leave it in peace.

Hinson tip: at the start of the process, Logger32 makes a *copy* of the open logbook and starts building a new database using the copy. If the rebuild process fails or is interrupted, the partly-rebuilt new database is abandoned. If the rebuild completes successfully, the original logbook is deleted and the rebuilt logbook is renamed to replace it.

Wait patiently while your valuable log is rebuilt. Have a builder's tea.

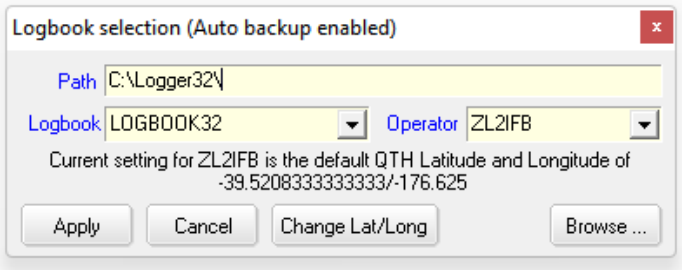
8.9 Changing and creating new logbooks

Logger32 supports multiple logbooks *e.g.* you *might* prefer separate logbooks for:

- Everyday DXing and contesting.
- Personal/individual and club/team calls.
- Different shacks or locations such as mobile and portable operations.
- Special event operations.
- Numerous DX stations managed by a busy QSL manager.
- Testing out Logger32 options, safely, or training/demonstrating Logger32 to your pals.

Copyright © 2026 Robert C. Furzer K4CY

Logger32 maintains the statistics, awards tables *etc*. independently for each logbook. Only one logbook can be open at a time. Open an existing logbook or create a new one using **File ⇨ Change logbook**.

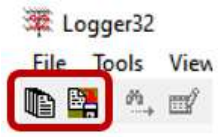


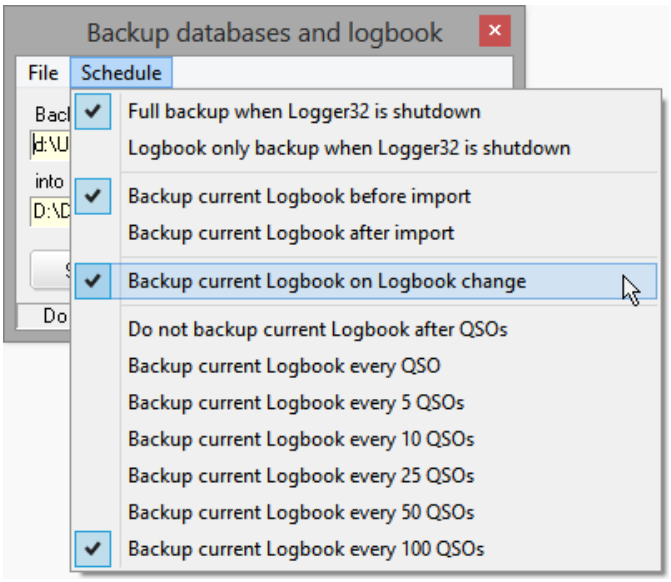
The **Change logbook** form shows the path (disk and folder), logbook name and the operator for the current (open) logbook ▶

If you move your station to a new QTH, click the <**Change Lat/Long**> button to adjust your latitude and longitude, updating the location of your station on the maps and adjusting the azimuth directions to DX stations.

The caption on the **Change logbook** form reminds you if backups are made automatically when you change logbooks (as recommended).

To change your automatic backup settings, click either of the backup icons (#1 or #2 on the main menu) to launch the backup function ▼ then click <**Schedule**> on the backup menu ▶





8.9.1 Select an existing logbook

There are two ways to select an existing logbook from **File ⇨ Change logbook**:

1. **Use the default logbook path**: simply select a logbook and operator using the drop-down lists on the form, then click <**Apply**>.

– or –

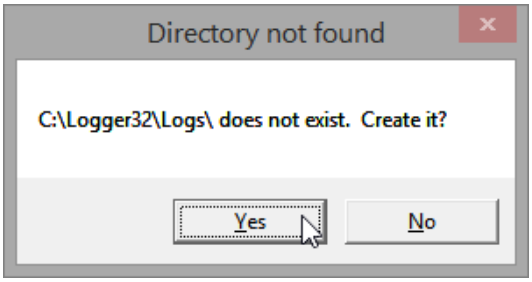
2. Click **<Browse …>** to open a dialog. The caption shows the path for the currently open logbook, with the folder and logbook name highlighted in the selection panels. Find and click the logbook you wish to open, then click <**Apply**> to close the current logbook and open the selected one instead.

8.9.2 Create a new logbook

1. Use **File ⇨ Change logbook** to open the form.
2. Accept the default path (disk and folder) or enter a new path in the path edit box (*e.g. C:\Logger32\Logs*).
3. Enter a distinctive new logbook name into the logbook edit box (*e.g.* K4CY 2026).
4. Select an operator or enter a new operator/station callsign into the operator edit box.
5. Click <**Apply**>.

Copyright © 2026 Robert C. Furzer K4CY

6. Wait a moment for Logger32 to make a backup of your current log before changing to the new one.
7. If you specified a new path, Logger32 confirms that you wanted to create a new folder (click <**Yes**> to proceed or <**No**> to go back) ▶
8. The new logbook opens. The logbook and statistics are empty.
9. If you wish, import an ADIF log file or just start using your shiny new log.



Hinson tip: multiple logbooks on the same computer *must* be uniquely identified with different file names or paths. Including the year and perhaps the month as part of the name makes it easier to tell the log files apart and identify the most recent (and usually current) one. If you maintain logs for multiple callsigns, include the relevant callsign/s in the log file names, and perhaps separate appropriately-named folders to keep things in order.

8.10 QSO numbering

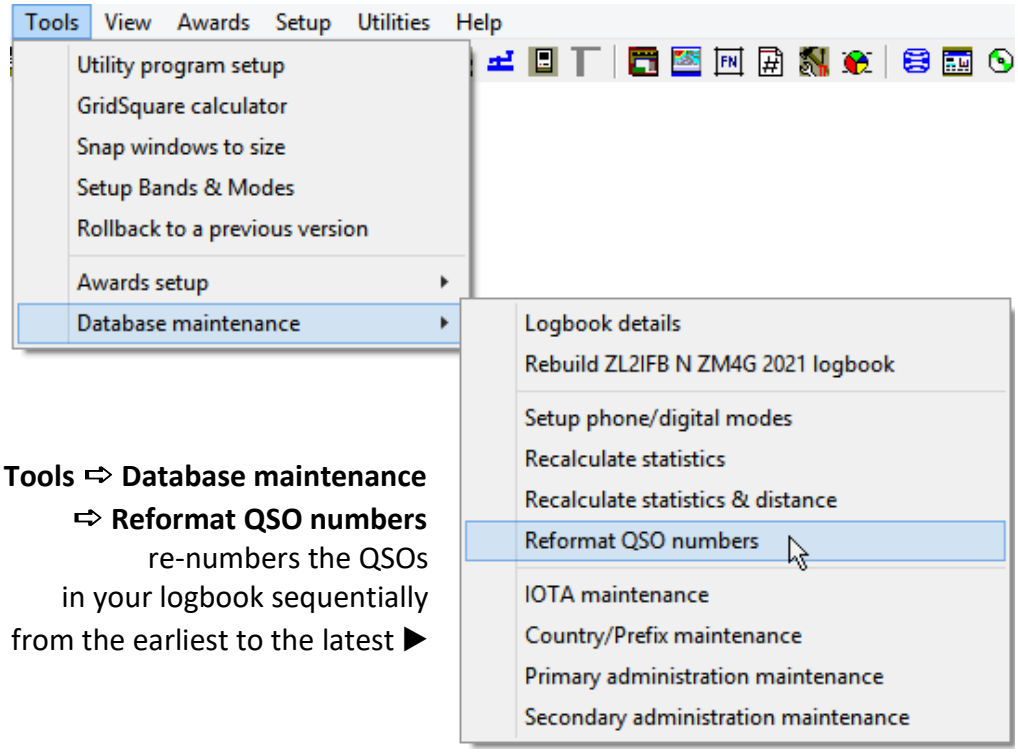
Are you one of those Logger32 users who is keen, perhaps obsessive about having all your QSOs neatly numbered according to the exact sequence in which they were made?

Do little discontinuities or anomalies in the QSO number sequence catch your beady eye and annoy you intensely?

Is this issue keeping you awake at night – as opposed to topband DXing?

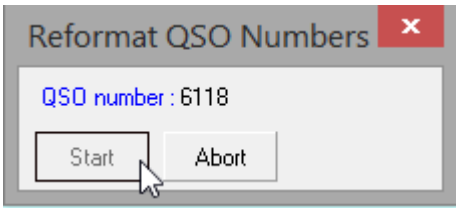
If so, this Logger32 function is for you!

If not, carry on as you were.



Tools ⇨ Database maintenance ⇨ Reformat QSO numbers re-numbers the QSOs in your logbook sequentially from the earliest to the latest ▶

Copyright © 2026 Robert C. Furzer K4CY



◀ If you have edited the dates or times of logged QSOs, or manually entered or imported any QSOs into your logbook with dates and/or times in the past, the QSO numbers may be sequence-out-of in the logbook. This function simply re-numbers all the logged QSOs strictly in date and time order, putting the universe neatly back in tidy, working order.

8.11 Comments and notes

There are two logbook columns to store additional information connected with a QSO: comments and notes.

There is a subtle difference in the way these two functions work and a considerable difference in the way information is displayed. In short, the "Comments" function is quick and easy to use, but not so easy to retrieve the information; "Notes" is a little more time consuming to use, but all notes associated with the present callsign are shown.

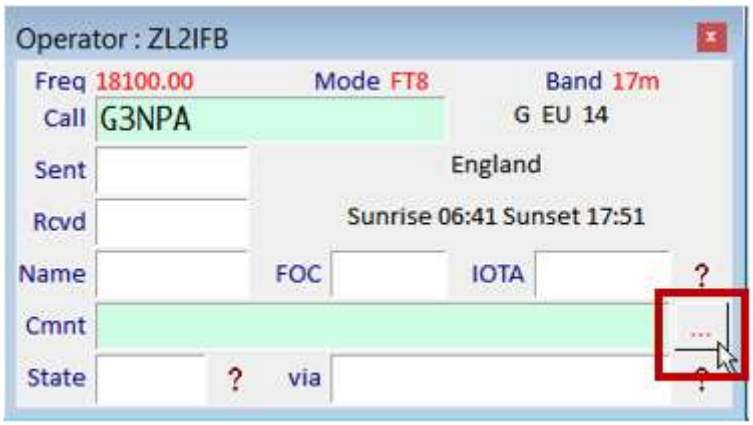
These are not exclusive options: a QSO may have *both* comments *and* notes attached. Or neither.

8.11.1 Comments

Comments can be saved in the logbook through the "Cmnt" field on the log entry pane, provided it is shown.

You can see the saved comments in your logbook if the comments column is selected in the logbook (see Grid configuration).

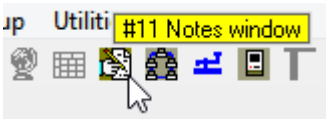
To conserve valuable screen space, clicking the three dots button [...] on the right side of the log entry pane cycles the field between Cmt (comment), QTH and Addr (address) ▶



Stop clicking the button with "Cmnt" shown, then type in a comment that will be stored in the comment field of the logbook against that QSO when it is logged.

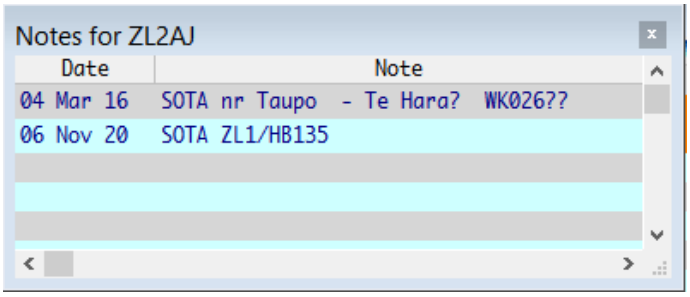
8.11.2 Notes

Open the Notes window[151] using icon #11 on the toolbar or **View ⇨ Notes window** ▶



[151] Prior to version 4, the Notes windows was a child window that was constrained to the main Logger32 window area. It has grown up in version 4 and can now roam free, anywhere on your displays. Drag it into position and size it as you wish. Being a petulant teen, however, it may not

Copyright © 2026 Robert C. Furzer K4CY



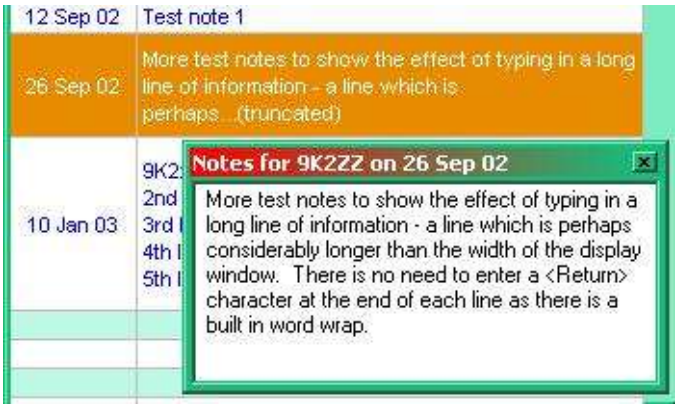
◄ If there are notes associated with the callsign presently in the log entry pane, the Notes summary window shows them.

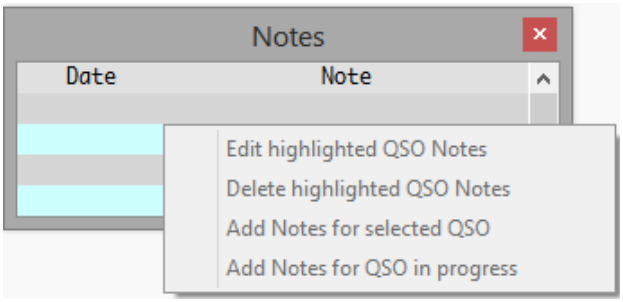
Notice the callsign in the caption.

If there is no callsign, or if the callsign does not have any notes associated with it, the Notes summary window is of course blank.

The Notes summary window can only display 100 characters: longer notes are cut short with "(truncated)".

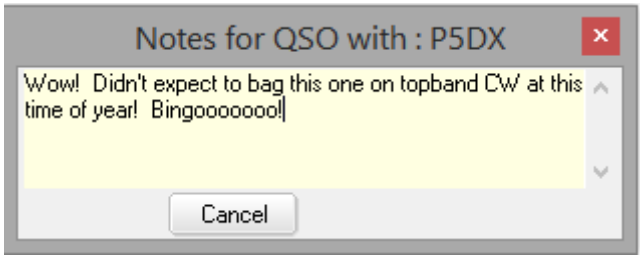
Clicking any of the notes shown will highlight that QSO in the logbook and "Notes for …" displays the complete note ►





◄ Right-click the notes window to bring up a little menu.

If you have selected a QSO in your logbook, or are in the process of logging a QSO, various options will be enabled/disabled, for instance when logging a QSO, click <**Add Notes for QSO in progress**> and type away in the little form that appears ►



When you are done typing, click the log entry pane to continue logging the QSO as normal.

You may also edit or delete a note from the Notes window using that little right-click menu. Click <**Edit highlighted QSO Notes**> to … yes, that's right … edit the notes for the selected QSO. Click <**Delete …**> to … well, I'm sure you can hazard a guess at what that is for!

8.12 Updating QSO flags

Logger32 uses 'flags' (binary bits) in the log to mark certain QSOs for export. After those QSOs have been exported, Logger32 offers to update the relevant flags and other fields.

---

stay put and occasional tantrums are likely. As it is listed under the <**View**> menu, you can reset it to the top left of the main Logger32 window using **View ⇨ Find lost windows**.

Copyright © 2026 Robert C. Furzer K4CY

Note: Only QSO records with the appropriate flag set will be exported. To see if a particular record is flagged, right-click the QSO in the logbook. You will see all three types listed: a tick against a given type means it is flagged for the corresponding export.

- **eQSL records:** if you choose yes to have these records flagged as having been sent, Logger32 will remove the export flag from all the records that were successfully exported.
- **QSL records:** if you choose yes to have these records flagged as having been sent, Logger32:
  - Removes the export flag from all the QSO records that were successfully exported.
  - Sets the QSL_Sent flags to Y for those QSOs.
  - Records the current date in the QSLSDATE fields for those QSOs.
- **LOTW records**: if you choose yes to have these records flagged as having been sent, Logger32 removes the export flag from all the records that were successfully exported.
- **CSV records:** the **File** ⇨ **Export files** ⇨ **Export QSL file** function gives you the option of generating ADIF or CSV file types.



QSO records in the CSV file ▲ have *two* QSO date fields:

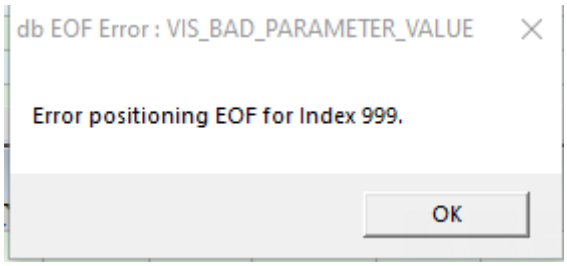
- **QSO_DATE** uses the unambiguous YYYYMMDD format which meets the ADIF spec and ISO 8601's basic date format but (unfortunately!) is uncommon on QSL cards.
- **APP_LOGGER32_QSO_DATE** uses the date format of your choice, making it easier to use that format on your QSL cards if you prefer.

8.13 Setup custom ADIF fields

You can setup custom ADIF fields when exporting QSL or LoTW files in the same manner as when exporting your log: click <**Setup custom ADIF fields**>. This is explained in the Exporting logs section.

8.14 Logbook FAQs

Q. After attempting to switch to a different logbook, I got an "Error positioning EOF for Index 999". Since then, I haven't been able to open my main log …



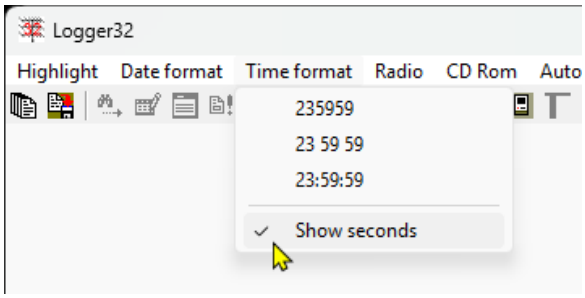
A. That cryptic Visual Basic error message tells you that, for some reason, Logger32 is trying to list your QSOs by column 999. Since there is no 'column 999' in the logbook, it fails, naturally.

The fix is simple. Click the <**OK**> button on the error message. Then click the mouse on your Logbook column header that shows QSO_DATE to sort and display the logbook chronologically, as normal. You may have given the column a different title *e.g.* "Date".

Copyright © 2026 Robert C. Furzer K4CY

Q. I'm only vaguely interested in what times my QSOs occurred. I honestly don't care about the seconds, so can I hide the superfluous detail?

A. Yes, simply click to *un*-tick "Show seconds" under **Setup ⇨ Time format** ▼



Seconds are optional in the ADIF standard ▼

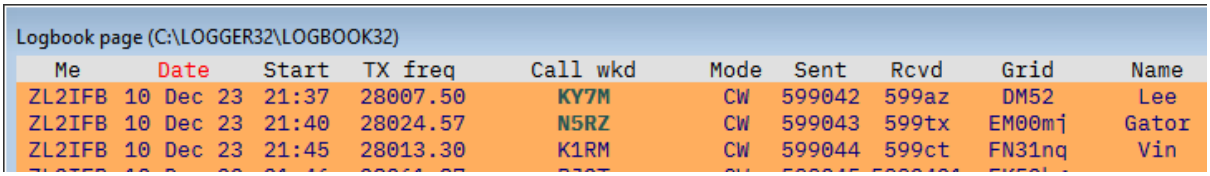
6 Digits representing a UTC time in HHMMSS format or 4 Digits representing a time in HHMM format, where

- HH is a 2-Digit hour specifier, where 0 <= HH <= 23 [use leading zeroes]
- MM is a 2-Digit minute specifier, where 0 <= MM <= 59 [use leading zeroes]
- SS is a 2-Digit second specifier, where 0 <= SS <= 59 [use leading zeroes]

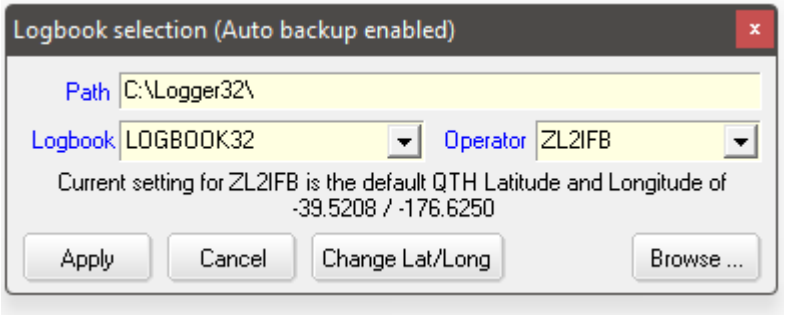
However, in Logger32's database architecture, QSO times are a key field that *must* be unique for each QSO record, so the seconds values are important to the system. Rabid contesters, pileuppers and FT2 ops may make more than one QSO per minute, and would be *aghast* if their quick-fire staccato QSOs were not all duly recorded in Logger32. Deselecting the **Setup ⇨ Time format ⇨ Show seconds** option simply hides the seconds values from you when *displaying* QSO times. Internally, the secret seconds remain, good as gold.

Q. What is the name of my logbook file?

A. Start by looking at the header line for your logbook pane, where the logbook path and logbook name are shown. Here's mine ▼

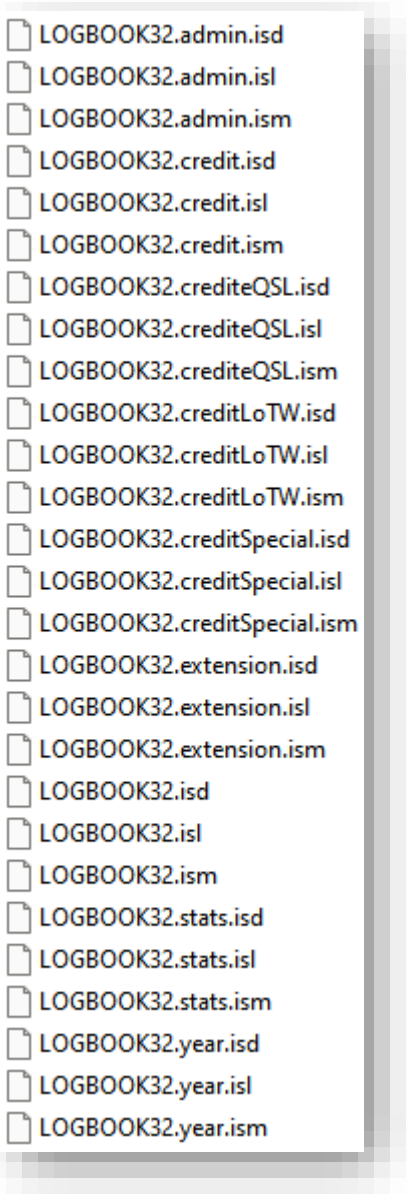


File ⇨ Change logbook also shows the path and logbook name, like this ▼



Copyright © 2026 Robert C. Furzer K4CY

Each logbook is stored on disk as a set, a family of database files - *roughly* 27 of them (some are created on demand) - with the logbook name as the first part of the file names and then one or two file extension parts, like this ▶



Hinson tip: if you delete or corrupt any of these *.isd*, *.isl* or *.ism* files, you will almost certainly mess up the corresponding logbook (the one called "LOGBOOK32" in this example), making it **unusable** and **irrecoverable**. That's fine *if* you actually *wanted* to delete your logbook. If not, **leave the administration of these files entirely to Logger32 *and* be sure to keep backups of your precious log!**

Q. 'Something' happened, and now when I launch Logger32, I get a worrying message about "Timestamp error" and my logbook comes up blank. Oh bother. Have I *really* blown it this time?

A. In Logger32, 'a logbook' is stored in a family of database files on disk. So that the database files and hence the log data remain synchronized and complete, the files each have timestamps that are cross-checked automatically when a logbook is opened. Normally, all the database files for a given logbook have identical timestamps and the world is a happy place. If not (*e.g.* if any of the files were corrupted or truncated, infected with a virus, renamed or deleted) Logger32 attempts to find and fix *minor* issues but, if that fails, it presents the error message and gives up in disgust, thereby preventing any further damage to your precious log.

Since the database is toast, you need to recreate it by creating a new log and importing the QSOs from your most recent pre-incident log backup.

If you have no steenkin' backups, you have indeed *really* blown it this time. You might try persuading Bob to attempt to reconstruct your log from the broken database files but I wouldn't hold out much hope of success. If you bribe him with one-too-many tots of Famous Grouse, the results are sadly predictable.

Q. Why is the background red?

A. The background to what? Give us a clue …

Q. Why do most if not all QSO lines in my logbook have a red background?

A. Because that's the way it is configured …

Q. But I don't like the red. How do I change it?

A. You reconfigure Logger32 …

Copyright © 2026 Robert C. Furzer K4CY

Q. Oh, *come on*! *How* do I reconfigure it?

A. Using Logger32's extensive color configuration options, obviously …

Q. Hey, quit winding me up! *What* configuration options control the background color of the QSO lines in the logbook?

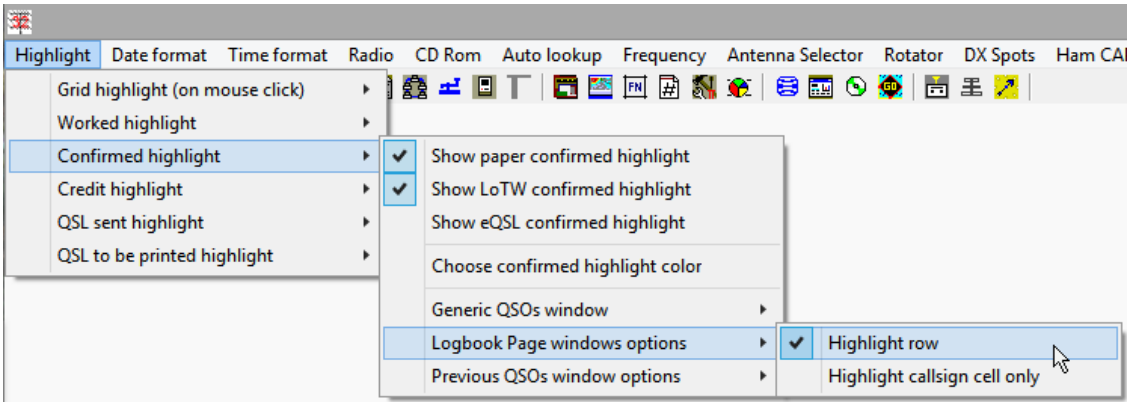
A. Aha, *finally* we're getting somewhere! You're seeing the red mist but at least now we know enough about your issue to offer a sensible solution. Hint: it would have been quicker if you'd started from here!

OK, so, think of the logbook colors as successive coats of opaque paint:

- The primer is determined by **View ⇨ Grid appearance ⇨ Grids background color**.
- The optional undercoat is a special paint that comes in fetching stripes. Check **View ⇨ Grid appearance ⇨ Grids background color (alternate rows).** Use this stripy paint to make it just a bit easier to read along each QSO row without your eye straying into the adjacent ones.

> **Hinson tip**: chose subtle but contrasting shades for the primer and undercoat.

- The next coat is even more special: it only applies a colored accent to particular *types* of QSO. Check **Setup ⇨ Highlight** for yet more options ▼



- Through the applicable **Setup ⇨ Highlight** submenus, choose which *types* of QSO to highlight (*e.g.* QSOs that have been confirmed, or QSOs that you plan to confirm on QSL cards …) and if so what colors to highlight them.

> **Hinson tip:** although the highlighting can be different, the logbook, previous QSOs and generic QSO 'grids' all share the same basic text settings (font, size and color) to retain some semblance of normality … and operator sanity.

- Notice the accent color can be applied either to the whole row, or only to the callsign field, for the relevant QSOs. Presumably when this was designed, Bob was contemplating color-coding individual logbook columns differently, giving either a tartan paint or LSD-inspired kaleidoscopic effect.

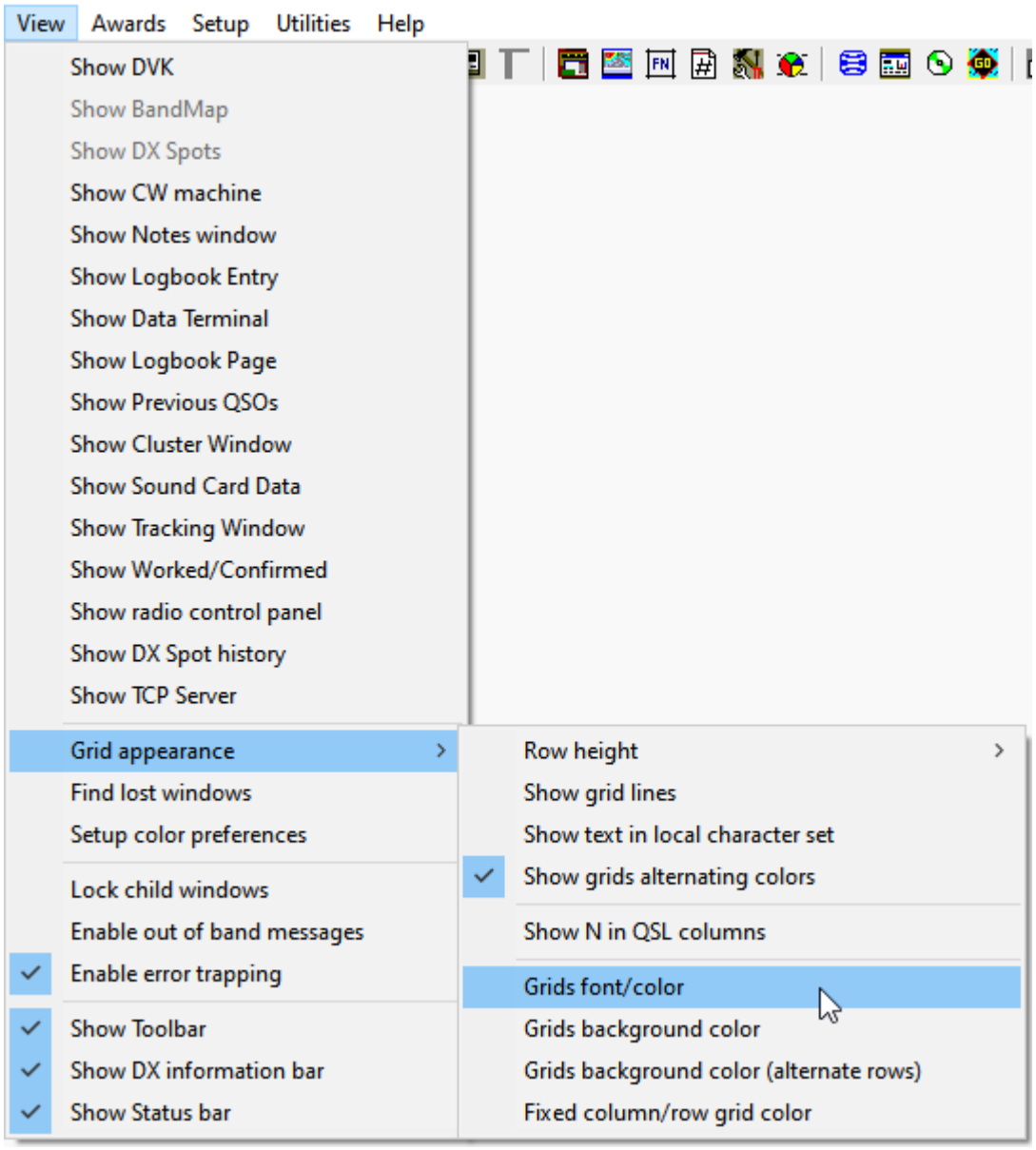
So, getting back to your question, the combination of some 40 configuration menu items is responsible for red backgrounds.

Good luck with those color charts and test pots.

Copyright © 2026 Robert C. Furzer K4CY

Q. Thanks! Now, what about the QSO text color and size?

A. Tucked away on the **View** ⇨ **Grid appearance** submenu are a few handy options including fonts and colors ▼



These settings affect the QSO text on the logbook, previous QSOs *and* generic QSO panes. Look here for more.

Q. I just loaded my log and I have *NOTHING* in the logbook Prefix column. Why?

A. The ADIF file you loaded presumably didn't contain the PFX field and you neglected to get Logger32 to determine the prefixes for you by selecting <**Ignore PFX field**> during the file import. That option tells Logger32 to disregard the PFX field (if present) in the ADIF file and instead to determine and store the *correct* PFX values in accordance with the ADIF standard.

You can fix this quite easily: export your entire log as an ADIF file (creating a valuable backup!), delete your 4 logbook files, and import the ADIF file you just created, this time with the <**Ignore PFX field**> option selected. See chapter 3 for more information.

Copyright © 2026 Robert C. Furzer K4CY

Q. I receive a Runtime Error 13 - type mismatch: errrr …

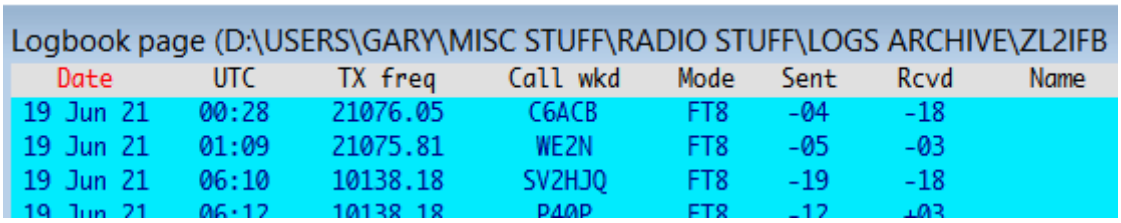
This error is usually associated with an error in the logbook data, so:

1. Backup your original logbook by saving a copy of four files named *C:\Logger32\[mylog].isd, isf, .isl* and *.ism* where *[mylog]* is the name you have given your log.
2. Start Logger32.
3. Export your entire logbook in *.ADI* (ADIF) format. Save it somewhere convenient.
4. Close Logger32.
5. Delete the four logbook files mentioned above.
6. Start Logger32 and import the *.ADI* file saved in step 3.

Q. When I sorted the logbook, my QSOs all disappeared. *C R I S I S !* Where have they gone? If Logger32 has lost my log, I'll be very, *very* upset …

A. When clicking the header cell of an individual column to sort your logbook by that column, bear in mind that **only QSOs with data values in that column will be shown**. If you sort by a column that has no data, you will see a blank logbook. Don't panic! The QSO information is still all there, just not displayed at this point.

To see your QSOs again, simply click another logbook column header to re-sort the logbook by that column ▼



Notice one column's header text is **red** indicating which the column is currently used to sort the logbook display.

Hinson tip: conventional logbooks show QSOs in chronological order … so **click the QSO_DATE[152] column header to turn it red**. Every QSO *must* have a date (it is a key field), so the entire logbook is sorted with all your QSOs shown in date and time sequence.

If there *still* appears to be QSOs missing, even with the logbook sorted by QSO_DATE, the missing QSOs may be:

- Completely or wholly hidden due to annoying bugs in the logbook display height adjuster code (there are workarounds in an FAQ at the end of the log entry pane chapter).
- Logged in a different logbook, not the one you currently have open.
- Lost, corrupted or deleted by a puff of logic or zap of static … which is why it is so important to make regular log backups.

[152] *Your* logbook column names probably differ from mine. While the underlying ADIF field names are determined by the ADIF standard and fixed, the column headings on the logbook display can be changed, for example using lower case instead of SHOUTY CAPITALS, and abbreviations such as "freq" instead of "FREQUENCY". See here for instructions.

Copyright © 2026 Robert C. Furzer K4CY

Q. After importing an ADIF file, Logger32 added a pair of asterisks to some QSOs in my log, like **P5DX. Why?

A. Some programs export ADIF files with the DXCC field for some QSOs set to 000 (zero) to indicate 'Not accepted for DXCC' (*e.g.* for incomplete QSOs or pirates). When Logger32 encounters such QSO records, it adds them to the logbook with DXCC entity code 000 and prepends two asterisks to the front of the callsign in the logbook as a visual cue in your logbook that this QSO has *not* been counted towards your DXCC award statistics.

Q. I don't like the colors in my logbook. Can I change them?

A. Yes. In the main menu, click **Setup** ⇨ **Highlight color** and follow the prompts.

Q. Why can't I edit the DXCC field in the logbook? I just get an error.

A. You didn't read the whole error message that popped up, did you? You can't simply type in a different DXCC entity because it is a key field used for database lookups. Change it by right-clicking the cell and selecting a DXCC entity from the predefined list. It *is* possible to maintain DXCC entities (*e.g.* add, change or remove countries or country names) if that's what you want to do, but not simply by editing the logbook field.

Q. How do I change the width of the columns in my logbook, previous QSOs *etc*.?

A. Move your mouse carefully over the line directly between two column headers until the cursor changes to a double-headed arrow. Click and hold the mouse button, then drag the cursor left or right and release the mouse button when the column reaches your desired width.

Q. How come text in the logbook and previous QSOs window is missing the lower portion (tail) of some letters (*e.g.* y looks like v, q and g look the same)?

A. The font you are using is too large for the row height you have chosen. Either choose a smaller font or increase the row height in your grids.

Q. What is a twip?

A. For a while, I thought twip was a typo for thrip ▶ ...



… but no, Wikipedia tells me a twip is one 20th of a point – as if that makes any sense. A twip is one 1440th of an inch (even tinier than a thou), if that helps at all.

In "metric" SI units, there are ~57 twips per mm.

This is surely a contender for the *World's Most Obscure Unit of Measure* award, if there is such a thing – maybe a Guinness record. Visual Basic works in twips. I don't.

Copyright © 2026 Robert C. Furzer K4CY

Q. I don't know how or when it happened but *somehow* the grid squares in my log are seriously messed up. Do I have to fix every one manually?

A. No – or rather, probably not. It depends on:

- How important it is to you to have the correct grids for *all* QSOs in your log.
- Whether you need/want/desire to have legitimate grids for the QSOs in your log.
- Whether you are happy to only have grids recorded against logged QSOs where there is a reasonable basis for them (for various values of 'reasonableness').
- Whether you want to eliminate patently "impossible" grids *e.g.* grids that do not correspond to the DXCC entity indicated by the logged station's callsign, or landlocked grids for maritime mobile stations.

Assuming you had the good sense to make pre-junk backups, congratulations: you 'simply' need to recover your pre-junk log from the backups, then bring it up to date with any subsequent QSOs – hopefully not too many if you have plenty of backups and noticed the problem quickly.

If you don't have pre-junk backups/archives, and if there are *hundreds* of *different* junk locators in your log, you may have to delete them all and then figure out how to recover the ones that matter (*e.g.* do a complete full-log LoTW sync).

If there are hundreds of QSOs with just a *few* specific junk locators, I would simply delete all occurrences of those few junk values.

Either way, although you *could* delete/update the grids manually one QSO at a time through Logger32's log editing functions, that would be laborious and slow. A more efficient method is to export your messed-up log as an ADIF file, then use either a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad, or better still an ADIF editor such as ADIFmaster, to make bulk changes on *a copy of* the exported ADIF log[153].

You should also try to figure out how the junk grids got into your log in the first place, and deal with that problem to avoid adding yet more junk grids in future, even after you correct your log. As a clue, Logger32 obtains the grid squares from:

- FT# CQ messages that include the 4-character grid.
- LoTW (updated from a file generated by LoTW, or through Club Log), or QSL card confirmations (used to update Logger32 manually), that specify and confirm the stated grids.
- Online callsign lookups from records that specify grids (if that field has been configured to be transferred to the log).
- Data entry by you, when logging or editing a QSO record in your log.
- Importing an ADIF QSO record with a valid ADIF <GRIDSQUARE> field (valid meaning correctly formatted, not necessarily the right QTH!). If there are lots of grid errors, this is the most likely cause: did you perchance pre-process your log to look up and add additional information? Did that process increase rather than eliminate the junk?

[153] Work on a *copy,* keeping the *original* ADIF log export safe as a pre-editing backup just in case things go from bad to worse during the editing. Thanks to the School of Hard Knocks for this tip.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: this situation is *yet another* good reason to make, say, monthly, quarterly or annual archive copies of your log, and store them safely offline – preferably multiple archive copies, stored separately and verified, if you're as paranoid as me about losing or corrupting your log. With sound backups or archives, the worst-case scenario is that that you *might* be unable to recover QSOs made since the last backup/archive … but all previous QSOs up to that point should be restorable. For bonus marks, try restoring your log to a separate test system.

Q. I've imported a log but the logbook doesn't show any of the QSOs. Woe is me! Where did they go? Don't tell me they are lost!

A. The number one cause of this problem is log sorting. The logbook can be sorted by clicking the header labels for most columns. The sort also applies a filter *i.e.* **if the column you have sorted on contains no data at all, no QSOs are shown so the logbook *appears* completely empty**. Simply click a different column header label (such as the QSO number or date– these always contain data) and *all* your QSOs should magically reappear.

Hinson tip: after sorting your log by some other field for some reason, *try* to remember to click the QSO date column header to sort the logbook back into the default order, with the oldest QSOs at the top and the newest down at the bottom – just like a paper logbook.[154]

Q. How can I log my transmit *and* receive frequency when using split?

A. Provided Logger32 knows which are your TX and RX VFOs and which radio/s you are using, by default it logs them *both* in the "FREQ" (*i.e.* your TX frequency) and "FREQ_RX" ADIF fields, respectively.

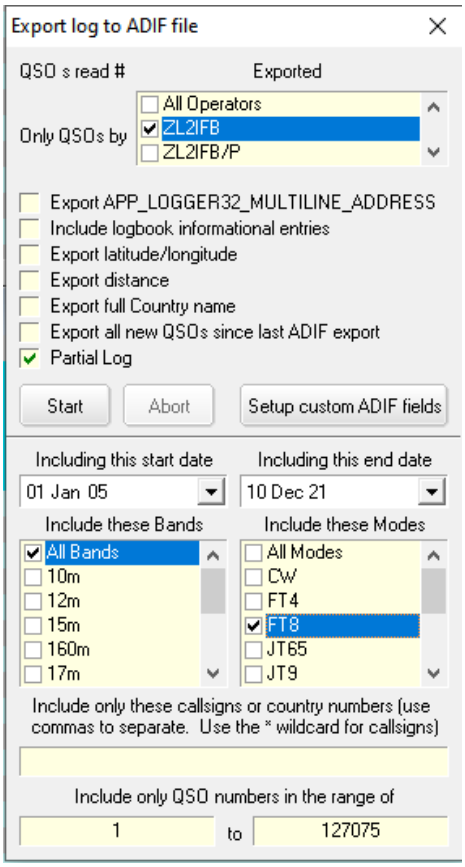
Q. How many QSOs have I made on (say) FT8?

A. Although unfortunately Logger32 doesn't (yet!) offer a report showing your QSO totals by mode, band *etc.*, there are third party utilities and services that can answer your question by analyzing your log having exported it as an ADIF file … which suggests a workaround: when doing a partial log export, you can select one or more bands and modes.

So, pick a selection or combination that intrigues you ▶ (*e.g.* just FT8, or maybe FT8, FT4 *and* FT2), run the partial log export, and scribble down the total number of QSOs exported ▼



Then repeat for whatever other modes, bands or operator callsigns you are keen to count.

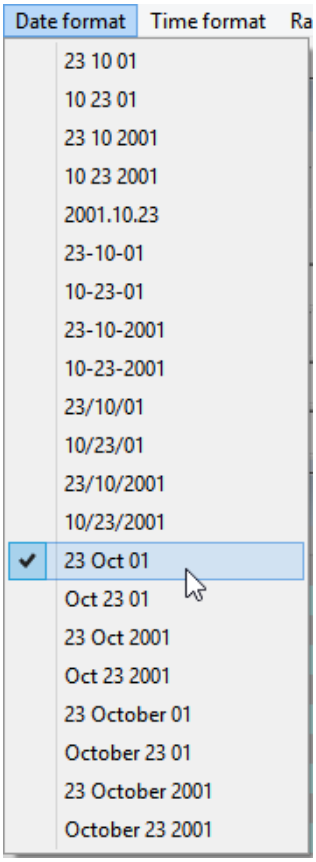


[154] Yes, this is a duplicate Hinson tip because it is perhaps the most frequent of FAQs.

Copyright © 2026 Robert C. Furzer K4CY

Q. Can I use MM/DD/YYYY for dates?

A. Yes.



Open **Setup ⇨ Date format** and click to tick your preferred date format from the long list available ▶

Regardless of the date format you choose, Logger32 will *only* use the ADIF-specified UTC date format when exporting ADIF files. Whatever you may think, anything else would not comply with the standard, and could be misinterpreted by other logging programs and hams.

Hinson tip: if you hand-scrawl your QSL cards, please be aware that the recipients – especially DX/overseas – may not use the same date format as you. Numeric formats are ambiguous in the day and month fields for the first 12 days of any month [155]. Is, say, 11/10/2025 the 10th of November or the 11th of October? Arguably, spelling out the month ("Oct") or using Roman numerals for the month is a little clearer, less likely to be returned "**N**ot **I**n **L**og".

Oh and stick with four-digit years if the Y2k debacle means anything to you. 11/10/25 is an even *more* ambiguous date: was that QSO in 2025 or 1925? Which of those three fields is the day?

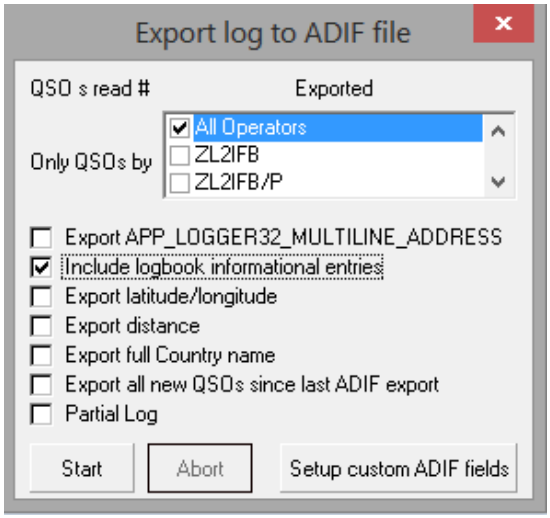
You can customize the display format for times in a similar way using **Setup ⇨ Time format**.

And one last thing: Logger32 works internally in UTC - not local time, summer time, GMT, astronomical time, lunchtime, play time, pastime, about time, out of time, closing time or any other kind of time. Again, that is as specified in the ADIF standard for global compatibility. Logger32 can cope if your PC clock shows your local shack time *provided* Windows itself is using UTC with the correct time zone (take another look at Computer clock setup).

Q. Can I rename my log files?

A. Yes … but not directly:

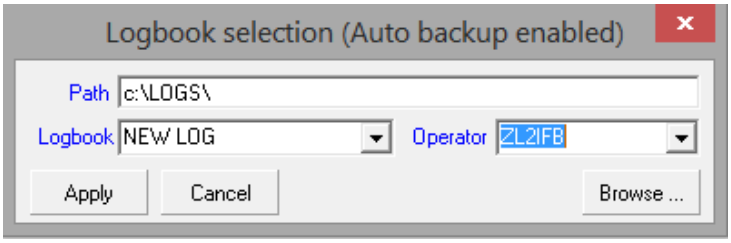
1. Export your complete log as an ADIF file using **File ⇨ Export Logs ⇨ ADIF (.adi) file**. Select <**All Operators**> to export all QSOs from your open log, and <**Include logbook informational entries**> to retain any pseudo-QSOs that you have logged in that way ▶
2. Prepare to open a new logbook using **File ⇨ Change Logbook**.



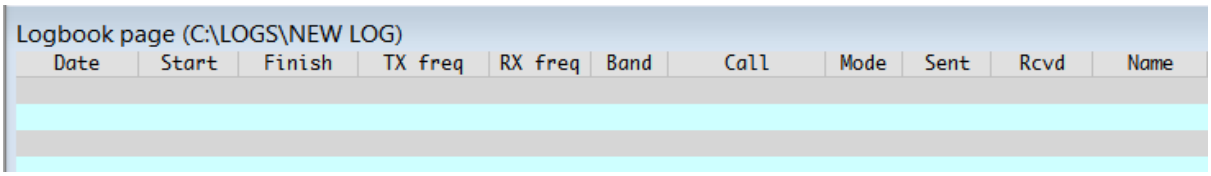
[155] ISO 8601 specifies YYYY-MM-DD or YYYYMMDD, two international standard date formats that aren't [yet] available in Logger32. YYYY.MM.DD (*e.g.* 2001.10.23) is as close it gets, using periods (full-stops) instead of hyphens (dashes). Logger32 doesn't [currently] support Roman numerals either. Too bad.

Copyright © 2026 Robert C. Furzer K4CY

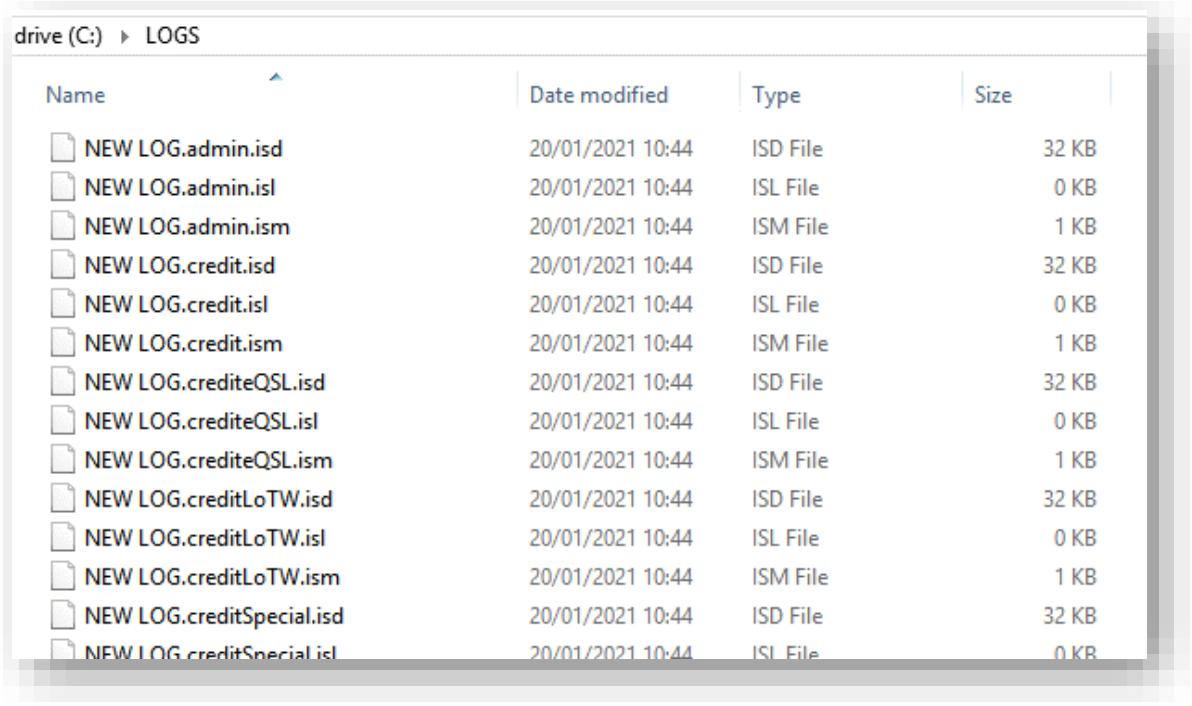
3. Either type in the path name or click the <**Browse**> button to find a suitable folder ▶



4. Give your new logbook a distinctive name to be shown in the caption for the logbook pane (which at this point is empty: *don't panic!*) … ▼



… and will be used for ~27 new logbook database files ▼



5. Click <**Apply**> and follow the prompts. Logger32 checks whether the folder exists already, and if not, it asks whether you want to create it. Presumably you do.
6. Now import the ADIF file you created into your new logbook, using **File ⇨ Import Logs ⇨ ADIF (.adi) file.** You *may* need to check and fix any import errors, hopefully none.
7. Check out the log.

Hinson tip: you *may* need to recalculate your statistics but probably not, since the import routine automatically checks/creates/updates the statistics as each QSO is added to the log.

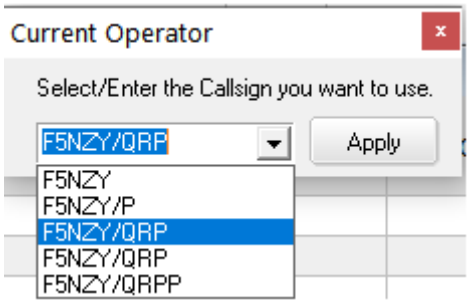
Copyright © 2026 Robert C. Furzer K4CY

Q. How come when I log a QSO, it doesn't appear in the logbook … unless I log it again, in which case *both* QSO lines appear. This is *so* frustrating!

A. See this FAQ in the Log entry pane chapter.

Q. Why does my callsign appear twice in the *Current Operator* list?

A. Eagle-eyed Steph noticed a duplicate of the callsign he uses when QRP ▶

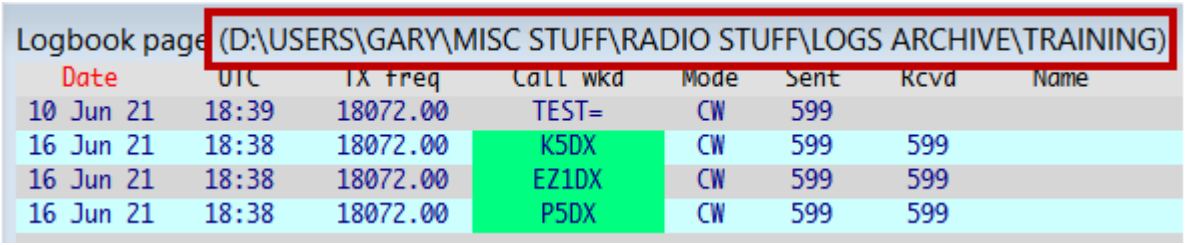


The reason was that this Current Operator list is compiled by Logger32 from the Operator field in the open log … and Steph had accidentally defined and logged his operator callsign for at least one QRP QSO as "F5NZY/QRP<space>". He sorted the log by Operator, found the offending QSO/s, removed the errant space, and the world tipped neatly back onto its proper axis with a faint click.

Q. Can I check something out without the risk of damaging my log?

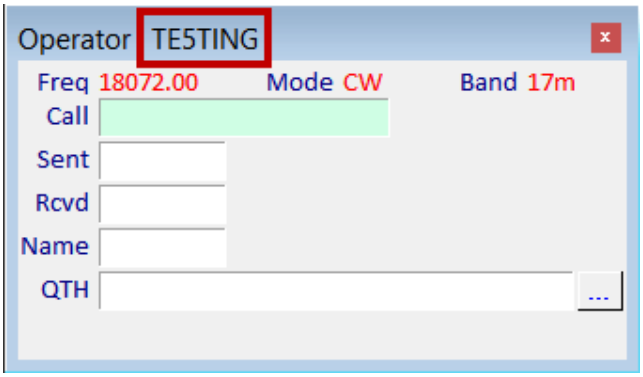
A. Logger32 beta testers use test logbooks to explore new functions and hunt down bugs in changed code. It is easy to generate a new, blank log: click **File** ⇨ **Change logbook** then come up with an appropriate name (such as TESTING, TRAINING …), preferably in a folder dedicated to testing in order to separate the test files from your main log (*e.g. C:\Logger32\TESTING*).

The log file name and path are shown on the caption for the logbook ▼



Hinson tip: for testing or training purposes *etc*., choose a fake "operator" callsign that is clearly *not* any of your normal callsigns. TE5TING or F00L work for me. You may prefer TR4INING, N0TMYCALL, TR1AL or something more creative, more distinctive, more fun.

The caption of the log entry pane reminds you that you are not using your real log ▶



Whereas a new logbook starts out empty, it is generally useful to have some QSOs in the log for testing purposes. An obvious approach is to import an ADIF file that you have exported from your working log - a *copy* of your real logbook for testing and training purposes. Then, if disaster strikes during the process, nothing too important is lost. Your original log remains just a couple of mouse clicks away, simply by changing back to your real logbook.

Copyright © 2026 Robert C. Furzer K4CY

The test/training log can be used to check out unfamiliar Logger32 functions, perhaps logging QSOs with fake callsigns on unusual frequencies/bands/modes, messing around with the QSL flags, changing award selections, logging informational logbook entries *etc*. If you then export an ADIF from the test system (preferably with a file name that clearly indicates it is a TESTING or FAKE log), you can re-use those QSOs in future testing, gradually building up a library of test cases just like professional software testers.

Aside from testing, a fake log is useful to demonstrate Logger32 to other hams, for training/master classes or to make instructional videos for other Logger32 users on YouTube (hint hint!).

Hinson tip: look at Logger32's configurations as well. For training purposes, it is worth setting up a basic configuration, close to the defaults and without all the clutter and complexity that an experienced Logger32 user tends to accumulate. For testing, it may help to have something closer to your normal setup, maybe deliberately choosing different screen layouts, colors and alerts to make it *obvious* that, while testing, you are not using your real log.

When no longer required, your *test* logs can simply be deleted. If you have done as we recommended, they will be distinct from your *real* log/s in *C:\Logger32\TESTING*.

Hinson top tip: pay attention, especially if you copy a real log for testing purposes. It is easy to get confused between the real and test logs. Don't start using a test log as your real log and of course be very careful *not* to mess up or delete your real log/s by mistake.

Special bonus Hinson tip: if it's too late, retrieve your real log/s from backups.

Make frequent backups if you know what you're doing.
Make them even more frequently if you don't.
Bob K4CY

***Extra*-special free bonus Hinson tip**: if you have no backups either, you're stuffed. Don't say you weren't warned. Don't you come running to me with your broken leg.

Copyright © 2026 Robert C. Furzer K4CY

Section B

Logger32 intermediate topics

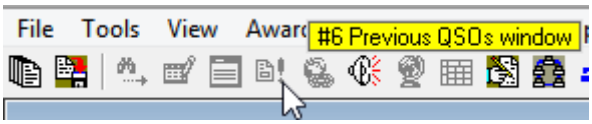
Copyright © 2026 Robert C. Furzer K4CY

9 Previous QSOs pane

> "Learn from yesterday,
> live for today,
> hope for tomorrow"
>
> *Albert Einstein*

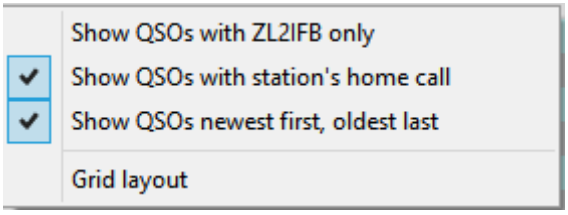
When you enter a full callsign in the log entry pane, or click a QSO in the logbook, Logger32 looks up any previous QSOs you've logged with the station, showing them in the previous QSOs pane.

Logger32 can display up to 1,000 matching entries from your logbook[156]. Open the previous QSOs pane from <**View**> or by clicking toolbar exclamation mark icon #6 ▶

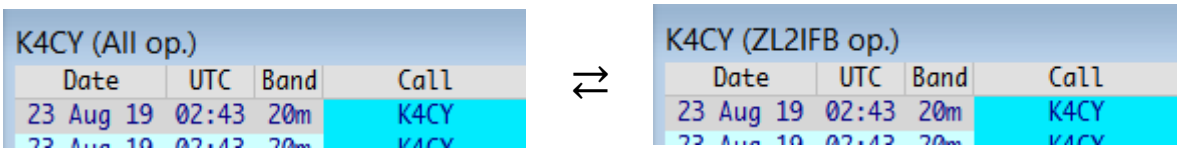


9.1 Configuring the previous QSOs pane

Right-click click any blank line in the previous QSOs pane to open the configuration menu ▶



- **Show QSOs with** [current operator's callsign] **only**: Logger32 can either show QSOs logged by any operator in the open logbook, or just those logged by the current operator. The caption reminds you which option you have selected ▼



- **Show QSOs with station's home call**: Logger32 knows that K4CY, K4CY/M and K4CY/P are all the same person – plus 5H/K4CY and K4CY/VE2 and other compound callsigns containing K4CY. With this option selected, the previous QSOs pane shows *all* your logged QSOs with the callsign in the Call field of the log entry pane and variants, both home and away.
- **Show QSOs newest first, oldest last**: by default, the pane displays previous QSOs from the top down. Some QSOs may therefore be out of sight, off the bottom of the visible pane. With this option selected, the most recent QSOs and probably most interesting ones are shown at the top, while older ones may be out of sight. Regardless of the setting, you can reverse the sort order dynamically by clicking the date or time header. Click it again to revert to the original sort order.

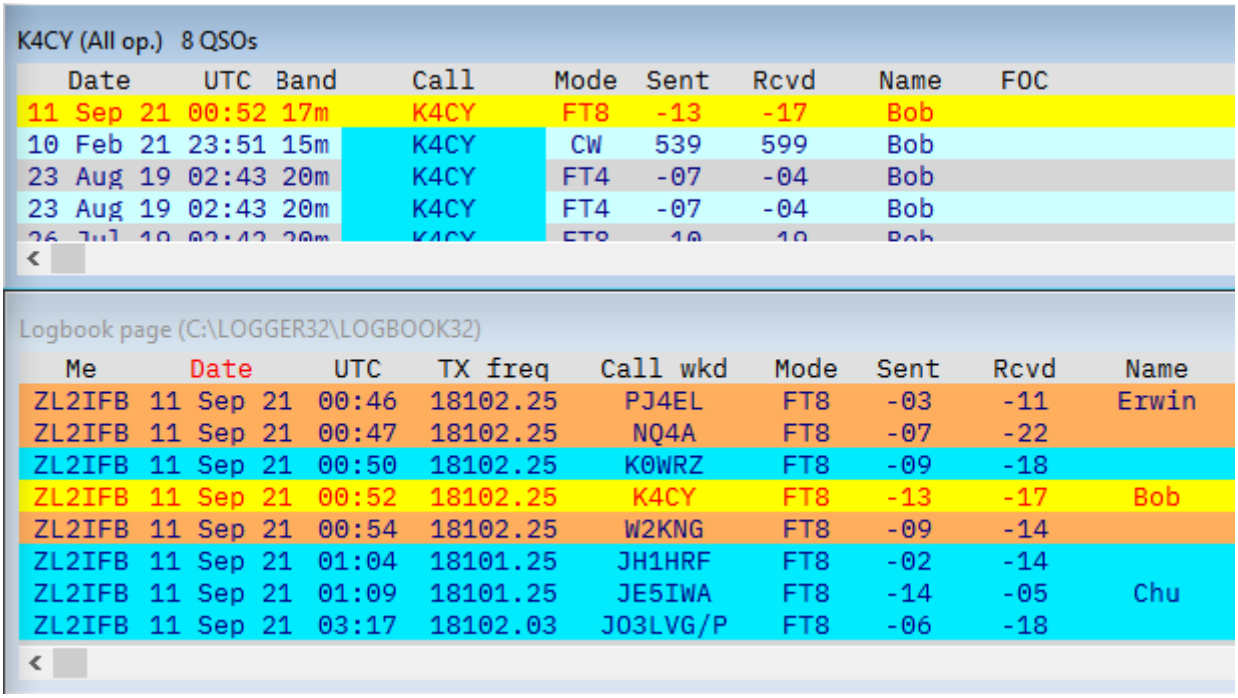
[156] If there is insufficient vertical screen space to show all the QSOs in the previous QSOs pane, the total number of QSOs is shown in the caption.

Copyright © 2026 Robert C. Furzer K4CY

- **Grid layout**: use this to choose which columns to show, change the column headings, left, center or right-justify the columns, and change their order. See the grid layout configuration section for details.

9.2 Selecting a QSO

Clicking any QSO in the previous QSOs pane highlights the QSO line *and* looks up and highlights the same QSO in the logbook, showing it in the context of contemporaneous QSOs ▼

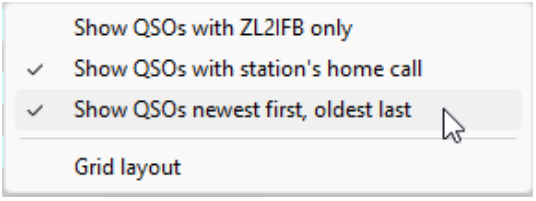


9.3 Changing the appearance of the previous QSOs table

You can change the appearance of the QSO grid such as displaying the grid lines, changing the row height and column width, alternating the row background colors and much more, using **View ➪ Grid appearance** *e.g.* <**Grid font (highlight)**> sets the color of the text when you highlight (select) a row, while <**Grid background (highlight)**> changes the background color for that selected row.

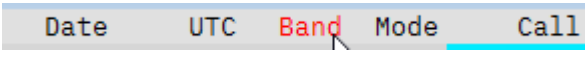
9.4 Grouping and sorting previous QSOs

By default, QSOs are displayed in the previous QSOs pane in date and time sequence, with the most recent QSOs at the top down to the oldest at the bottom. Click the date or time header to re-sort the displayed QSOs in the reverse order. Right-click an empty row in the previous QSOs pane for the option to reverse the default chronological order ▶



Likewise, click the band or mode column headers to group the displayed QSOs together by bands or modes. Within each band or mode group, previous QSOs are sorted chronologically in the default order.

As with the logbook, the header of a sorted column turns red ▶



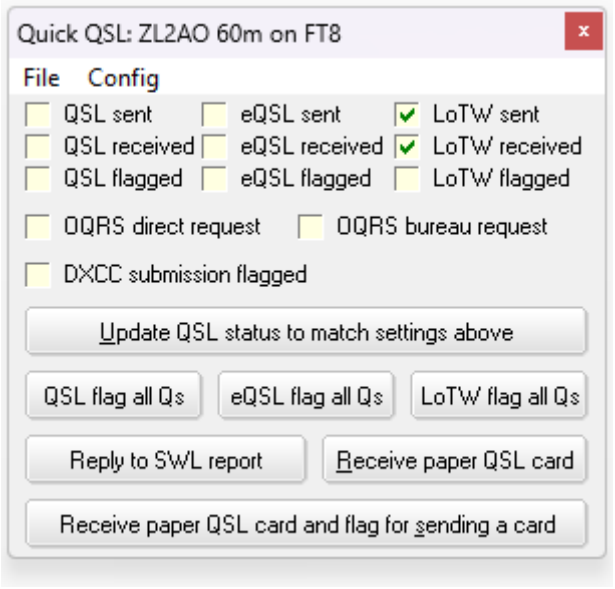
Copyright © 2026 Robert C. Furzer K4CY

Successive clicks on the same column header reverse the group sequence.

Hinson tip: this function is easier to use than to explain! Try it for yourself. In your logbook, click the line for someone you have contacted several times, and study the previous QSOs pane. Click the band or mode headers to show blocks of QSOs on each band or mode respectively.

9.5 Check/update QSLs

Right-clicking a QSO in the Previous QSOs pane selects the QSO on the log page, then opens the <**Quick QSL**> form to check or update the QSL information for that QSO ▶



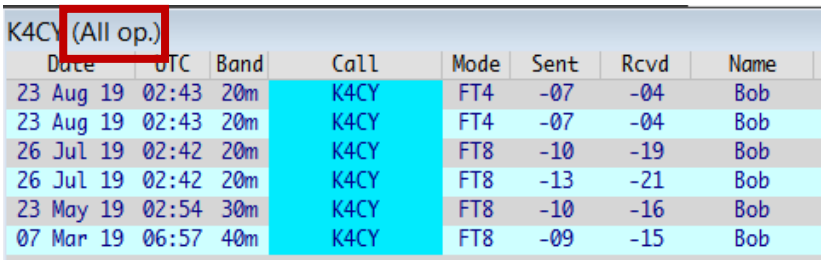
You'll find more about this in the QSLing chapter.

9.6 Previous QSOs pane FAQs

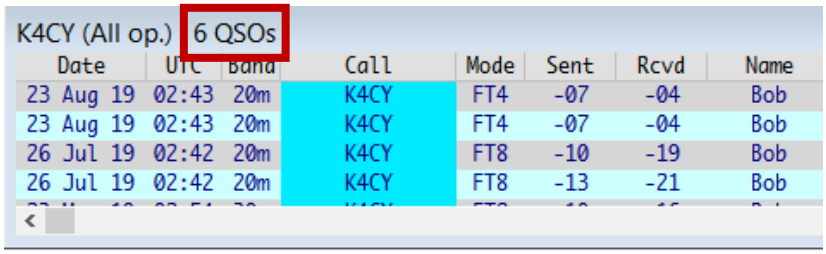
Q. How many times have I worked (say) K4CY?

A. Put the callsign in the log entry pane and the previous QSOs pane shows your previous QSOs with that callsign. That's a good start.

If several operators have made K4CY QSOs in your open log (*e.g.* you and a significant other who shares the shack), "(All op.)" in the window caption reminds you that it is showing all the QSOs it can find in the log, regardless of which operator actually made them ▲



If you have made more QSOs than can be shown given the height of the previous QSOs pane when it is populated, the caption also tells you the total number of QSOs because you can't quickly count them on the screen by eye without scrolling ▲



Copyright © 2026 Robert C. Furzer K4CY

The QSO count isn't given in the caption if the pane is tall enough to show all of them.

Finally, if you had worked K4CY while he was out mobile, portable, in a ship, in a plane or on a jolly, so long as he was using his home callsign (even with location modifiers such as ZS/K4CY/M), it will show those QSOs too and include them in the count. Clever stuff!

Serial DXpeditioner and world traveler Nigel G3TXF (now OE3TXF) is a genuine example from my log … and maybe yours too!

46 fabulous DX QSOs so far ▶

G3TXF (All op.) 46 QSOs

Me	Date	UTC	Band	Mode	Call	Sent	Rcvd
ZM4G	12 Sep 20	18:22	40m	CW	G3TXF	559	559
ZL2IFB	03 Feb 19	17:42	40m	CW	G3TXF	599	i99126(
ZL2IFB	02 Feb 19	07:56	80m	CW	G3TXF	599	i99126(
ZL2IFB	30 Jan 19	17:37	30m	FT8	G3TXF	-09	-12
ZL2IFB	11 Jul 18	07:27	30m	FT8	G3TXF	-05	-24
ZL2IFB	15 Mar 17	04:41	17m	CW	3B8/G3TXF	559	599
ZL2IFB	14 Mar 17	03:34	30m	CW	3B8/G3TXF	229	599
ZM4G	12 Mar 17	05:36	15m	CW	3B8/G3TXF	599	599
ZM4G	12 Mar 17	03:20	20m	CW	3B8/G3TXF	599	599
ZM4G	11 Mar 17	14:38	40m	CW	3B8/G3TXF	599	599
ZM4G	05 Feb 17	09:03	20m	CW	CT9/G3TXF	599	599
ZM4G	05 Feb 17	08:46	15m	CW	CT9/G3TXF	559	599
ZM4G	04 Feb 17	07:31	40m	CW	CT9/G3TXF	449	599
ZL2IFB	24 Jan 16	09:47	40m	CW	G3TXF	599001	99785e
ZL2IFB	12 Dec 15	20:45	10m	CW	PJ6/G3TXF	599001	599
ZL2IFB	09 Dec 15	20:27	30m	CW	PJ6/G3TXF	599	599
ZL2IFB	30 Aug 15	05:29	40m	CW	G3TXF	599013	157
ZL2IFB	03 Jun 15	00:51	17m	CW	ER/G3TXF	559	599
ZL2IFB	14 Mar 15	20:21	20m	CW	V5/G3TXF	005	669
ZL2IFB	14 Mar 15	20:14	15m	CW	V5/G3TXF	004	664
ZL2IFB	28 Jan 15	07:27	80m	CW	G3TXF	589	599
ZL2IFB	27 Jul 14	06:15	20m	CW	TF/G3TXF	599014	99 159
ZL2IFB	27 Jul 14	03:33	40m	CW	TF/G3TXF	599003	99 139
ZL2IFB	13 Jul 14	09:21	20m	CW	G3TXF	60	27
ZL2IFB	17 Feb 14	08:25	15m	CW	6W/G3TXF	599	599
ZL2IFB	15 Feb 14	22:48	10m	CW	6W/G3TXF	559	559
ZL2IFB	02 Feb 14	07:14	40m	CW	EA8/G3TXF	599	599
ZL2IFB	01 Feb 14	19:32	20m	CW	EA8/G3TXF	559	599
ZL2IFB	11 Dec 13	18:38	12m	CW	PJ7/G3TXF	339	449
ZL2IFB	11 Dec 13	18:35	17m	CW	PJ7/G3TXF	559	599
ZL2IFB	07 Dec 13	09:05	30m	CW	PJ7/G3TXF	559	599
ZL2IFB	28 Sep 13	20:16	15m	CW	PJ4/G3TXF	599	599
ZL2IFB	25 Sep 13	00:27	17m	CW	PJ2/G3TXF	599	599
ZL2IFB	21 Aug 13	19:06	80m	CW	G3TXF	569	569
ZL2IFB	02 Jun 12	20:57	30m	CW	GI3TXF/P	559	599
ZL2IFB	09 Dec 11	18:27	10m	CW	G3TXF	559	579
ZM4G	21 Aug 10	23:01	20m	RTTY	G3TXF	599	599
ZM4G	24 Jul 10	21:09	20m	CW	G3TXF	599	599
ZM4G	14 Mar 10	07:40	40m	CW	G3TXF	599	599
ZM4G	14 Mar 10	06:06	80m	CW	G3TXF	599	599
ZM4G	13 Mar 10	18:37	20m	CW	G3TXF	599	599
ZL2IFB	03 Apr 09	07:07	15m	CW	VK4/G3TXF	599	599
ZL2IFB	26 Mar 08	19:06	30m	CW	OE5/G3TXF	559	599
ZL2IFB	02 Feb 08	07:45	40m	CW	FJ/G3TXF	599	599
ZL2IFB	05 Nov 07	08:38	40m	CW	JW/G3TXF	549	599
ZL2IFB	05 Nov 07	07:32	40m	CW	JW/G3TXF	339	599

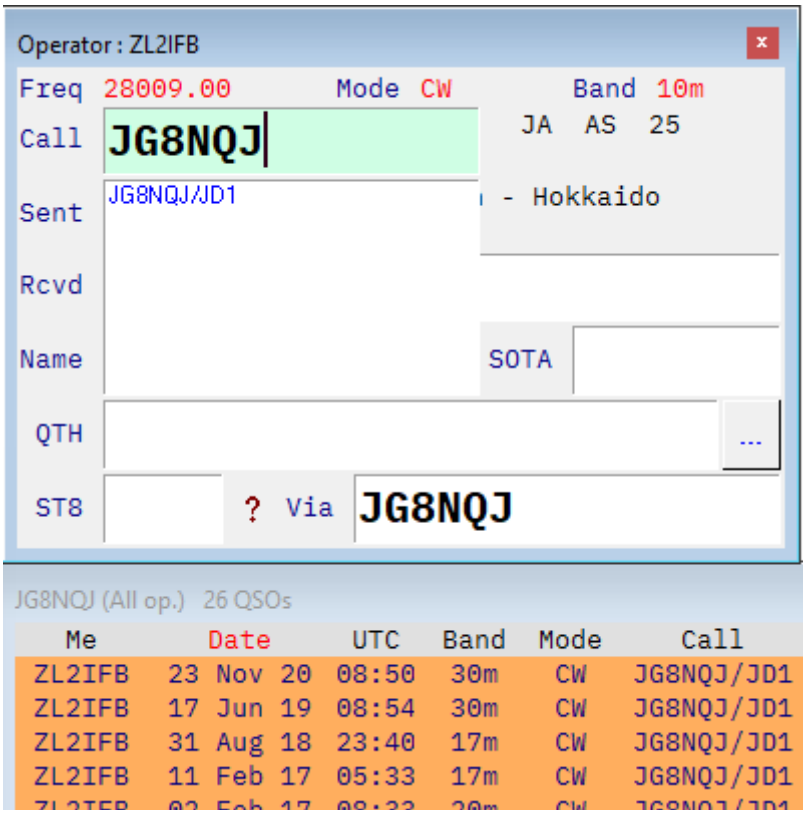
Copyright © 2026 Robert C. Furzer K4CY

Q. I'm *sure* I've worked him before, so why is the previous QSOs pane is empty?

A. In order to find and display previous QSOs in the Previous QSOs pane, Logger32 uses its internal statistics table … so, if you are *certain* there are previous QSOs in the open log that are not being shown, try recalculating your statistics.

Aside from the familiarity of a callsign, there are other indications that it has indeed been logged despite the yawning emptiness of the Previous QSOs pane:

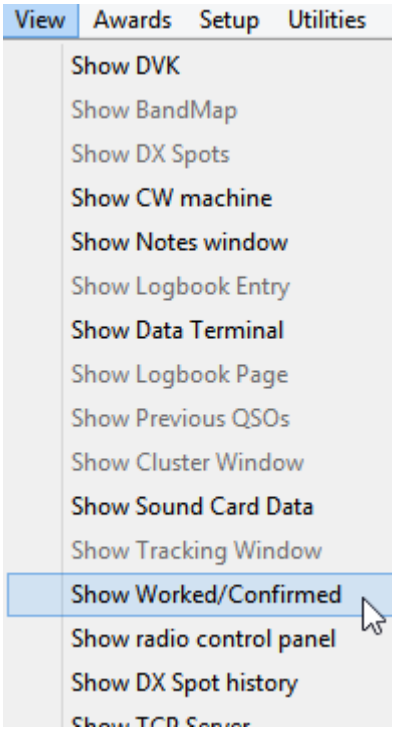
- A logbook search for the callsign should, of course, find and display the QSO or QSOs as a block;
- Information previously recorded in logged QSOs with the station (such as the operator's name, IOTA reference and QSL route) may be carried forward *i.e.* transferred to the Log entry pane when you are next logging him, if so configured in the QSO mask.
- The callsign should appear in blue under the Call field thanks to the callsign preview function, if so configured, possibly with location modifiers ▼



Copyright © 2026 Robert C. Furzer K4CY

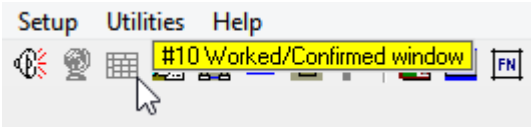
10 Worked/confirmed table

> "The goal is to turn data into information, and information into insight"
>
> *Carly Fiorina*

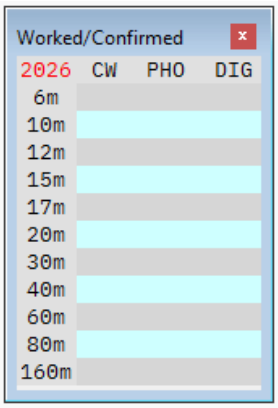


◀ Use **View ⇨ Show Worked/Confirmed** …

… or click toolbar icon #10 ▼



… to open a window like this ▼



Your worked/confirmed table probably has different modes and bands, and may show "ALL" rather than "2026" in the top left corner. It may have information in various cells of the table and scroll bars … so it may look quite different to mine.

This chapter explains how to customize and get the most out of this *powerful* DXing tool.

10.1 The power of worked/confirmed

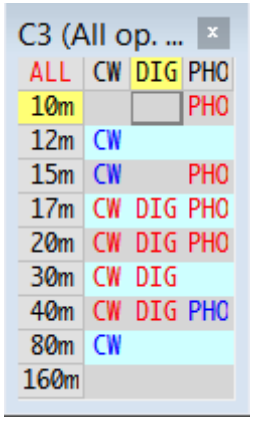
The worked/confirmed table shows, at a glance, which band and mode combinations ("slots") have been filled for the DXCC entity you are currently logging, or (having clicked a QSO in your logbook) the one you have worked and logged previously.

The window is populated (*i.e.* it displays data) as soon as a recognized prefix is entered into the Call field of the log entry pane, unless that particular entity has not been logged before … so an empty worked/confirmed table means you are logging a new one, an exciting piece of information for a DXer right there!

The table resembles a multiplier checklist in a multimode contest – and can be used for that purpose if you are using Logger32 to log a casual contest, starting with a new logbook.

Copyright © 2026 Robert C. Furzer K4CY

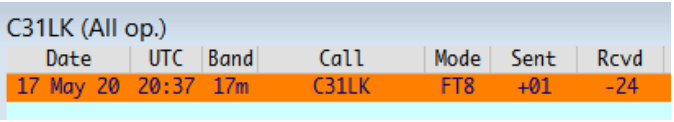
Suppose for example I see a spot for C31LK highlighted on my UDP BandMap when I'm monitoring 10m on FT8. The highlighting tells me I haven't worked Andorra on 10m digital. Have I worked Andorra on the other high HF bands? Who have I worked, and on what modes? Clicking the C31LK spot on the UDP BandMap tells Logger32 to send a 'reply' message, triggering JTDX to call him when he finished his QSO or calls CQ … and puts his callsign into the Call box of the log entry pane … and populates the worked/confirmed table with information about the Andorra QSOs in my log ▶



While the computer patiently awaits its opportunity to call C31LK, I review the information to answer those and other questions.

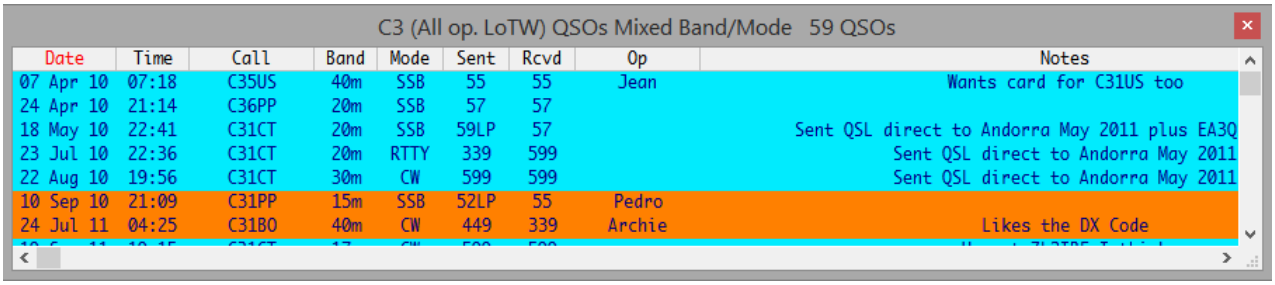
Looking across at the 10m row in the worked/confirmed table, first of all, the red "PHO" tells me that I have worked a C3 on one of the phone modes on 10m – almost certainly SSB since I have only ever made a handful of AM and FM QSOs. The text being red tells me at least one of my 10m phone QSOs with a C3 has been confirmed – otherwise it would be blue meaning I have worked a C3 in that slot but not (yet!) received a confirmation. So, I know I am *capable* of working Andorra on 10m if the path opens … and today, thanks to the weak signal capabilities of FT8, it seems I'm in with a chance.

Looking now down the DIG column, the red "DIG" cells mean I have worked and confirmed C3's on 17, 20, 30 and 40m using digital modes – possibly FT#, JT65, RTTY, PSK or … If it matters, I could reconfigure the worked/confirmed table to show a separate column for each mode, perhaps just RTTY if I am chasing an RTTY award, with the remaining digimodes grouped together into a consolidated DIG column.



◀ The previous QSOs pane tells me I have worked and logged C31LK, specifically, once before, on 17m FT8 a few months ago: he was strong with me (I sent him +1 dB) but his report of -24 dB indicates I was barely readable there. Maybe he has a high noise floor from the city … or perhaps he runs a fair bit of power on FT8. Either way, my chances of making it on 10m are looking slim. On the upside, he has already confirmed the QSO on LoTW, so *if* we make it today we can both expect a prompt confirmation. That leaves me weighing up whether to invest much time and effort into making a 10m FT8 QSO with him today, or to chase any other slot-fillers that appear on my BandMaps.

Returning to the worked/confirmed table, I click one of the empty cells for a quick look at *all* my logged C3 QSOs – 59 of them according to the caption of the 'generic QSOs' window that duly appears ▼

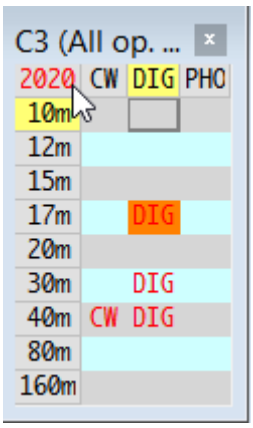


There are clearly a few active HF DXers in Andorra, so if I don't make it with C31LK, all is not lost: other C3's may turn up. On average, I make about 6 Andorran QSOs per year.

Copyright © 2026 Robert C. Furzer K4CY

This section was written in 2020. The corner cell showed the then-current year.

Clicking "ALL" in the top left corner of the worked/confirmed table flipped it to "2020", showing information about the C3 QSOs I have had in that year ▶



4 slots are filled, so *at least* 4 C3 QSOs had been logged (and confirmed!) so far in 2020. Clicking an empty cell listed them ▼

Date	Time	Call	Band	Mode	Sent	Rcvd
27 Jan 20	08:17	C31MF	40m	FT8	-06	-08
03 May 20	06:14	C31CS	30m	FT8	+02	+01
17 May 20	20:37	C31LK	17m	FT8	+01	-24
04 Nov 20	08:04	C31US	40m	CW	579	559

4 QSOs with 4 different C3's on 3 HF bands and 2 modes … hmmmmm …. Meanwhile, C31LK's signal had faded away without us making a QSO. Such is life! I have enjoyed the chase, though, the research keeping my brain gainfully occupied while the computer did its thing. Hopefullly we'll have better luck next time. I wonder if C31LK has worked/confirmed ZL on 10m/FT8, yet?

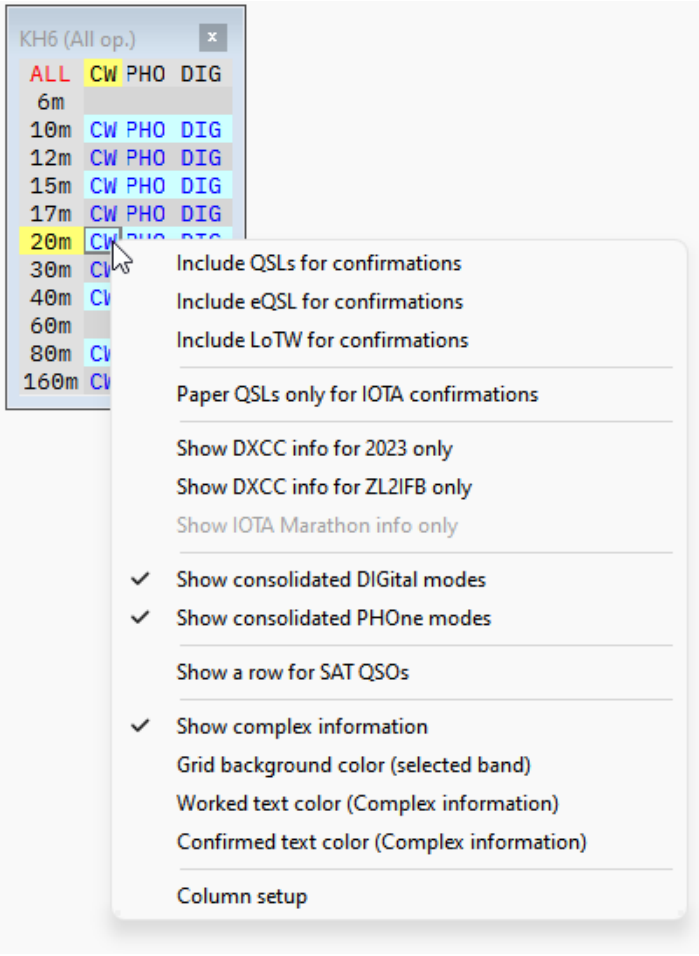
10.2 Configuring the worked/confirmed table

As usual in Logger32, **right-clicking the worked/confirmed table** opens a menu with a stack of configuration options …

10.2.1 Include QSLs|eQSL|LoTW for confirmations

The top 3 options select which types of confirmations are to be shown, if any, by coloring the text entries on the table.

With *no* confirmation options selected, the outlined 20m CW cell contains CW in blue text, simply indicating that I have *worked* KH6 (Hawaii) on 20m CW this year ▶



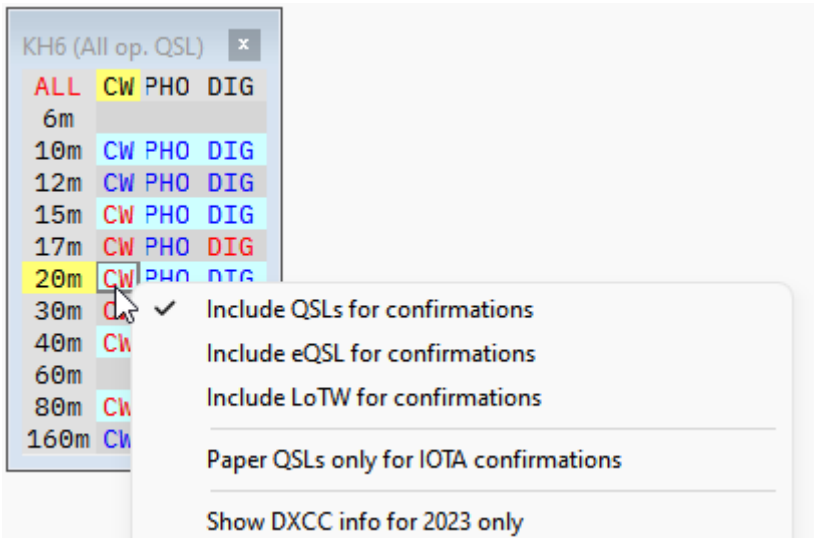
Hinson tip: this setting *also* affects the visual highlighting for new ones in the DX Spots pane.

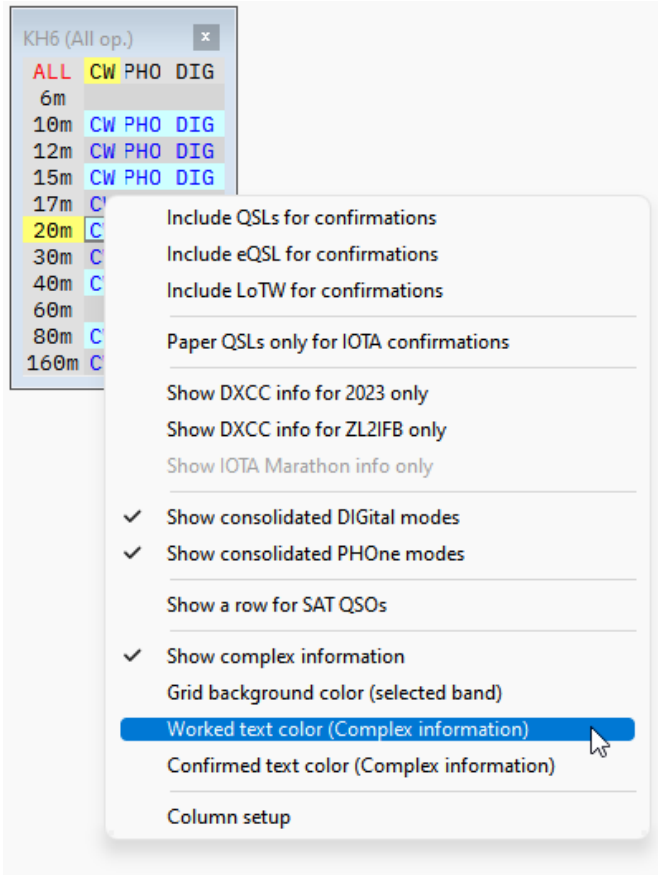
Copyright © 2026 Robert C. Furzer K4CY

Having ticked <**Include QSLs for confirmations**>, the same cell now contains CW in red text, indicating that I have not only worked but received a QSL card from KH6 on 20m CW ▼

The other red-text cells in the table show other band/mode combinations for which I have received and recorded receipt of QSL cards for QSOs with KH6, whereas the blues are worked but not confirmed on QSL cards.

eQSLs and LoTW confirmations work in the same way, either ticking the options individually or in any combination (*e.g.* tick QSLs *and* LoTW to show confirmations valid for the DXCC award).





◀ If you don't like the blue/red text coloring for worked/confirmed slots, there are options on that right-click menu to change the colors.

10.2.2 Paper QSLs for IOTA confirmations

This option affects the highlight colors in the DX Spots pane rather than the worked/confirmed table. For example, if a particular IOTA group has been worked but not yet confirmed by QSL card, DX spots for that IOTA will be highlighted.

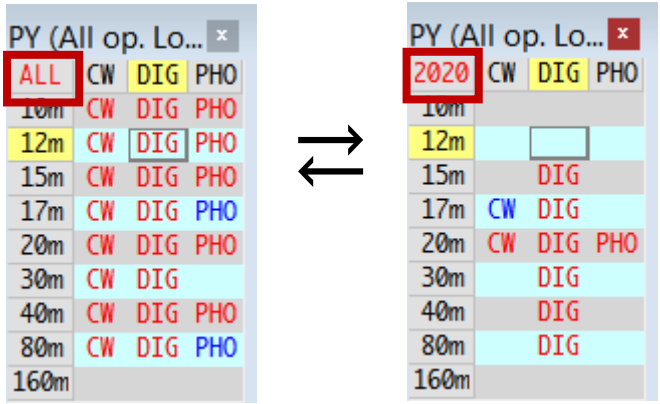
Copyright © 2026 Robert C. Furzer K4CY

10.2.3 Show DXCC info for [this year] only

This option causes the worked/confirmed table and DX Spots pane to display information based only on QSOs made *within the current UTC calendar year* rather than all-time *i.e*. the entire open logbook.

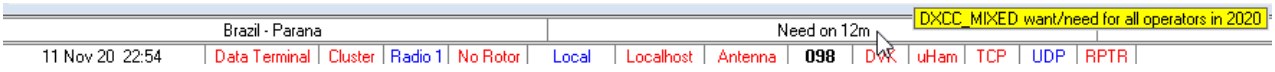
'The current UTC year' is derived from your PC's clock, and the statistics are updated automatically on January 1st at precisely 00:00 UTC (not necessarily when *your* new year fireworks go off!).

When selected, the upper left-hand corner of the worked/confirmed table changes from ALL to [this year]. **Click the corner cell to toggle back and forth between these settings**: the table is updated accordingly ▶



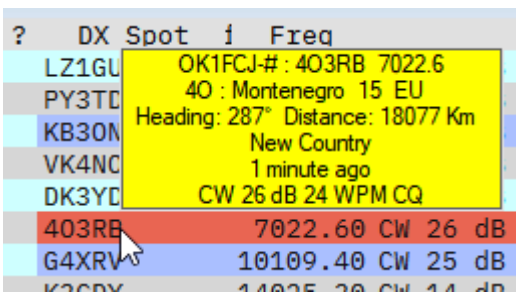
This makes it easy to track your performance within the year, for the DX Marathon or other annual challenges. It affects the:

- Information shown in the worked/confirmed table.
- DXCC award table (the current year option at the bottom of the table is selected by default, but can be changed back to all-time).
- DXCC Challenge award table.
- DX Spots pane and BandMaps (*e.g*. the **X** denoting spotted stations you have already worked is only displayed if you have worked them *this year*).
- DX Activity window.
- Visual highlighting of new ones in various windows *e.g.* if you have chosen to include QSLs, eQSLs and/or LoTW confirmations on the worked/confirmed table, stations that have *not* yet been confirmed by any of the selected methods in the current year are highlighted in the DX spots window, the BandMaps *and* noted on the DX information bar near the bottom of Logger32's main window ▼



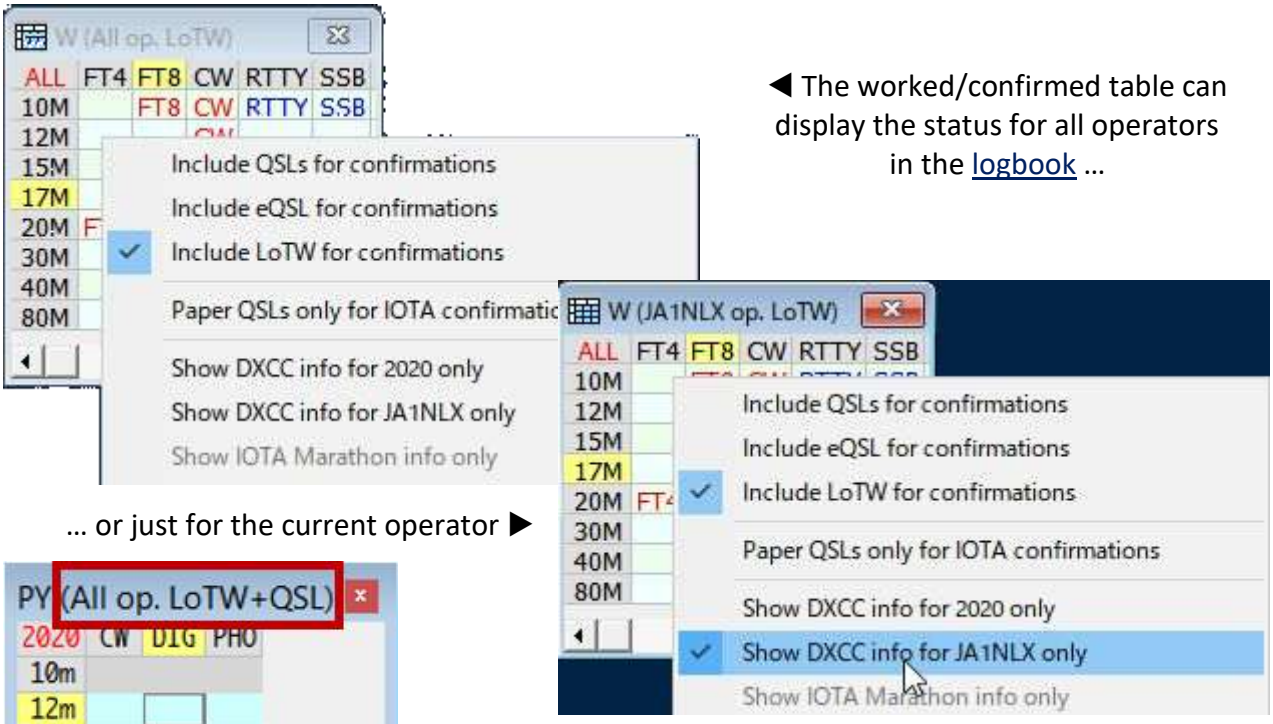
Mouseover the notification message or DX spots for pop-ups with more detail ▲▼

I have worked and confirmed Montenegro many times before … but *not yet* this year, so (with *my* color settings) the 4O3RB DX spot was shown in red and the PC sounded an audio alert ▶



Copyright © 2026 Robert C. Furzer K4CY

10.2.4 Show info for [current operator] only



◀ The worked/confirmed table can display the status for all operators in the logbook …

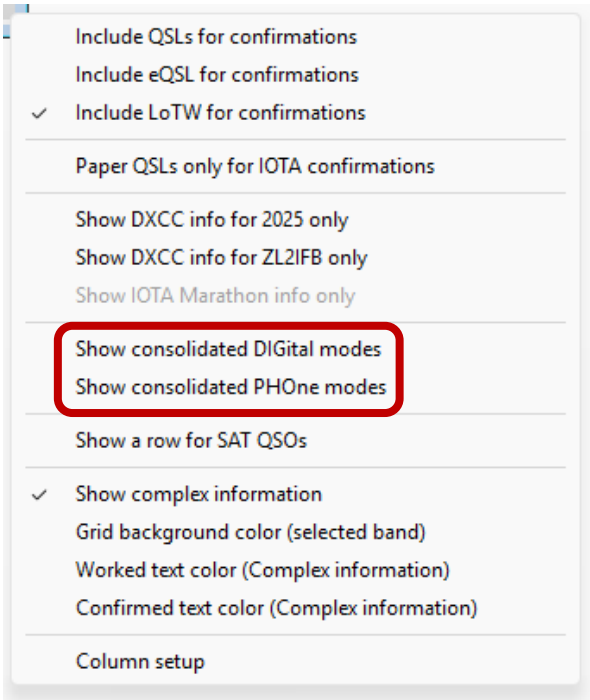
… or just for the current operator ▶



◀ The status is shown in the captions of the worked/confirmed table and previous QSOs pane, along with a succinct reminder of which "Included confirmations" options are currently set *e.g.* LoTW and QSL cards in this instance[157].

10.2.5 Show consolidated DIGital|PHOne modes

If you use several digital modes (*e.g.* FT#, RTTY) or phone modes (*e.g.* SSB, AM, FM), you may prefer to see your consolidated (grouped) digital and/or phone statistics rather than the details for each individual mode ▶

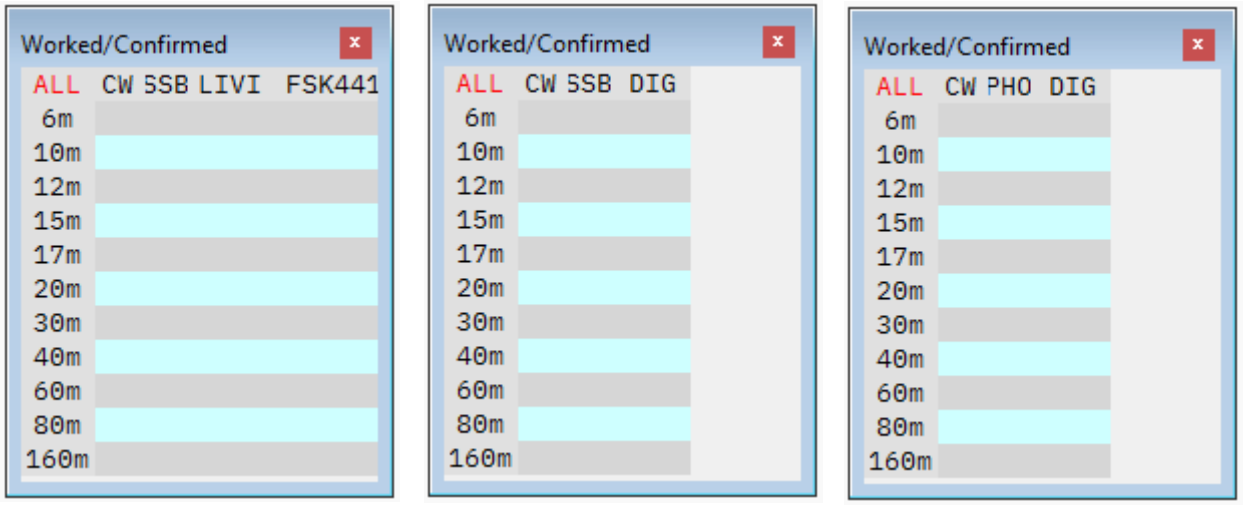


To make this work, we first need to tell Logger32 which ADIF modes/submodes we consider to be "digital" or "phone". It's not entirely obvious *e.g.* should digitized speech be classed as "digital" or "phone"? Given that the carrier is either transmitted or not, is CW "digital"? Logger32 lets us decide for ourselves.

[157] My table is normally too narrow to see the whole caption: I had to grab the righthand border and drag it out to read the whole caption and grab this image, then dragged it back. With so much going on in Logger32, screen space is a scarce commodity!

Copyright © 2026 Robert C. Furzer K4CY

Next, decide whether to consolidate digital, phone, neither or both modes, ticking or unticking the relevant options in the middle of the worked/confirmed table right-click menu:



No consolidation

Consolidate digital modes only[158]

Consolidate digital *and* phone modes

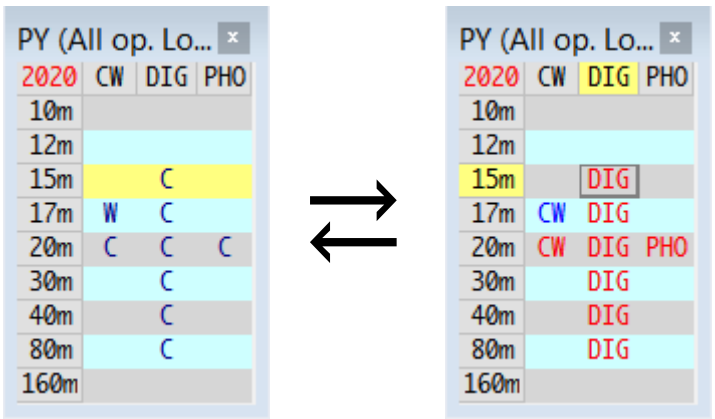
Notice that the unconsolidated table fills the space, and in fact extends further to the right to show all the modes listed in my Bands & modes table. Although I have configured mine without a horizontal slider, I can display more of the worked/confirmed table by dragging the right-hand margin further out to the right[159].

In contrast, the two consolidated tables only show 3 mode columns, and I can drag the right-hand margin to the left to conserve valuable screen space.

10.2.6 Show complex information

The body of the worked/confirmed table can display simple or more complex information. Right-click anywhere within the table and toggle <**Show Complex Information**> to compare them ▼

Simple **Complex**



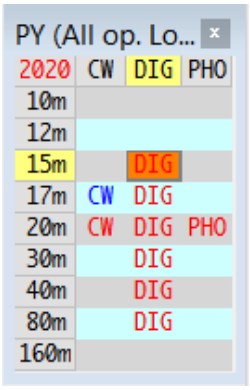
[158] I have just a few SSB QSOs in my log, with none on FM, AM *etc*. You may see extra columns here.

[159] Presumably "LIVI" means OLIVIA, which is odd as, although it is present in my Bands & modes table, I don't have *any* Olivia QSOs in my logbook. Unfortunately, the columns have fixed width and don't expand to show their full headings as I extend the table width. I'm not sure of the column sort order either.

Copyright © 2026 Robert C. Furzer K4CY

In the **simple** form, "W" and "C" show the band and mode combinations on which PY (Brazil) have either been **W**orked-but-not-yet-confirmed, or worked-*and*-**C**onfirmed, respectively. The entire row for the band currently in use at the time (15m) was highlighted in yellow.

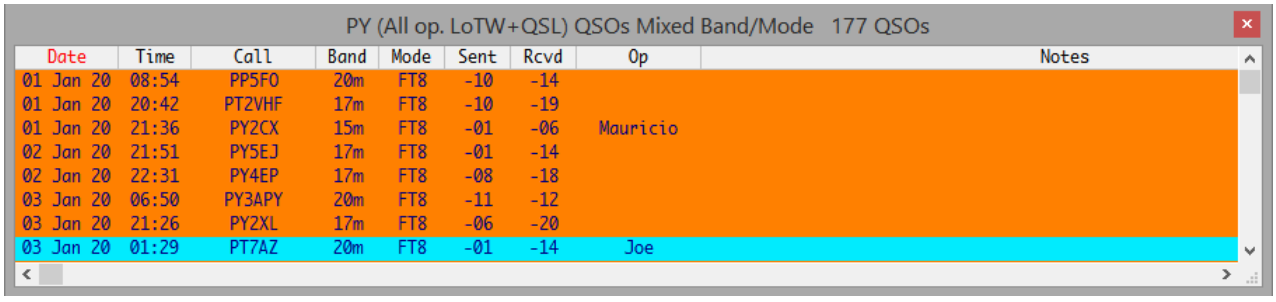
In the **complex** form, blue text shows the band/mode slots on which PY has been worked but not yet confirmed, while red text shows the band/mode slots that have been worked-and-confirmed. The outlined cell with yellow highlighted row and column names shows the band *and* mode currently in use at the time (15m digital).



Any band/mode slots on which *the callsign currently in the log entry pane Call field* (that specific station, not just anyone from that country) has been worked/confirmed have orange backgrounds ▶

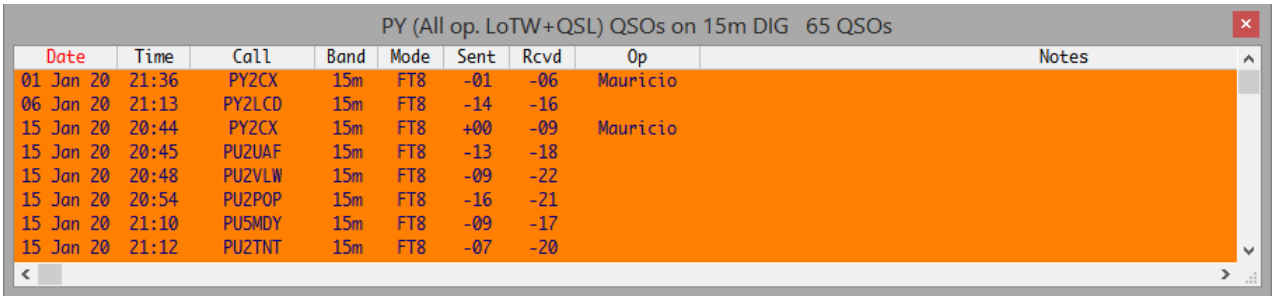
10.2.7 Additional information window

Clicking any *filled* cell in the worked/confirmed table shows the corresponding QSOs for the selected slot, from the currently open logbook. For example, clicking the orange slot on the table above brought up this window ▼ showing all my PY QSOs on 15m digital modes in 2020[160]


PY (All op. LoTW+QSL) QSOs Mixed Band/Mode 177 QSOs

Date	Time	Call	Band	Mode	Sent	Rcvd	Op	Notes
01 Jan 20	08:54	PP5FO	20m	FT8	-10	-14		
01 Jan 20	20:42	PT2VHF	17m	FT8	-10	-19		
01 Jan 20	21:36	PY2CX	15m	FT8	-01	-06	Mauricio	
02 Jan 20	21:51	PY5EJ	17m	FT8	-01	-14		
02 Jan 20	22:31	PY4EP	17m	FT8	-08	-18		
03 Jan 20	06:50	PY3APY	20m	FT8	-11	-12		
03 Jan 20	21:26	PY2XL	17m	FT8	-06	-20		
03 Jan 20	01:29	PT7AZ	20m	FT8	-01	-14	Joe	

Clicking an empty 15m cell in the worked/confirmed table displayed an additional information window listing *all* my PY QSOs on 15m so far in 2020, regardless of mode ▼


PY (All op. LoTW+QSL) QSOs on 15m DIG 65 QSOs

Date	Time	Call	Band	Mode	Sent	Rcvd	Op	Notes
01 Jan 20	21:36	PY2CX	15m	FT8	-01	-06	Mauricio	
06 Jan 20	21:13	PY2LCD	15m	FT8	-14	-16		
15 Jan 20	20:44	PY2CX	15m	FT8	+00	-09	Mauricio	
15 Jan 20	20:45	PU2UAF	15m	FT8	-13	-18		
15 Jan 20	20:48	PU2VLW	15m	FT8	-09	-22		
15 Jan 20	20:54	PU2POP	15m	FT8	-16	-21		
15 Jan 20	21:10	PU5MDY	15m	FT8	-09	-17		
15 Jan 20	21:12	PU2TNT	15m	FT8	-07	-20		

The window caption showed the total number of QSOs if they are not all displayed in the window (in which case I could count them myself!).

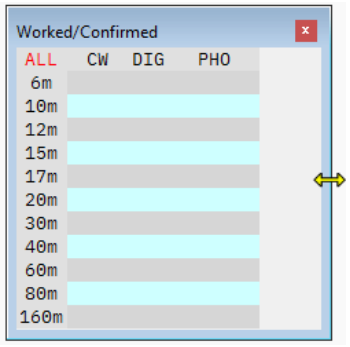
Clicking any QSO in the additional information window opens the previous QSOs pane listing all previous QSOs with that station, and the clicked QSO is highlighted in the logbook.

Right-clicking the additional information window lets you customize the grid layout in the same way as for the logbook and previous QSOs pane *e.g.* to show/hide and rearrange the columns, change the column header text *etc*.

[160] Did you notice "2020" in the top left corner cell? Pay attention! The image was grabbed in 2020.

Copyright © 2026 Robert C. Furzer K4CY

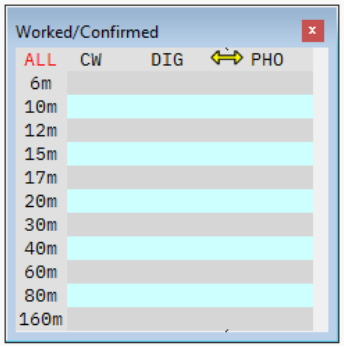
10.2.8 Changing worked/confirmed mode column widths



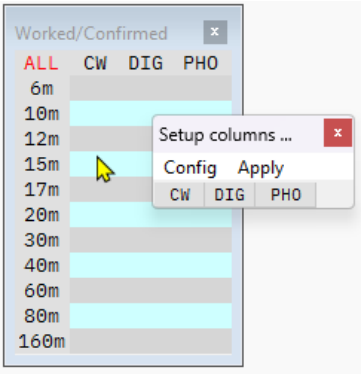
◀ First, click and drag the right-hand border of the worked/confirmed window to widen it, making space.

Now point the mouse at the [invisible] border *between* column headers until it changes to a double-headed arrow, then click and drag the border to narrow or widen the column to the left ▶

And finally, readjust the window width to your liking.



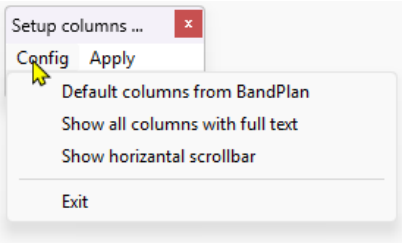
10.2.9 Changing worked/confirmed mode column order



There are two ways to change the order or sequence of the mode columns.

First, the hard way. Right-click the worked/confirmed window, then click <**Column setup**> at the bottom of the menu to open ◀ this little 'Setup columns …' menu.

Click <**Config**> on its menu ▶ to open a further menu with 4 options.



- <**Default columns from BandPlan**> sorts the mode columns according to the sequence in which "Y" appears in the <**Stats**> column of your BandPlan ▼



In order to show every band+mode combination, it is only necessary to mark one "Y" for each mode (on any band), and one "Y" for each band (on any mode): any more "Y's" are superfluous. In this example, the topmost "Y" is for CW (on 6m as it happens). The next one is for SSB (on

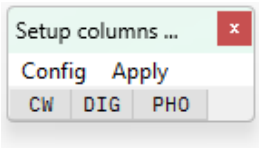
Copyright © 2026 Robert C. Furzer K4CY

10m), then RTTY (on 12m) – so that is the *default* sort order of the **mode** columns shown in the worked/confirmed window[161].

- <**Show all columns with full text**>: when the <**Column setup**> window is opened, its column widths match your worked/confirmed table. If you have narrowed the worked/confirmed column widths by dragging the right-hand margins of the headers such that the mode description is no longer readable, <**Show all columns with full text**> resizes the columns and labels to their full widths.
- <**Show horizontal scrollbar**>: by default, a scroll bar appears along the bottom of the worked/confirmed table if it is narrowed, allowing you to scroll the window contents to the left or right. If this option is *de*selected, followed by <**Exit**> then <**Apply**>, the horizontal scrollbar is disabled so you can narrow or widen the window to fit the screen space even without pixel-precise mousing.
- <**Exit**> closes the menu.
- Click <**Apply**> to apply any changes to your worked/confirmed window and see their effect.

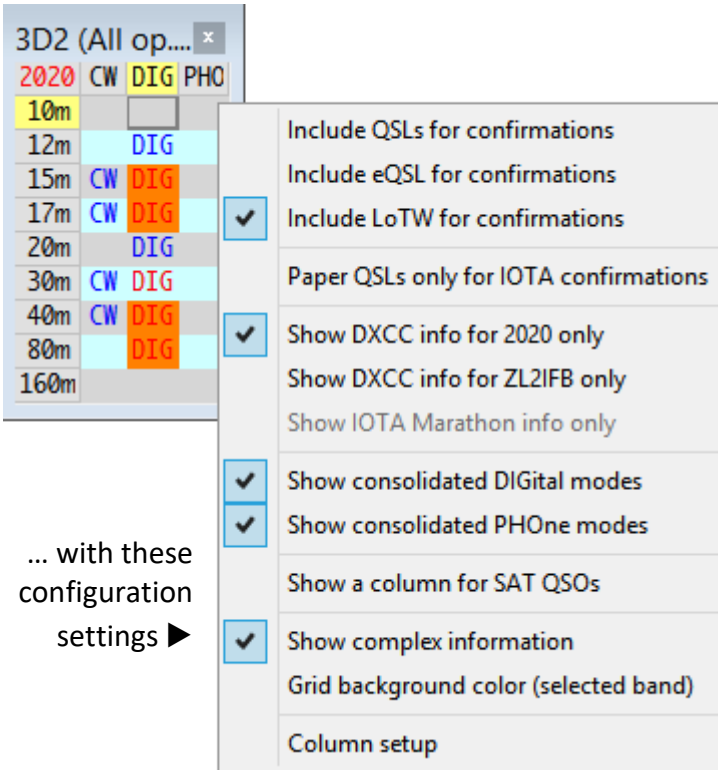
Alternatively, there is a *much* easier way to change the column widths:

Right-click the worked/confirmed window, then click <**Column setup**> at the bottom of the menu to open this little window ▶



Now simply click and drag the columns in the little window into your desired order, click <**Apply**> then close the little window using the corner X. All done! Easy, huh?

With the bandplan above, my worked/confirmed table for 3D2 (Fiji) looked like this back in 2020 ▶



... with these configuration settings ▶

[161] The **band** rows are always sorted numerically, either ascending or descending.

Copyright © 2026 Robert C. Furzer K4CY

10.3 Worked/confirmed table FAQs

Q. The worked/confirmed *band* column is missing: where did it go?

A. The column widths are stored automatically in an .INI file … that you can edit manually:

1. Close Logger32.
2. With an ASCII editor such as Notepad edit *C:\Logger32\Logger32.INI*
3. Look for the section starting [wndWorkedConfirmed Settings]
4. Find the line *Column 0 Width=xxx*
5. Change *xxx* to, say, 600.
6. Save the change and close the file.
7. Run Logger32.
8. OK now? If not, try again with a different *xxx* value.

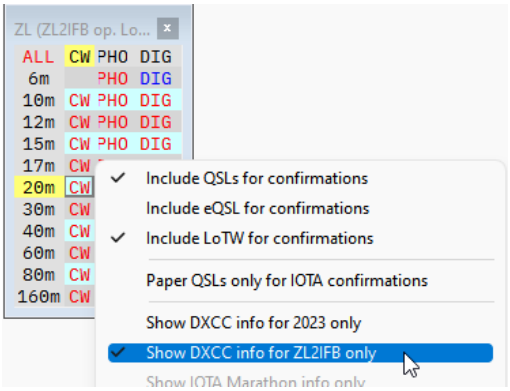
Q. How do I change the columns shown in the worked/confirmed table?

A. Modify your bandplan *i.e.* the Bands & Modes table. Put at least one "Y" in the stats column for every QSO using that mode to be accounted for (not necessarily shown separately: digital modes may be combined under the DIG heading, and voice modes under the PHONE heading).

Q. How do I search my log for QSOs with a particular country on a given mode and/or band?

A. You can use the worked/confirmed table, like this:

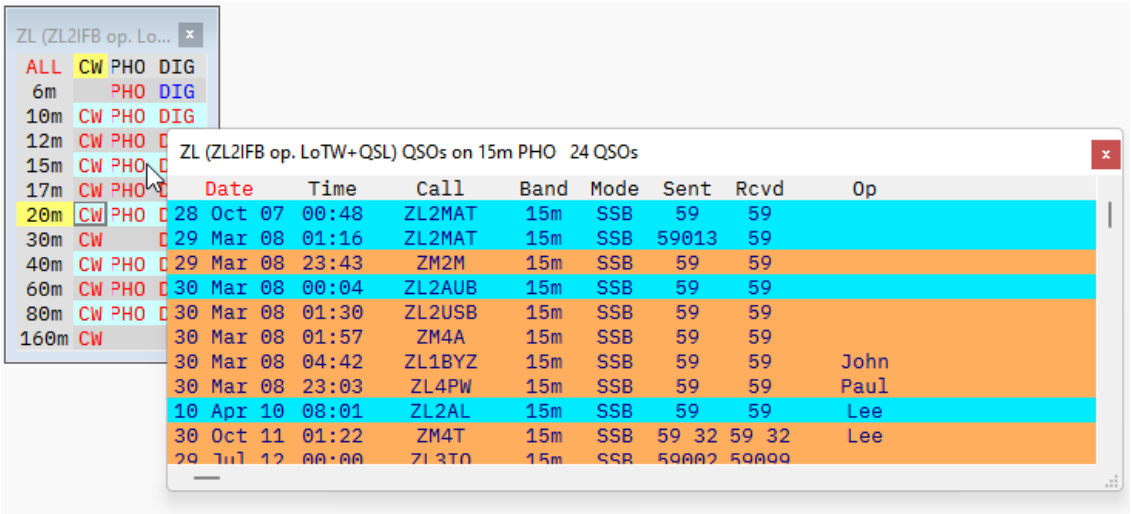
- Display the worked/confirmed table using **View ⇨ Show Worked/Confirmed**;
- If the band/s and mode/s of interest are not shown in the worked/confirmed table, configure it so through the Bands & Modes table (see the *previous* FAQ!)[162];
- Decide whether you want to search only for QSOs logged in the current calendar year, or logged for all time, clicking to toggle the top left corner of the worked/confirmed table accordingly;
- If you *only* want to search for QSOs logged by the current operator's callsign (as noted in the caption to the log entry pane), right-click the body of the worked/confirmed table ▶ and click to tick <**Show DXCC info for [**current operator**] only>**, otherwise it will find all relevant QSOs in the open log regardless of who logged them;
- Decide whether you are only interested in specific types of confirmation (*e.g.* LoTW), ticking them in the same right-click menu.



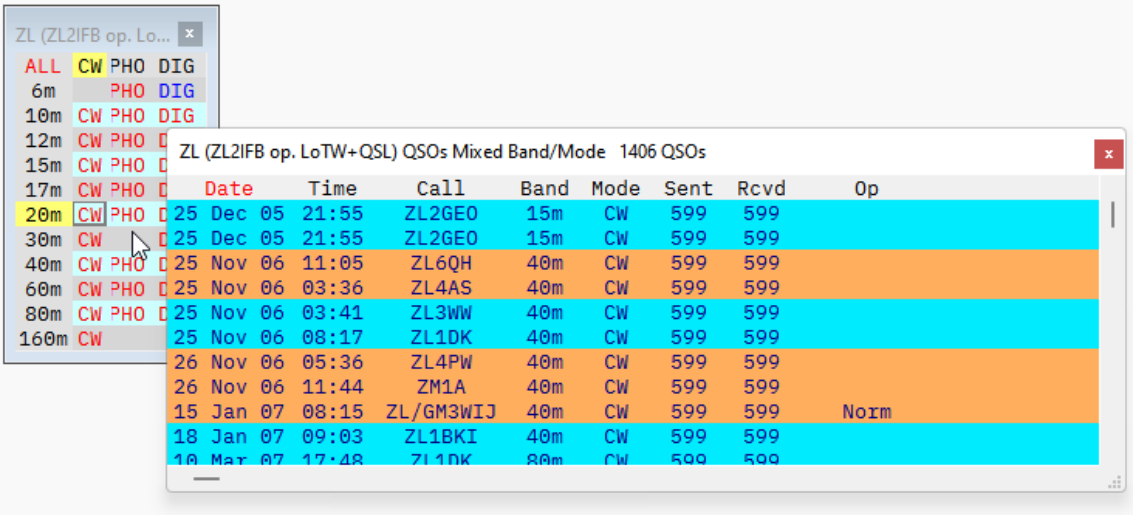
[162] Consolidated mode groups such as phone or digital will display all relevant QSOs with modes in the selected groups. If you specifically want to check individual modes (such as AM or RTTY), you can either reconfigure the worked/confirmed table to show those columns, or simply scroll manually through the found mode-group QSOs looking for QSOs on the particular modes that interest you.

Copyright © 2026 Robert C. Furzer K4CY

- Put at least the *prefix* of a callsign for the country of interest into the Call field of the log entry pane, checking that Logger32 looks up and determines the DXCC entity of interest (but don't hit enter to log it!)[163];
- Click the relevant cell/s of the worked/confirmed table:
  - If a cell contains any text, that means you have logged at least 1 QSO in that mode+band slot. Clicking such a populated cell displays a table listing those logged QSO/s *e.g.* here I clicked the 15m PHO cell to list all the ZLs that I have contacted on 15m phone ▼



  - If instead you click an empty cell, *all* your logged QSOs with that DXCC entity are shown, regardless of band or mode *e.g.* clicking the empty 30m PHO cell ▼

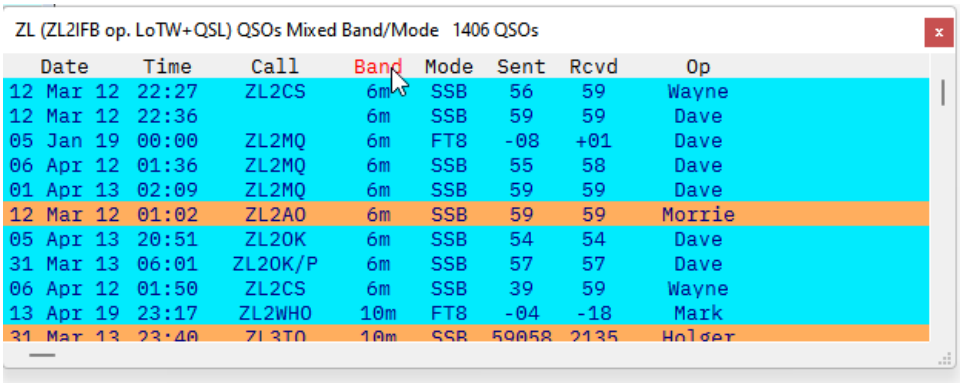


  - Notice the caption for the results panel tells us what is shown, including how many QSOs the search has dug up from the open logbook;
  - Rows are colored in the same way as QSOs in your logbook;

[163] You may need to get creative with the prefix or complete callsign for some entities. To find your QSOs with the North Cook Islands, for instance, "E5" is an ambiguous prefix that defaults to the South Cooks, whereas "E5\N" tells Logger32 to look for rarer QSOs with the North Cooks instead.

Copyright © 2026 Robert C. Furzer K4CY

- Initially, QSOs are listed in ascending date and time order. Clicking a column header re-sorts the list by that column, turning the sorted column's header text red *e.g.* ▼



Copyright © 2026 Robert C. Furzer K4CY

11 Callsign lookups

> "The internet is just a passing fad"
>
> *Bill Gates*

You can look up information about a station you hear, are working or have logged using:

- An online callsign database such as HamQTH or QRZ (requires Internet connection!);
- A callbook CD-ROM loaded in a CD drive or copied to your hard disk (requires the CD-ROM!);
- A local database of QSL manager information on disk (requires the database!).

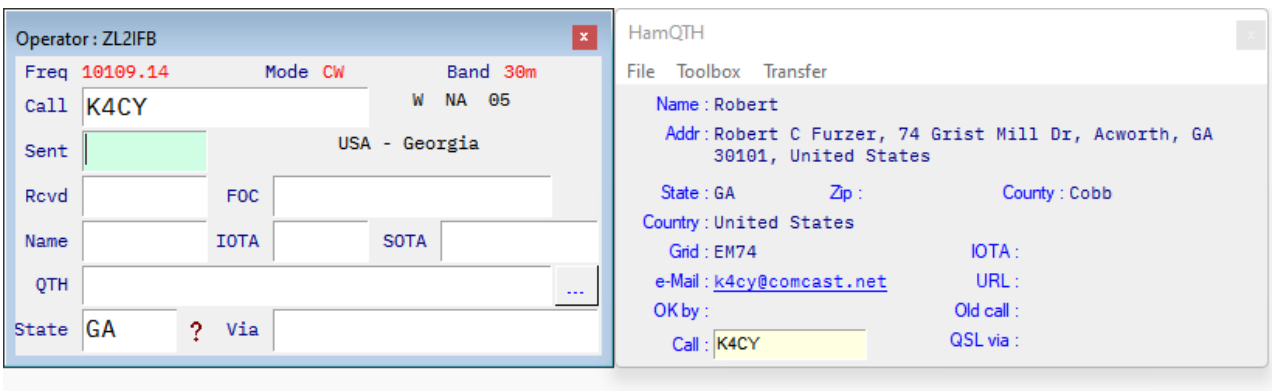
Information such as the person's name, home address and grid square, preferred QSL route *etc*. can (if you wish) be:

- Looked up automatically or on request;
- Displayed on screen (so you know a bit more about the people you are contacting);
- Used to calculate the direction and distance to their stated location;
- Transferred to the log entry pane, ready to be saved when you log the QSO.

11.1 Online callsign database lookups

Provided you have Internet access, Logger32 can automatically check an online database for the callsign of someone you are currently working and logging, displaying details from the corresponding record in the database - if such a record exists that is (some hams haven't got around to registering yet, or elect *not* to share their information online).

With autolookups configured correctly (see below), when you type a callsign into the log entry pane then <**Tab**> or click to another field, or if you click any QSO already in your logbook, that callsign is looked up in an online callsign database for you and information from the corresponding record is displayed in a results pane by Logger32 *e.g.* ▼



11.1.1 'Internal' online lookup function

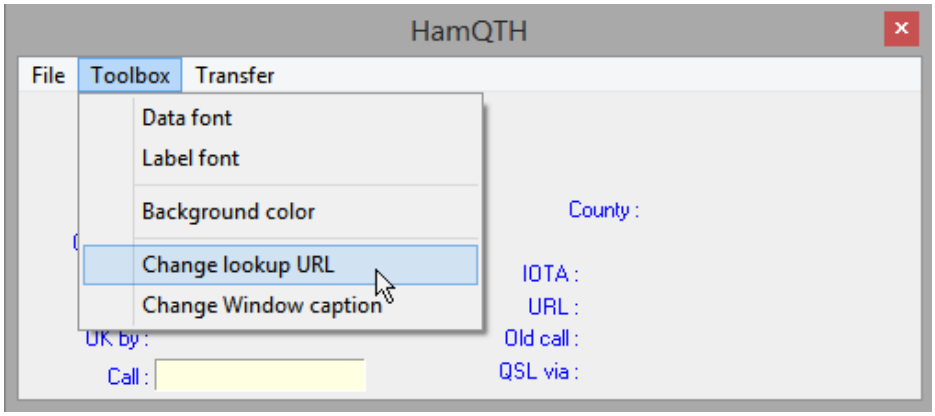
Logger32's 'internal'[164] auto-lookup function is coded to use the online callsign database at HamQTH. If you would prefer to use a different online callsign database, try 'external' lookups instead.

Online callsign lookups are not performed by default - you need to configure this first. Here's how to set up the 'internal' online lookups …

Open the 'internal' HamQTH callsign auto-lookup manually using toolbar icon #23 ▶



A pane appears[165], ready to display information from HamQTH (if any[166]) for a given callsign ▼



HamQTH allows detailed callsign lookups only for registered users, which means to get this function working properly you need to:

1. Browse to HamQTH and register for a free user account, assuming you don't already have one. Your username will be your callsign. Remember your password …
2. From the callsign lookup pane's text menu, open **Toolbox ⇨ Change lookup URL**, then edit the lookup URL to incorporate *your* username and password for HamQTH. It should end up like:

   `https://HamQTH.com/adif.php?u=CCCCCC&p=PPPPPP&callsign=CALLSIGN&prg=LOGGER32`

   where CCCCCC is *your* HamQTH username (your callsign), and PPPPPP is *your* HamQTH password[167]. Do not modify anything shown in blue above.

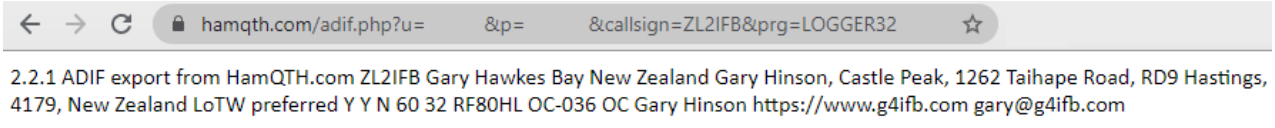
---

[164] 'Internal' indicates that the HamQTH lookup function is built-in to Logger32.

[165] The HamQTH results pane is not available from the <**View**> menu.

[166] Many of us have registered at HamQTH.com, so that the people we contact can look up our information. Have you? It's quick, easy and free! Browse to www.hamqth.com/register.php and complete the form.

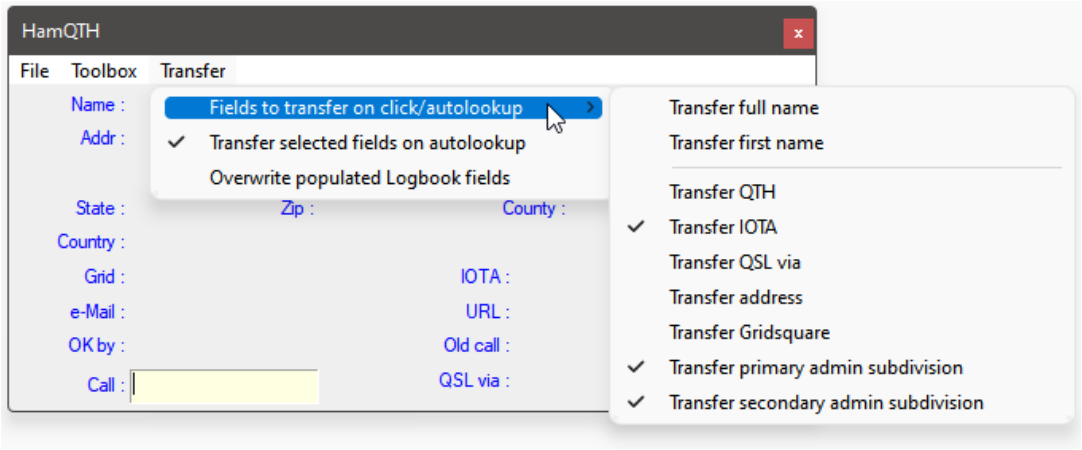
[167] Logger32 dynamically modifies the URL to do the actual lookups, replacing "CALLSIGN" with the callsign currently in the Call field of the log entry pane. You can check that this is working using your web browser: copy the lookup URL with your username and password from Logger32 into the browser address field, replace "CALLSIGN" with an actual callsign such as ZL2IFB, then hit return to open the web page. If it works correctly, you'll see the plain text record for ZL2IFB, which is the information that Logger32 displays in its callsign lookup pane when you are working me *e.g.*



Copyright © 2026 Robert C. Furzer K4CY

Using **Toolbox ⇨ Change Window caption** you can also alter the caption for the results pane as a handy reminder of where the looked-up information comes from.

Next, tell Logger32 what to do with the information it gets from HamQTH ▼

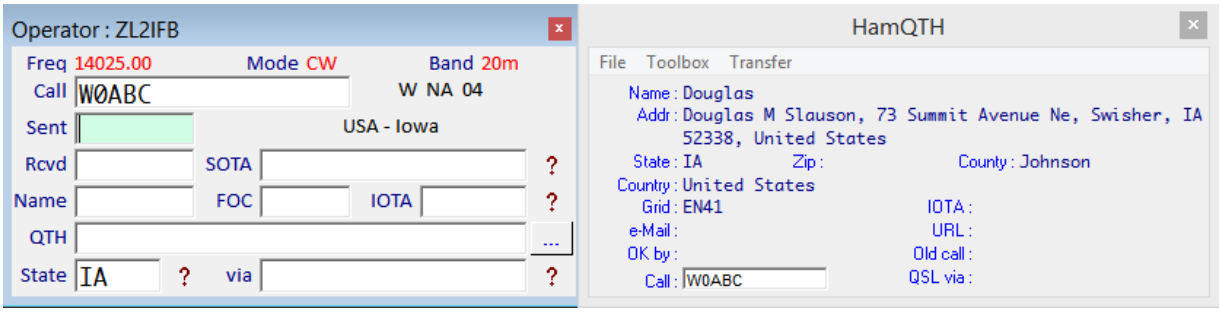


▲ On the results pane menu, <**Transfer**> opens a submenu plus two configuration options:

- **Fields to transfer on click/autolookup**: this submenu lists the nine data fields from which you may copy data from the Internet database to the log entry pane[168] (while you are logging a QSO) or to your logbook (for QSOs already logged). Simply click to select (tick) whichever information fields you want to be transferred.

> **Hinson tip**: the information transfer occurs 'via' the log entry pane. So, for instance, if you want someone's *state* from a callsign lookup to be saved to your log when logging a contact, your log entry pane must have a *state* field configured. Actually showing the field is optional.

- **Transfer selected fields on autolookup**: while you are logging a QSO, ticking this option *automatically* pre-fills the log entry pane with data from the fields you have selected[169].



▲ Despite not having worked W0ABC before, as soon as I entered his callsign and hit <**Tab**>, his details were looked up at HamQTH and displayed in the auto-lookup form, plus his state[170] (or rather, his 'Primary administrative subdivision' in ADIF terms) was typed for me.

---

[168] Configure your log entry pane to show the selected fields, otherwise the information goes nowhere.

[169] Autolookups are limited by the quality of the database. Information in the log entry pane is not yet committed to your log. You can easily correct or delete it before hitting <**Enter**> to log the QSO. After that, you can still edit the QSO in the logbook manually *e.g.* using information from a QSL card or in QSO.

[170] I could easily have ticked to grab any of the other fields from the lookup too … except that normally I prefer to get a person's name, QTH and so on during the QSO. He/she may be using a nickname on-air, and may have moved since saving his details at HamQTH.com

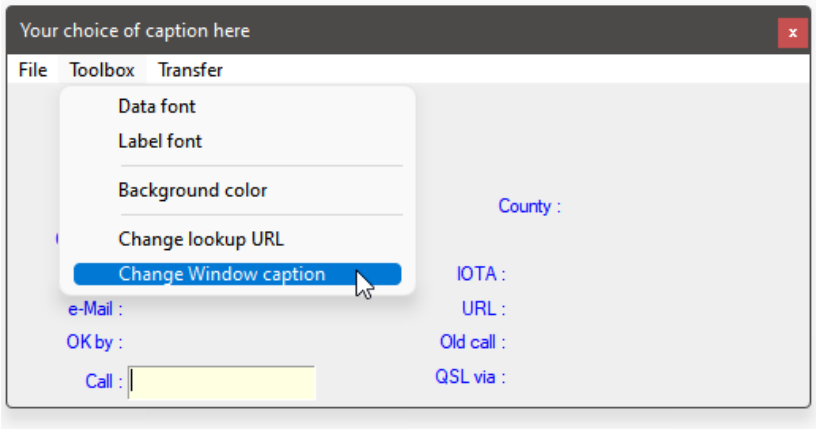
Copyright © 2026 Robert C. Furzer K4CY

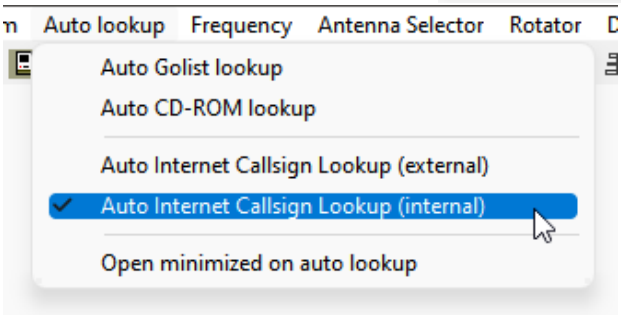
In that example, W0ABC has no IOTA reference. Logger32 determined his bearing and distance from me as well, displaying it on the lower status bar[171].

- **Overwrite populated Logbook fields**: you can pull information from online lookups *directly* into your logbook.

> **Hinson tip**: take care with this option as information retrieved from the online database simply overwrites anything already in the selected fields in your log, without warning. It may be a handy way to correct things if you have been using the wrong fields *e.g.* you have been logging US states under comments or notes, but if you have been logging information correctly, it may replace your manually-typed entries with incorrect, perhaps out of date information retrieved from the online database.

The lookup results pane's **<Toolbox>** menu lets you change the fonts used for the field contents (data) and names (labels), the pane's background color, the lookup URL (see below) and its title (caption) ▶





◀ Finally, tell Logger32 to look up callsigns automatically with the 'internal' function using **Setup ⇨ Auto lookup ⇨ Auto Internet Callsign Lookup (internal)**[172]

11.1.2 'External' online lookups

As well as 'internal' lookups using HamQTH, Logger32 has an **A**pplication **P**rogram **I**nterface allowing the use of 'external' (meaning separate, third-party) apps to handle lookups on other online callsign databases such as:

- L32Lookup by N2AMG, that works with QRZ's HTML (free but limited) lookups, or the XML (paid-for and unlimited) lookups.
- Lookup_QRZ_XML, Lookup_HamQTH_ADIF and Lookup_HamQTH_XML by JA1NLX are self-explanatory.

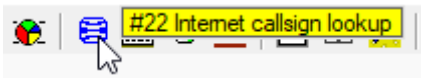
---

[171] His DXCC entity gives a rough location, while his state narrows it down and his 4-character grid square locates him within a few square kilometres – assuming he has accurately determined and shared his grid in his HamQTH.com record. Location information is used, even if it is not transferred to the log entry pane.

[172] You can select either, both, or neither auto-lookup options, as you wish. Requesting, downloading and processing the information takes a little time, a few bytes over the network and some CPU cycles.

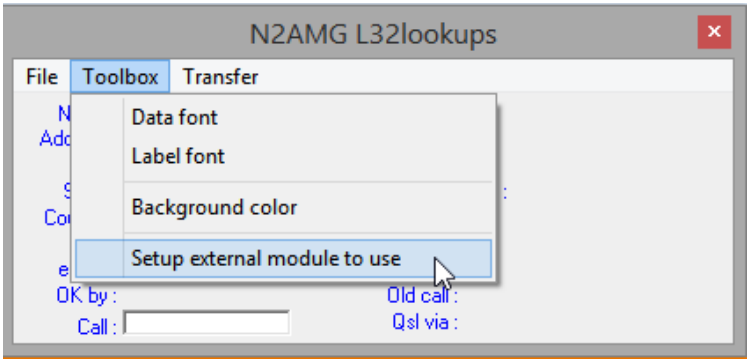
Copyright © 2026 Robert C. Furzer K4CY

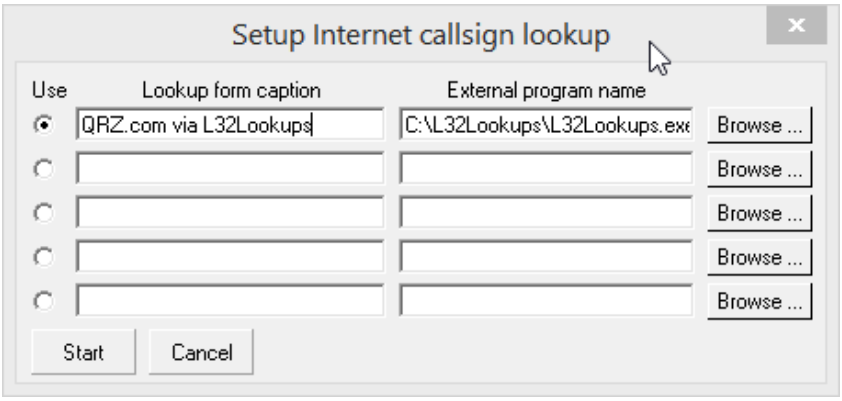
Since these online lookup apps are *not* part of Logger32, you'll need to locate, download and install one. Its installation instructions should lead you through the process.



◀ Having installed your choice of lookup app, open the 'external' Internet callsign lookup window using the toolbar blue globe icon #22.

The <**Toolbox**> menu is where you specify which 'external module' (online callsign lookup app) to use ▶

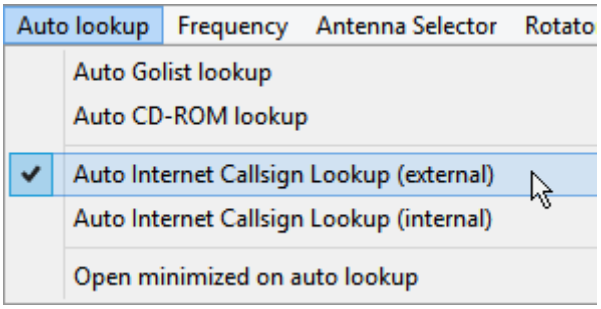




◀ Configure the online lookup app you have installed by browsing to the executable file, and giving the app form a suitable caption/name.

You can install several apps, but Logger32 will only use one at a time – the one with a dot in the <**Use**> column.

Now tell Logger32 to look up callsigns automatically with the 'external' app using **Setup ⇨ Auto lookup ⇨ Auto Internet Callsign Lookup (external)**[173] ▶



11.2 CD-ROM callsign databases

Before the Internet became mainstream, callsign databases were originally compiled and delivered on paper, then on floppy disks and CD-ROMs. Users were charged to recoup the costs of compiling/maintaining and delivering them. Commercial CD-ROMs are still available today, providing a callsign lookup service that does not depend upon Internet connectivity.

[173] You can select either, both, or neither auto-lookup options, as you wish. Requesting, downloading and processing the information takes a little time, a few bytes over the network and some CPU cycles.

Copyright © 2026 Robert C. Furzer K4CY

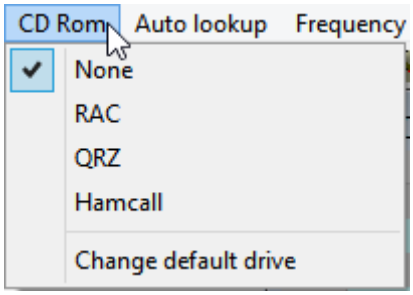
11.2.1 Install CD-ROM database on hard drive

You need not leave the CD-ROM in the drive if the data has been copied onto your hard disk:

- Copy the contents of the CD-ROM data folder from the CD-ROM into a folder called:
  - *C:\Data* for RAC.
  - *C:\Callbk* for QRZ.
  - *C:\ham0\HamCall* for HamCall. The HamCall CD-ROM can install its database on your hard drive: insert the HamCall CD-ROM into the CD drive and the HamCall main menu should appear. Click <**Copy database to hard drive**>.
- Copy the relevant DLL from the CD-ROM into your Logger32 folder *i.e*.
  - *raccd32a.dll* for RAC.
  - *qrz32.dll* for QRZ.
  - *Hamcal32.dll* for HamCall.
- Click **Setup ⇨ CD ROM ⇨ Change default drive** then enter the appropriate drive letter for whichever hard drive you are using for this (*e.g.* C:). Callsign lookups will now be much quicker.

11.2.2 CD-ROM database lookups

Click **Setup ⇨ CD Rom** then select the appropriate type ▶



Unless you <**Change default drive**>, Logger32 expects the CD-ROM to be loaded in the 1st (often the only) CD drive on your computer.



◀ To display the data, click toolbar CD icon #24 ▼

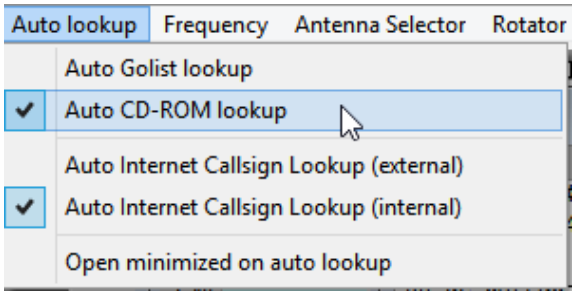


Type a callsign into the Call box and press <**Enter**> to search for and (if found) display the corresponding information from the CD-ROM.

The <**Toolbox**> menu lets us change fonts and colors.

The <**Transfer**> menu lets us transfer looked-up information to the log entry pane.

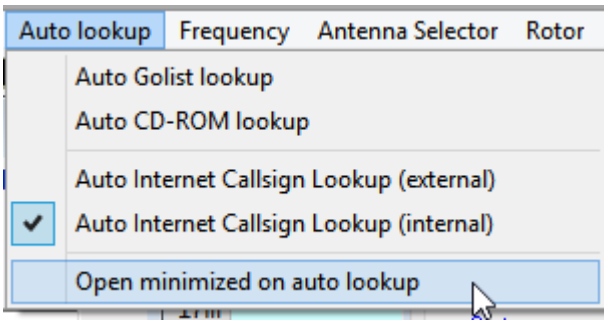
Logger32 can *automatically* search for and display data from the CD-ROM using **Setup ⇨ Auto Lookup ⇨ Auto CD-ROM Lookup** ▶



The callsign presently in the Call field of the log entry pane is looked up when you hit <**Enter**>, <**Tab**> or click the mouse to move the cursor to another field.

Copyright © 2026 Robert C. Furzer K4CY

11.3 Minimized display

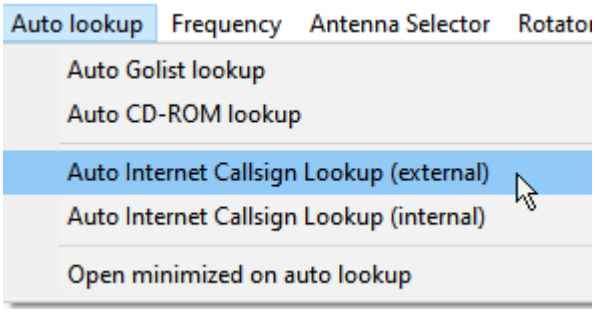


◀ If you are short of screen space, you can run Logger32 with the callsign lookup panes minimized using **Setup ⇨ Auto lookup ⇨ Open minimized on auto lookup**.

The callsign lookups and any automatic information transfers to the log entry pane or logbook still occur in the background, even without the panes being visible. Logger32 can see them, even if you can't.

11.4 Callsign lookup FAQs

Q. Every time I enter a new QSO, the "QRZ lookup" window pops up followed by another "Can't find QRZ lookup module". How do I stop this?



A. Open **Setup ⇨ Auto lookup**, then click to *un*-tick <**Auto Internet Callsign Lookup (external)**> ▶

You *may* prefer the more reliable internal (HamQTH) callsign lookup function instead.

Q. Why doesn't the internal autolookup use QRZ.com? *Everyone* uses QRZ, don't they?

A. No they don't, but anyway the issue is that QRZ gained its popularity by initially providing free online access to their callsign database (containing *our* data!). Some years ago, they changed to a subscription model, limiting free lookups. They also tend to restructure their web pages from time to time without notice, breaking apps that simply 'screen-scrape' the data by parsing the HTML code that paints the callsign lookup results pages. In response to that strategy, Logger32's 'internal' lookup function was changed to use HamQTH (which allows unlimited free lookups, in the true spirit of amateur radio), and an external program interface was developed allowing third party apps to handle other online lookups – including QRZ's paid-for XML lookup service.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why have my QRZ external autolookups stopped working?

A. Assuming you are using L32Lookup by N2AMG, check Rick's site for a new version. Whenever QRZ changes, Rick responds quickly with updates to the app. If you fall behind with the updates, don't be surprised if the utility fails to function as it should.

If you are using some other third-party lookup utility, contact the programmer for assistance.

If you have written one yourself, you're on your own! Good luck!

Q. Why isn't state being transferred to the logbook entry window?

A. Two things worth checking:

1. Your log entry pane's "State" data-entry field may be expecting something other than the ADIF STATE value. The field's displayed name/caption/title/label is irrelevant: check that one of the user fields is configured for 'STATE', which is the official ADIF field name for 'primary administrative subdivision'.
2. On the callsign lookup pane's <Transfer> menu, check that <Transfer selected fields on autolookup> is ticked. Also under <Fields to transfer on click/autolookup>, check that <Transfer primary admin subdivision> is ticked.

"I have switched from Log4OM2 to Logger32 as the XML retrieval from QRZ was very slow & that was affecting my logging speed during Pileup. I have found the Logger32 to be extremely fast for the XML retrieval from QRZ & I can address the person by his name … Also the QO-100 Crossband logger is extremely useful & I have setup 2 profiles, one for Satellite & one for HF."

Ravi S79VU

Copyright © 2026 Robert C. Furzer K4CY

12 Radio control (CAT)

> "Each CAT has a
> distinct purrsonality"
>
> *Richard L. Peterson*

Most modern radios have a CAT (**C**omputer **A**ided **T**ransceiver) capability to send and receive data from computer apps such as Logger32 via a serial (COM) port and cable, often now a USB connection.

This chapter explains how to set up the hardware and software to connect various radios to PCs running Logger32. Section D of the manual provides additional information for connecting and CAT-controlling popular radio brands and models, and covers the **R**adio **C**ontrol **P**anel.

If your radio is CAT-capable and supports the requisite CAT functions/commands, connecting it to your computer unlocks many valuable capabilities within Logger32, such as:

- Displaying, changing and logging the radio's transmit and receive VFO frequencies and modes of operation.
- Controlling and adjusting the radio's:
    - RIT/XIT offsets;
    - Filters and slope tuning;
    - CW/voice memories;
    - **D**igital **S**ignal **P**rocessing and **N**oise **B**lanker settings;
    - Volume and power output;
    - Antenna selector and **A**ntenna **T**uning **U**nit settings;
    - *etc*. … if other facilities/functions are available in your radio.
- Running macros to configure the radio appropriately under various operating circumstances *e.g.* simply click a DX spot for a DX station working split in order to:
    - Move the radio's receiving VFO to the spotted frequency;
    - Select the appropriate radio mode;
    - Narrow the receiver filters to focus on the DX;
    - Shift your transmitting VFO up /down as specified in the spot comment (*e.g*. "UP 1");
    - Enable split;
    - Turn on the sub-receiver to monitor your transmit frequency, perhaps with the filters opened up to listen *around* the pileup for a decent gap;
    - Pop his callsign and other details into the log entry pane ready to log him.

  … All that in the blink of an eye, reliably every time, potentially giving you a head start on working the DX while other DXers are still frantically prodding buttons on their radios or transmitting simplex causing QRM.

Copyright © 2026 Robert C. Furzer K4CY

- Putting the radio into transmit at the appropriate time (*e.g.* while sending CW, RTTY or FT# messages).
- Turning the radio on or off *e.g.* for remote operation via the Internet.

Logger32 also supports **S**oftware **D**efined **R**adios (SDRs) using *virtual* COM ports with the PowerSDR software.

12.1 Configure the CAT's COM port

In order to set up CAT for the radio, first determine which serial communications (COM) port to use for radio control. The COM port itself must be recognized in Windows Device Manager before Logger32 can even see it.

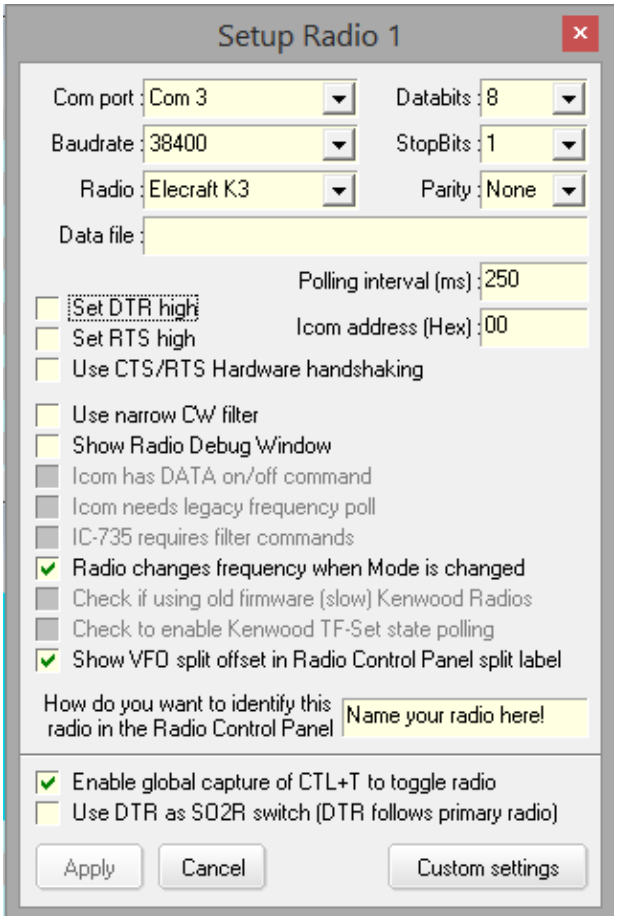
To determine which COM ports are currently available, open the **Windows Control Panel** ⇨ **Device Manager** ⇨ **Ports (COM & LPT)**. Hopefully you will see your CAT interface port or device listed: if not, confirm that **View** ⇨ **Show hidden devices** is ticked.

Logger32 supports the use of [some[174]] USB-RS232 adapters, and multifunction USB radio interfaces such as the *micro*HAM, RigExpert and RigBlaster devices. Some of these have their own software/drivers and configuration functions. Some transceivers have USB ports built-in.

Refer to your radio instruction manual for details of the command protocol, the COM port settings, and the cable and interface requirements[175].

Now in Logger32, configure the COM port to communicate with your radio:

Click **Setup** ⇨ **Radio** ⇨ **Radio 1|2 configuration**[176] ▶



Select your radio type from the <**Radio**> drop-down list. Although Logger32 supports a wide range of transceivers **it cannot reasonably be expected to keep up with every single radio manufacturer, range, model, variant, option and (these days) firmware version**, at least not the very *instant* they are released!

---

[174] Given the choice, choose USB-RS232 adapters based on FTDI chipsets that are qualified to operate with your version of Windows. Due to cheap, low-quality fakes flooding the market, Prolific devices, in particular, are unreliable and best avoided. More info.

[175] More detailed instructions for specific radios may be found in section D of this User Manual.

[176] Logger32 supports up to two radios on two CAT ports, routing CAT commands to the appropriate port and radio depending on what you are doing. See the SO2R chapter for more.

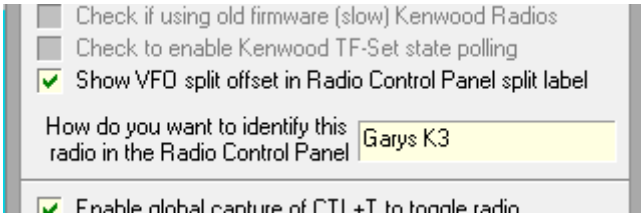
Copyright © 2026 Robert C. Furzer K4CY

If your particular radio model is *not* listed, you will generally find that a similar listed model works OK since the settings and command sets are often similar, the basics at least. Novel radio features or CAT commands may not be supported however[177].

Accept the default values for the remaining serial port parameters or change them as required, according to the specific requirements of your radio.

Ensure that the radio is physically connected to the configured COM port and the port is opened in **Setup ⇨ Radio ⇨ Open port**.

Hinson tip: notice the field towards the bottom of the radio configuration form, asking <**How do you want to identify this radio in the Radio Control Panel>** ▶



The text you enter here is, in fact, used in several menus and other places in Logger32 as a more obvious label than just "Radio 1"[178] and "Radio 2" *e.g.* a label for the radio in CAT port and on the spectrum display in the Radio Control Panel. A sensible name helps cope with the complexities of an SO2R setup. Use the radio model, maybe, its position on the shack desk ("Lefty" & "Righty"?), its age ("New", "Old", "Shiny", "Virtual", "Ancient", "Boat-anchor") or some other short but distinctive name, like "Bob".

The default values of 9600 baud for the COM port speed and 1000 ms (1 second) for the polling interval are merely suggestions. Use the highest baud rate that provides reliable operation with your specific hardware. Reducing the polling interval (increasing the rate of radio polling) will provide a faster and smoother display of band, frequency and mode changes.

Hinson tip: COM port numbers can be changed through Windows Device Manager: in the **Port Settings** tab, click <**Advanced**> then select a COM Port Number using the drop-down list. However, it will change automatically if a given device is plugged into a different USB socket connected internally to a different USB hub ... so try to stick with the usual USB sockets to avoid confusion. It is also possible to change the "Friendly name" for each COM port by updating the Windows registry – see N6TV's detailed instructions – but if the COM port gets renumbered, the "Friendly name" is lost, so get the COM port numbering right first.

It has taken several weeks to get my new radio to work with L32 & JTDX. The fault (after a lot of finger trouble) turned out to be the USB cable itself. I now have a new double-screened cable with a ferrite at each end. It all works with no more random crashes & lock-ups that have been plaguing me. I urge anyone with a new radio to get a good USB cable to go with it.

Chris G4ZCS

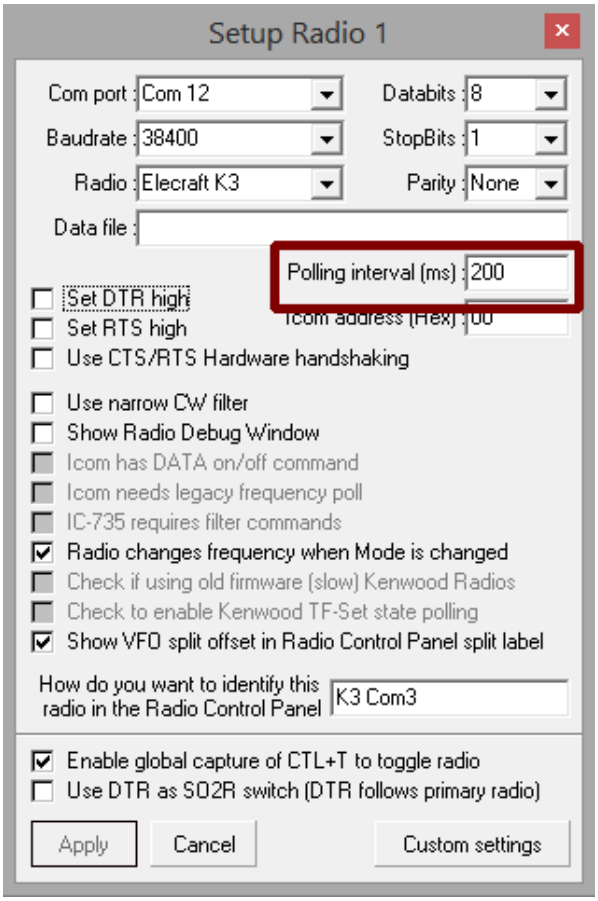
[177] By all means seek support/advice on the Logger32 reflector but please be considerate: you are not the only Logger32 user, and the developer and support crew are always busy with a million and one things (such as enhancing Logger32, testing changes and maintaining this very manual).

[178] Attentive readers may have noticed that in some screenshots, my Radio 1 is called "K3 Com3" although it is configured on Com12: the reason is that I was using LP_Bridge to share the K3's actual serial connection (on Com3, a USB-RS232 adapter) between several programs (Logger32, N1MM+ and JTDX). Com12 was the virtual port assigned to Logger32 in LP_Bridge. Thank you for asking.

Copyright © 2026 Robert C. Furzer K4CY

12.1.1 Polling the radio

Logger32 repeatedly polls (sends queries to) the connected radio for information such as its VFO frequency and mode when the CAT port is open.



The millisecond[179] interval between polls is configured from the radio configuration ▶

Polling is automatically suppressed during transmit unless <**PTT by Radio Control**> is enabled in **Sound Card ⇨ Settings ⇨ Radio PTT Options**.

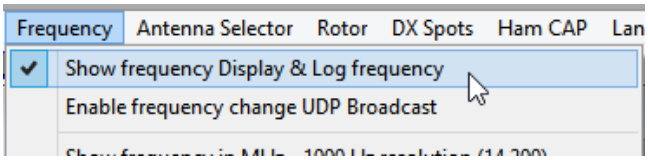
Hinson tip: if you occasionally see an error message complaining about a random frequency not being in your bandplan, try increasing the polling interval by, say, 50 to 100 ms, giving the system a better chance of keeping up. Alternatively, if the error message only appears while you are transmitting, you may have some RF feedback.

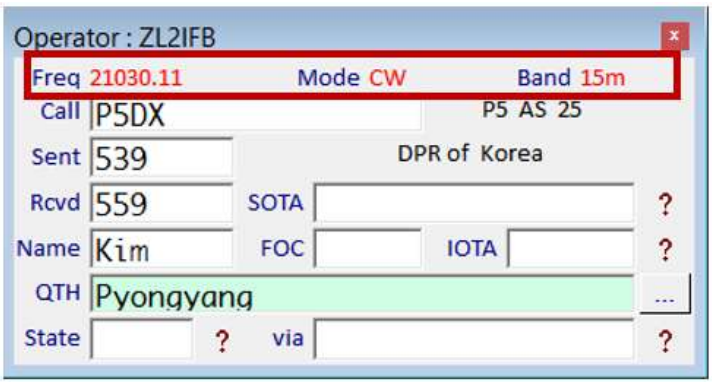
12.1.2 Logger32 bandplan

Not all radios respond with their mode when polled via CAT. Also, when attempting to QSY a radio to a DX spot some guesswork is required in determining the mode to put the radio on *e.g.* is a DX station spotted on 14080 kHz using CW, FT4, RTTY or some other mode? To address this difficulty, see the Bands & Modes table section.

12.1.3 Display frequency from radio

Open **Setup ⇨ Frequency ⇨ Show frequency Display & Log frequency** ▶





◀ The top line of the log entry pane shows the radio's transmit VFO frequency, mode and band (calculated from the frequency), all of which will be recorded in the log (along with the data in the white fields beneath) when you complete and log this QSO.

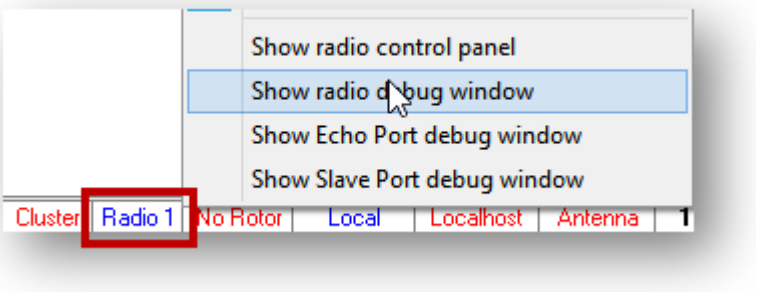
[179] 100 milliseconds is but a brief blink of an eye. 400 ms is a lazy blink. 1,000 ms is a wink.

Copyright © 2026 Robert C. Furzer K4CY

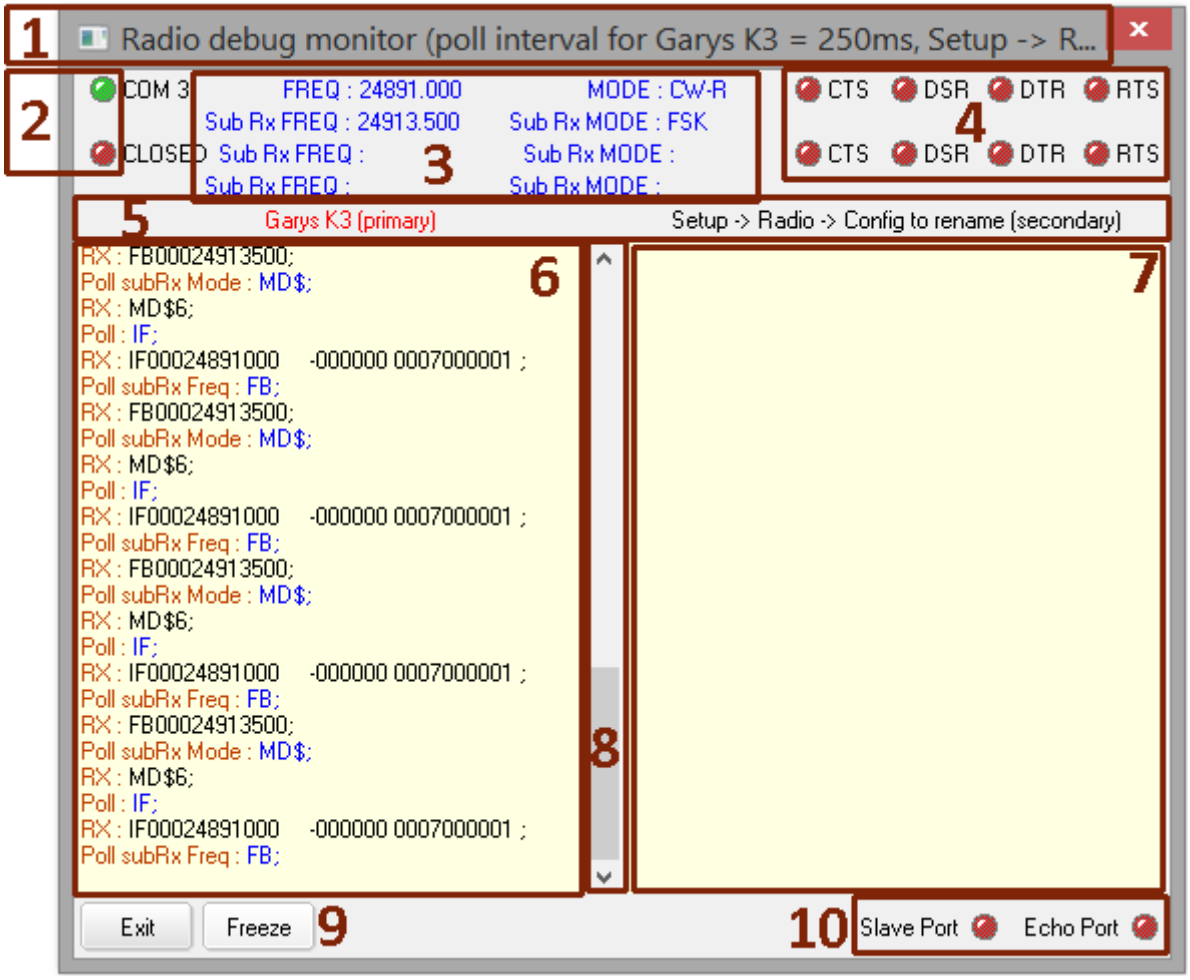
12.2 Radio debug window

If things aren't going to plan, or if you are interested to see what's going on, the radio debugger is a diagnostic instrument with which to troubleshoot the serial CAT communications between Logger32 and your CAT-connected radio/s.

Open the debugger using **Setup ⇨ Radio ⇨ Show radio debug window**, or right-click the <**Radio**> panel on the status bar, then click <**Show radio debug window**> ▶



The debugger shows the data flowing each way between Logger32 and the radio's CAT port ▼



The debugger shows:

1. The CAT port polling rates (truncated in this example due to the long name for Radio 2).
2. "LEDs" indicating whether the CAT ports are open (green) or closed (red). Radio 1's CAT port is above, with Radio 2 beneath.
3. The frequencies and modes reported by each radio for VFOs A (main) and B (sub).

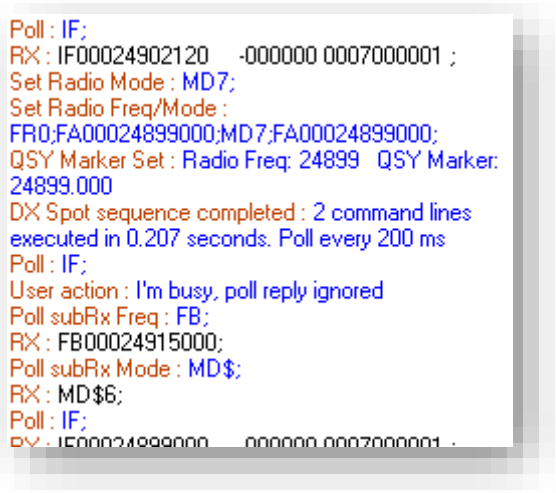
Copyright © 2026 Robert C. Furzer K4CY

4. Two sets of 4 “LEDs” showing the logic levels on the modem control lines for each CAT port: high (green) or low (red).
5. The names you have given Radio 1 and Radio 2 appear across here. You can change them using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**.
6. Displays the rapidly-scrolling sequence of CAT communications flowing to and from Radio 1. Lines beginning “Poll” are Logger32 sending queries or commands to the radio. The blue text is the query or command string sent (*e.g.* “IF;” is the Elecraft CAT command for “What is your VFO A frequency?”). Lines beginning “RX” are messages Logger32 received back from the radio. The black text shows what the radio sent - data responses to the commands and queries sent. The format depends on the radio manufacturer and model.
7. This is where the CAT comms for Radio 2 would appear, if you have 2 CAT radios in use.
8. You can click and drag the scroll slider up the screen to stop the scrolling temporarily while you look back at recent messages …
9. … or click <**Freeze**> to stop the screen updates completely and scroll at your leisure.
10. Two “LEDs” tell you whether the Slave and Echo Ports are open (green) or closed (red).

The debug messages flow rapidly, making it hard to read, let alone understand what is going on, so it may help to:

- Stretch the debug window to the full height of your screen, maximizing the visible text area.
- After sending a command, hit the **<Freeze>** button to stop the action, giving your eye and brain a chance to catch up! With <**Freeze**> on, you can scroll back for a look at something you missed, and it is easier to drag the mouse cursor over any text you wish to copy into the Windows clipboard with a <**Ctrl**+**C**>, ready to paste with <**Ctrl**+**V**> into an email to the Logger32 reflector *etc*.

This is what I saw moments after typing 24899 <**Enter**> in the log entry pane to QSY my K3 to 24899 kHz ▶



12.3 CAT control of radios (commands)

The $**Command**$ and $**HexCommand**$ macros can send any legitimate command to a CAT-connected radio. Each radio manufacturer maintains their own set of commands, often changing when new/updated models or firmware is released … but luckily there are only a few ‘control families’, some of which are shared by different manufacturers (*e.g.* Kenwood and Elecraft).

12.3.1 DX spots

Clicking a DX spot in the DX Spots pane, the DX cluster pane or the BandMaps immediately sets the CAT-connected radio to the spotted frequency and the mode specified for that frequency in

Copyright © 2026 Robert C. Furzer K4CY

the Bands & Modes table, or the mode defined in the spot comment. At the same instant, the log entry pane is populated with the callsign, frequency and mode, ready to the QSO (if you make it: that part is not guaranteed, unfortunately!).

- **Post a DX spot**: with the focus on the log entry pane, pressing <**Ctrl**+**D**> opens the DX Spots pane ready to post a spot *to* DX cluster, either for the callsign currently in the Call field, or the callsign most recently logged if there is none currently being logged. The transmit VFO frequency is sent automatically with the spot, and you may want to add a comment (such as *Long path via Antarctica*) to add a little value and interest to your fellow DXers.
- **Reset frequency and mode**: right-click the DX Spots pane then click <**Reset Radio 1|2 frequency – nnnnn.nn**> to return the radio to its previous frequency and mode, as it was before you last clicked on a DX spot.

12.4 Controlling the radio's Push To Talk

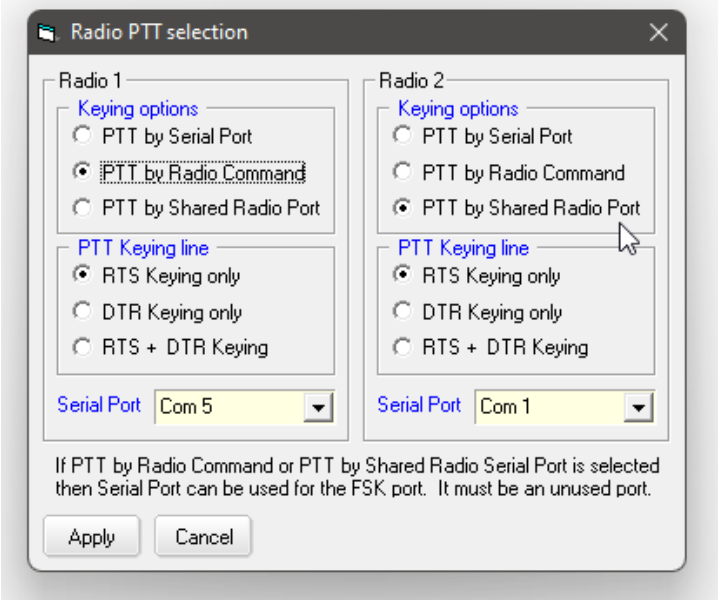
There are four different ways to put the radio into transmit to send a message via RTTY, PSK, FT# *etc*., then return it to receive. The first two are controlled automatically by Logger32, whereas the remaining two rely on the radio or the operator. The following four subsections cover them all.

12.4.1 PTT control by CAT

Many modern radios[180] can be switched between transmit and receive by sending the appropriate CAT commands. Logger32 can be configured to send the appropriate CAT commands to the radio, for instance commanding the radio into transmit before sending a CQ call on FT#, returning to receive after the message is sent.

Hinson tip: CAT is the simplest, preferred method for controlling PTT *if* there is a CAT connection and the CAT commands for 'transmit' and 'receive' are supported by your radio.

To configure Logger32 for PTT by CAT, from the Sound card data window, open **Settings ➪ Radio PTT Options ▶**



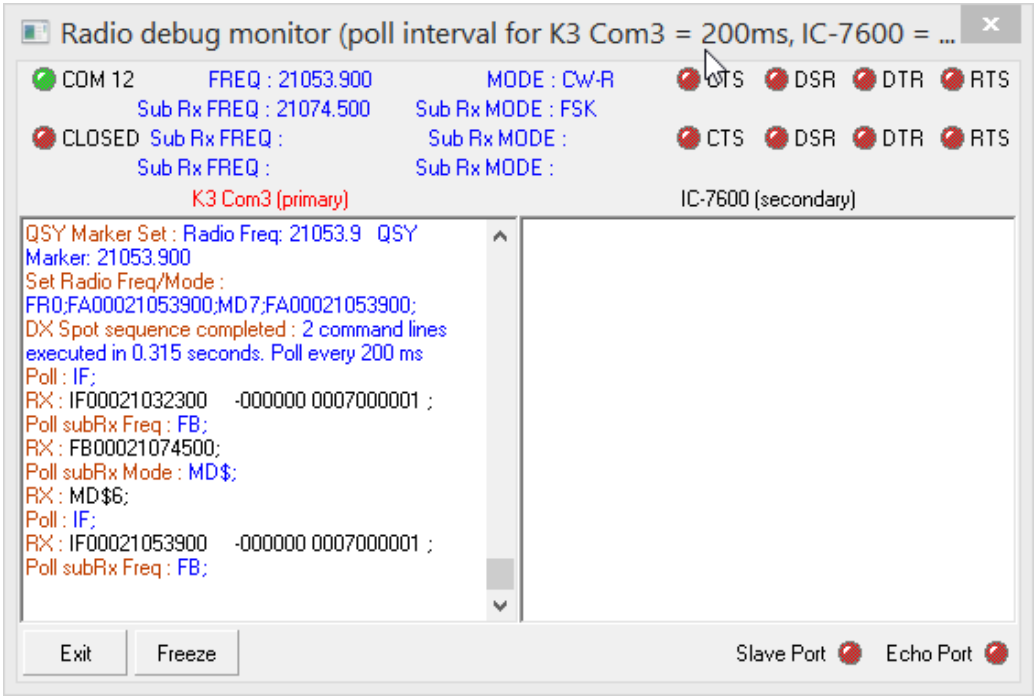
Shown here on the left side (Radio 1), PTT by Radio Command (*i.e*. CAT) is selected using serial port Com5.

The $**Transmit**$ and $**Receive**$ macros allow us to program the corresponding PTT commands into CAT sequences sent to the radio.

[180] TEN-TEC uses the ICOM command set, supplemented with an TX/RX changeover command that ICOMs lack. It is still worth configuring the CAT interface to exchange mode, frequency, filter and other information between the computer and ICOM, but you may need to implement PTT differently.

Copyright © 2026 Robert C. Furzer K4CY

We can monitor the CAT commands and data flowing back and forth between Logger32 and the radio using **Setup ⇨ Radio ⇨ Show Radio debug window**[181] ▼



You may need to study the radio manual to figure out what the commands and responses mean, but the mere fact that the CAT port is open (hence the green "LED" for COM 12 above), there are messages scrolling through the left pane, and the radio is reporting its VFO frequencies, indicate that the CAT connection is working.

If PTT is not being controlled reliably, try changing the polling rate in **Setup ⇨ Radio ⇨ Radio 1|2 configuration**. The polling rate may be too fast for old/slow radios to respond, so try increasing the polling interval *e.g.* to 500 or 1,000 ms. If that setting works OK, you can reduce it towards the point it starts to fail, then increase it a little, leaving a bit of slack for greater reliability.

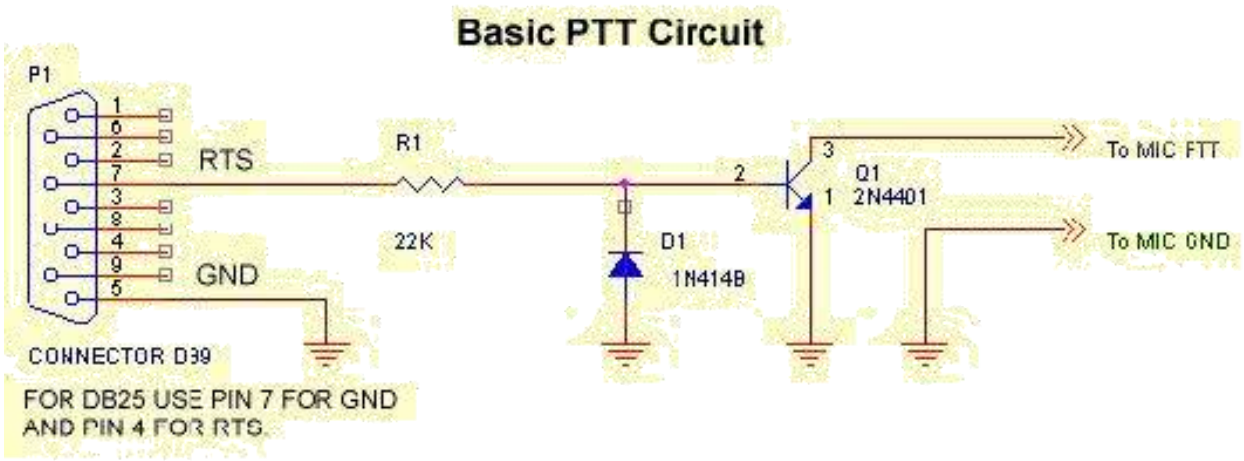
Hinson tip: once you command the radio *into* transmit using CAT or PTT via RTS or DTR, if the computer locks up or crashes (perhaps due to RF feedback), you may have difficulty taking the radio *out* of transmit. So, plan ahead. Before you find yourself in this awkward situation, think about your options. Rebooting the PC will probably work but takes valuable time – not good in a sked, net, pileup or contest. Turning the radio off and back on will generally clear all CAT commands quite quickly … but will not cancel a stuck RTS/DTR PTT or CW keying line, or a naughty sound card transmitting a continuous tone. If you are operating a remote station, how can you regain control of the PC *and* the radio if something goes wrong? Can you do it quickly and reliably enough to avoid annoying other radio users and perhaps losing your license and/or your finals?

[181] You can also open the radio debug window after right-clicking the **Radio** panel on the status bar.

Copyright © 2026 Robert C. Furzer K4CY

12.4.2 PTT control using RTS or DTR lines on a serial port

RTS (**R**eady **T**o **S**end) and **DTR** (**D**ata **T**erminal **R**eady) were originally used to control the flow of data to modems – particularly the early ones that with very slow data rates and limited cache memories. Before sending data to a serial port, the PC would assert RTS and wait for a response from the modem. The modem would finish whatever else it was doing (*e.g*. converting digital data received previously into audio tones and sending them down the telephone line, either on the same port also being used for CAT control of the radio, or on a separate serial port. To key the transmitter, Logger32 asserts (raises the voltage on) the serial port's RTS and/or DTR line (depending on how the port is configured) meaning pins 4 or 20 of a DB-25 connector, or pins 7 or 4 of a DB-9. However, your radio almost certainly needs its PTT line to be *grounded* for transmit. A simple single-transistor interface (an electronic switch) is all you need, something like this ▼



When the transistor base (#2) voltage is raised by the PC, current flows between the collector (#3) and emitter (#1), grounding the radio's PTT line. The components are non-critical: almost any general-purpose NPN transistor and diode should work. Other equally simple PC-radio interface circuits are shown in the CW Machine chapter, since keying the radio's PTT or Morse key inputs through a PC serial port is essentially the same job.

12.4.3 PTT control by VOX

An easy way to control transmit/receive is to use VOX in the radio (if it has a VOX function). Simply enable VOX on the radio. Now, when sufficient audio arrives from the PC sound card or digital comms interface device, the radio should automatically go to transmit, returning to receive when the audio stream ends.

Having set the correct audio drive level, you *may* need to adjust the radio's VOX gain, anti-VOX and hang/delay times to trigger reliably. Set the VOX sensitivity on the radio such that the normal audio output from the computer sound card reliably triggers the radio into transmit, and it drops back to receive soon after the audio ends.

Copyright © 2026 Robert C. Furzer K4CY

You may find that your radio transmits unexpectedly, *between* overs: this is often a sign that your PC is generating audio alerts which are being sent to the radio, triggering the VOX. Fix this by opening the Windows Volume Mixer (normally by right-clicking the loudspeaker icon on the Windows task bar), selecting the sound card dedicated to your radio if (as recommended) you have more than one, and **muting all audio sources other than the digimode software** (*e.g.* MMVARI or JTDX)[182].

Hinson tip: keep an eye on the rig's ALC meter and monitor your own transmit audio from time to time, even on digimodes, both to check the levels and to confirm that the only sounds you are transmitting are your generated digimode audio. If you listen on any busy FT# frequency, you may notice bursts of noise, bleeps and chimes being transmitted inadvertently by inattentive/inept hams and occasionally audio alerts such as "*New country!*". Doh!

12.4.4 Manual PTT control

The most basic method is for you to operate your radio's PTT manually *e.g.* by pressing a XMIT/MOX button on the front panel, clutching the microphone switch or kicking a PTT footswitch, then sending the message, then releasing the control.

Manual PTT may be handy for initial setup and testing purposes but is tiresome and soon becomes a nuisance if used routinely. It requires that you:

a) Operate the rig's PTT with a switch; *and*

b) Tell the digimode software when you want to transmit or receive.

Variable timing can mess up the transmission and reception of FT# and other time-dependent digimodes. You may forget to key the transmitter although you have already started transmitting audio out of the computer, and sometimes you will notice that the digimode software's receive waterfall remains blank … because your radio is still in transmit. However, it may be the only viable option to run digimodes on a lovingly-maintained or restored boat-anchor sideband transceiver. Good luck with that!

12.5 CAT echo and slave ports

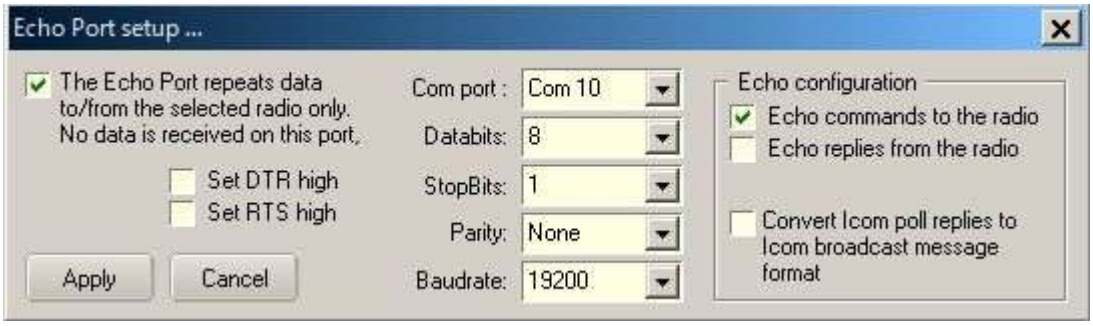
An **echo port** can be configured to replicate CAT data sent to the active radio, sending it also to some other device (an amplifier, antenna controller or whatever). Logger32 simply copies the CAT data to a serial port to which the other device is connected by CAT: the other device therefore needs to understand the same CAT protocol as the active radio. Echo is a *uni*directional (broadcast) function. Any CAT data or responses generated by the other device are callously discarded.

A **slave port** can be configured to synchronize a slave receiver/transceiver (of any type) with the main radio's frequency and (optionally) its mode. If <**Enable polling of the slave receiver/transceiver....**> is enabled, it works the other way too *i.e.* the main radio also follows the slave radio's frequency and mode. In contrast to the echo function, this implies *bi*directional CAT communications with both radios.

[182] The Volume Mixer only shows sources that have recently generated audio on the selected device and still have an audio session open. Sources that only intermittently generate (such as "You've got mail!") may not be shown when you look, so check again later. Also, Windows 10 updates have an annoying tendency to reset various audio defaults, making the Hinson tip pertinent. Don't be a clueless bleeper-blooper!

Copyright © 2026 Robert C. Furzer K4CY

12.5.1 CAT echo port setup



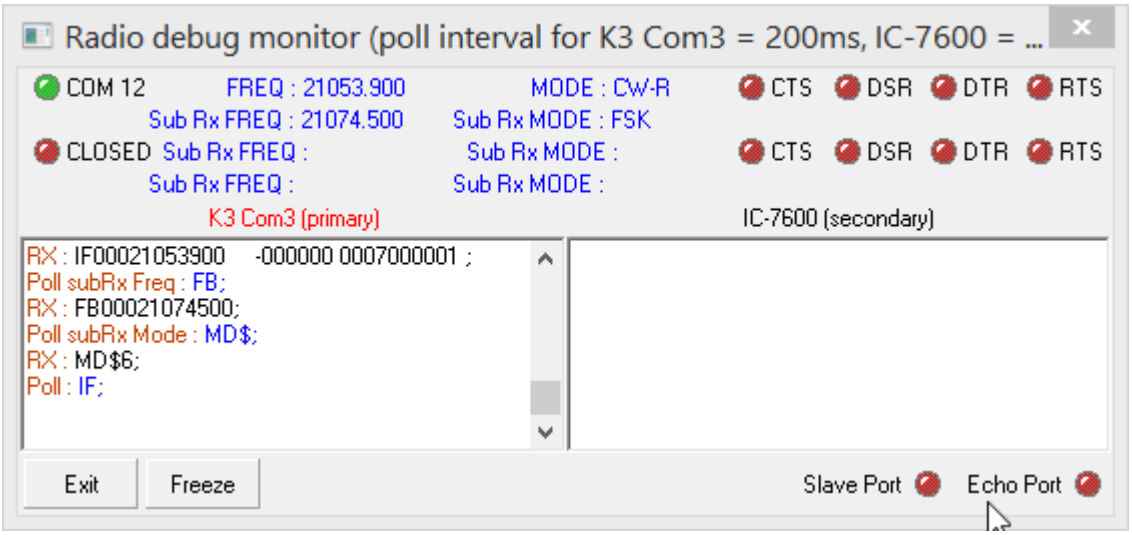
Open **Setup ⇨ Radio ⇨ Echo port** ▲ to configure the CAT echo port:

- The COM port parameters should be configured as required for the connected device (check the manual, and identify which port is connected using Windows Device Manager).
- If the connected device requires hardware handshaking, you may need to <**Set DTR|RTS high**>.
- Choose whether to echo *just* the CAT commands sent by Logger32 to the radio, or those *plus* the radio's replies.
- If this is a 'transmit only' connection, select <**The Echo Port repeats data to/from the selected radio only. No data is received on this port,**>
- Click <**Apply**> to save and activate the settings.



Open or close the echo port by right-clicking the radio pane on the status bar ▶ or by clicking **Setup ⇨ Radio ⇨ Open|Close port – Echo**

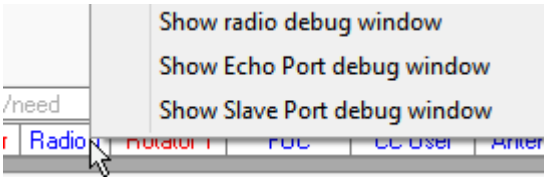
The echo port's current state is shown by the color of an "LED" at the bottom right of the radio debug window ▼

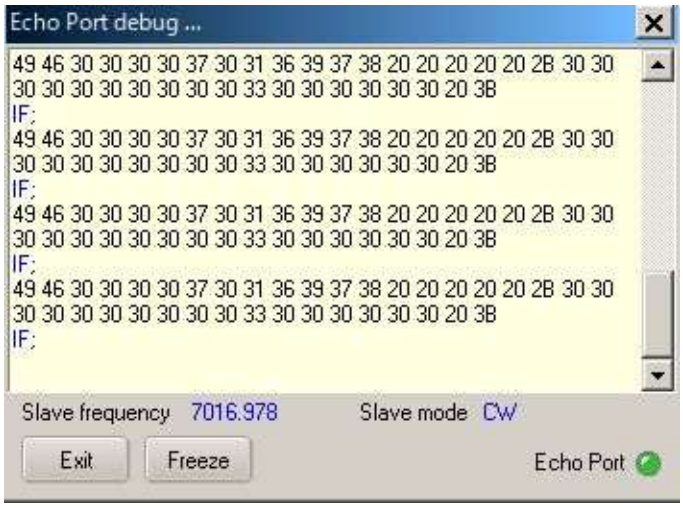


Green means the port is open for business, red means it is closed.

Copyright © 2026 Robert C. Furzer K4CY

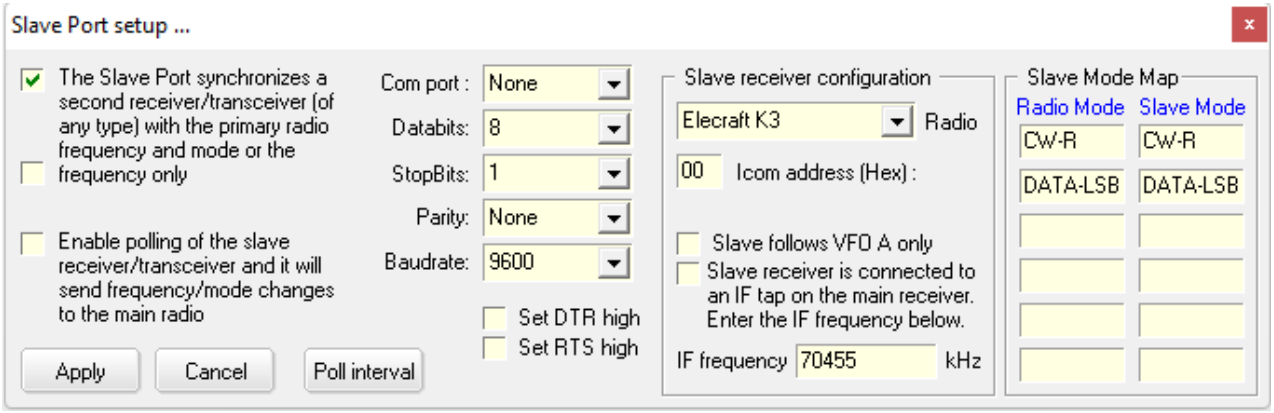
You can watch the flowing data communications ▶ by right-clicking the **Radio** panel on the status bar then clicking <**Show Echo Port debug window**> ▼





12.5.2 CAT slave port setup

To configure the slave port, open **Setup ⇨ Radio ⇨ Slave port configuration** ▼

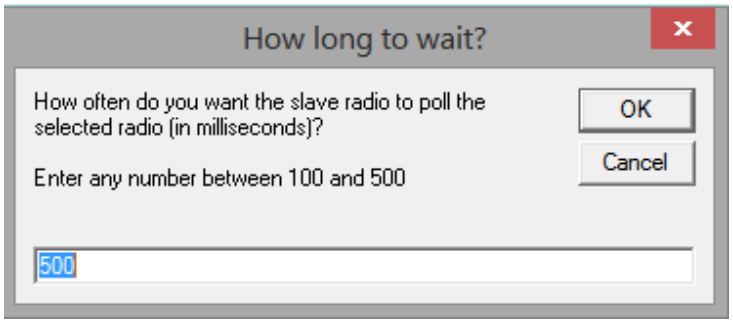


The upper and lower checkboxes on the left are self-explanatory. The middle 'frequency only' checkbox stops Logger32 passing mode commands to the slave – a workaround for slaved transceivers that handle the mode commands differently to the main transceiver (*e.g.* if they don't shift the VFO frequency to compensate for mode offsets, but the main transceiver does).

If your main transceiver reports both its VFO A *and* VFO B frequency/mode via CAT but the slave can't handle both (*e.g.* old ICOMs), un-tick <**Slave follows VFO A only**>.

The <**Poll interval**> defaults to 500 ms but can be reduced to improve responsiveness … *provided* your PC and slave radio can keep up ▶

So, 'suck it and see'.



<**Slave receiver is connected to an IF tap on the main receiver**> sets the SDR to the IF frequency and does not send any more frequency commands – in fact it stops polling the slave radio.

Depending on the sequence in which you start Logger32 and the SDR, there can be difficulty synchronizing the SDR to the IF frequency. If you start Logger32 and open the slave port, the command to set the slave to the IF frequency is sent automatically. If you then turn on the SDR, it

Copyright © 2026 Robert C. Furzer K4CY

missed the IF frequency command so it just sits there with no eye-candy. You can force the slave to resynchronize either by closing and re-opening the slave port, or by executing an RCP macro $**SlaveSync**$.



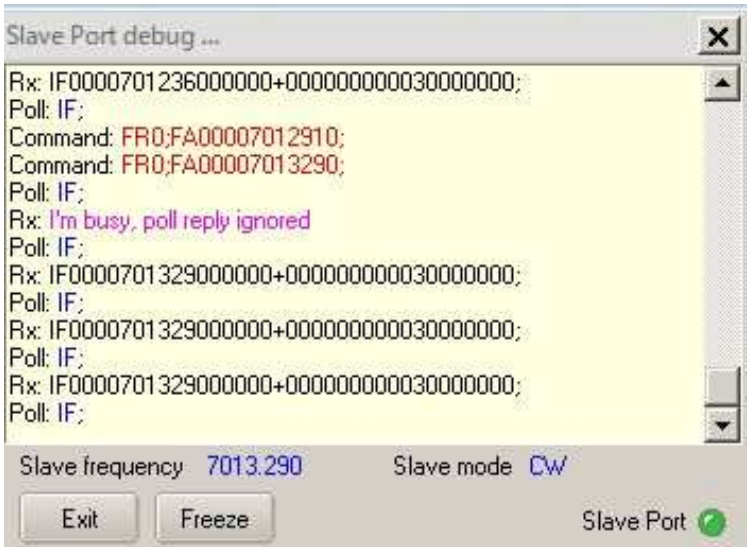
To open or close a slave port, right-click the <**Radio**> panel on the status bar, then click <**Open|Close port- Slave**> ▶

or use **Setup ⇨ Radio ⇨ Open|Close port – Slave**

The slave port state is shown by its "LED" at the bottom of the radio debug window ▼



To confirm it is properly configured, right-click the **Radio** pane on the status panel, then click <**Show Slave Port debug window**> and watch the CAT data flowing nicely, hopefully ▶



Copyright © 2026 Robert C. Furzer K4CY

12.6 Interfacing examples

For the examples, a K3 is connected to Logger32 using COM2 as the main radio, and a TS-590SG is connected to the echo port on COM1 as a slave radio.

The TS-590SG is controlled by echo replies from the main radio (the K3 in this case).

Select <**Echo commands to the radio**> in the echo port setup form.

The following command macros in the **R**adio **C**ontrol **P**anel can control the TS-590SG via the echo port:

- $**Command FA;**$ reads the K3's current VFO A frequency and sends the TS-590SG to the same frequency.
- $**Command MD;**$ reads the K3's current mode and sends the TS-590SG to the same mode.

12.6.1 TS-590SG controlled by echo command to the radio

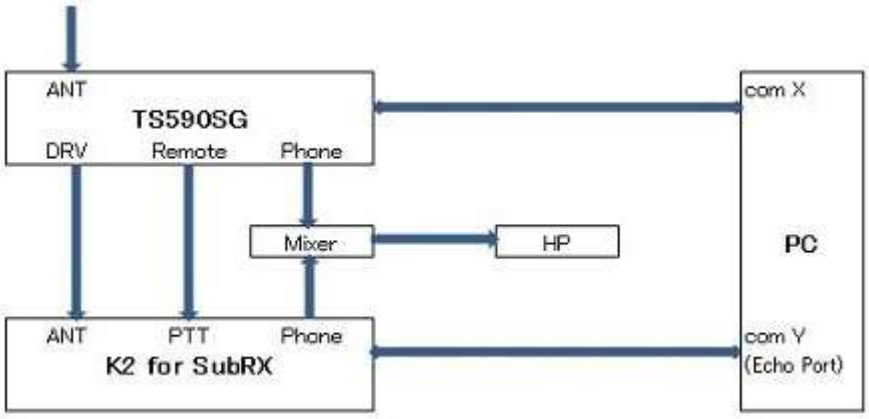
Select <**Echo commands to the radio**> in the echo port setup form.

The following CAT command (typically defined on one of the function buttons in the **R**adio **C**ontrol **P**anel) sets *both* the main K3 and the slave TS590SG to 14040 kHz CW:

$**Command FA00014040000;MD3;**$

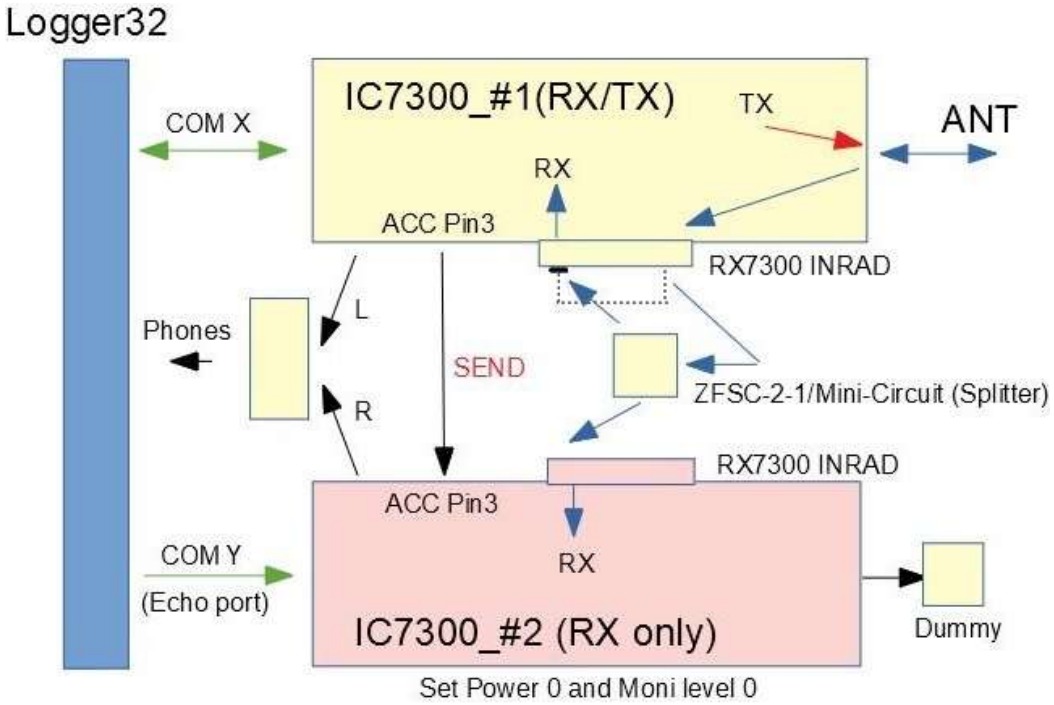
12.6.2 K2 controlled through echo port as second receiver of TS-590SG

When you click a DX spot, both radios tune to the spotted frequency. When you turn the TS-590SG's VFO knob, you need to send the CAT command **FA;** to tell Logger32 to tune the K2 to same frequency as the TS-590SG.



Copyright © 2026 Robert C. Furzer K4CY

12.6.3 IC-7300 as a second receiver for … another IC-7300



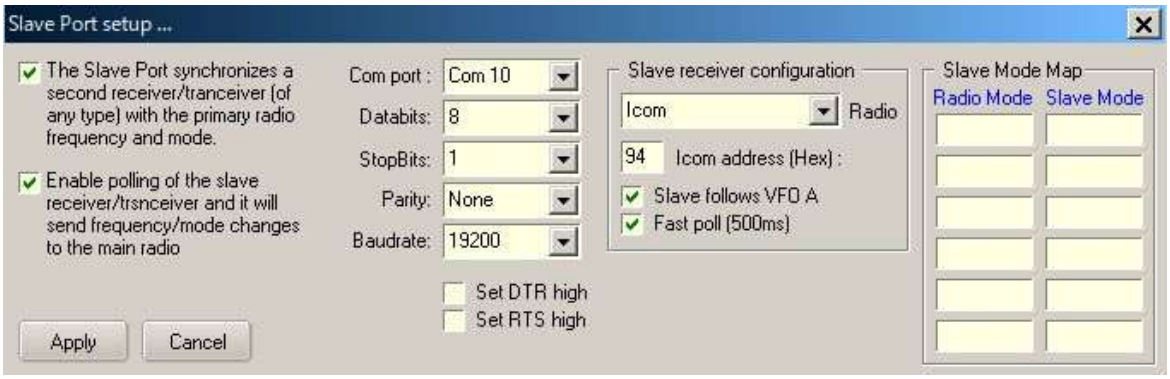
▲ IC-7300 #2 connected to an echo port automatically follows the frequency and mode of IC-7300 #1 when you click a DX spot, type a frequency into the Call field of the log entry pane, or trigger a macro command to change frequency or mode. To call a DX station working split, click a DX spot or type his frequency. Both #1 and #2 IC-7300 are tuned to the spotted frequency. While you listen to him on IC-7300 #2, tune around his pileup on IC-7300 #1 to find a clear frequency on which to call him. Transmit on IC-7300 #1 when appropriate.

12.7 Command a slave receiver or transceiver

Examples of slave port settings for various combination of main/slave radios follow.

12.7.1 Slaving a pair of IC-7300s together

In this example, the slave IC-7300 is connected to COM10 ▼



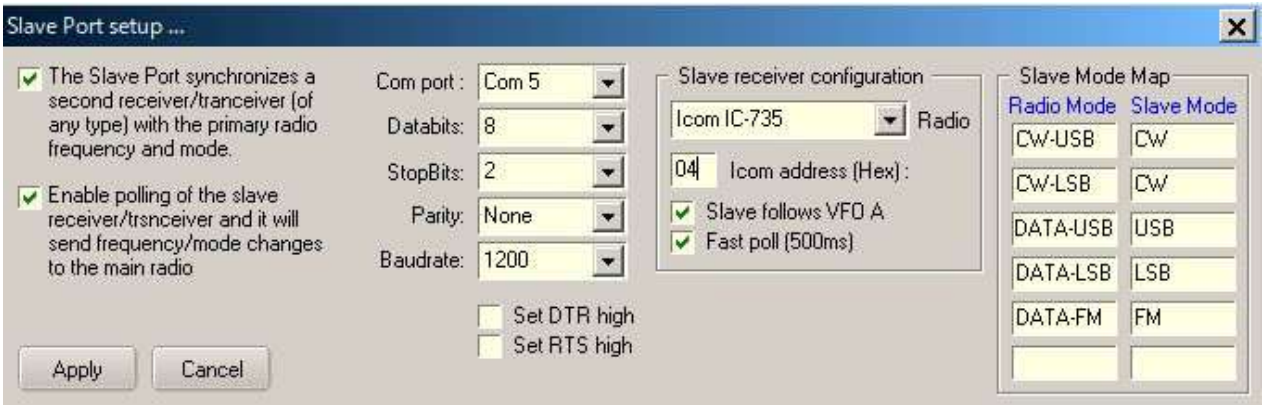
Normally, the slave follows the main radio's frequency and mode when clicking DX spots, changing frequency in the log entry pane or sending direct command macros. Both radios move together like synchronized swimmers, only without as much bobbing.

Copyright © 2026 Robert C. Furzer K4CY

Likewise, the main radio follows the slave's frequency and mode as you tune the slave around using its VFO knob and buttons.

12.7.2 FT-920 as main radio and IC-735 as slave radio

In this case, the slave IC-735 is connected to COM5 ▼

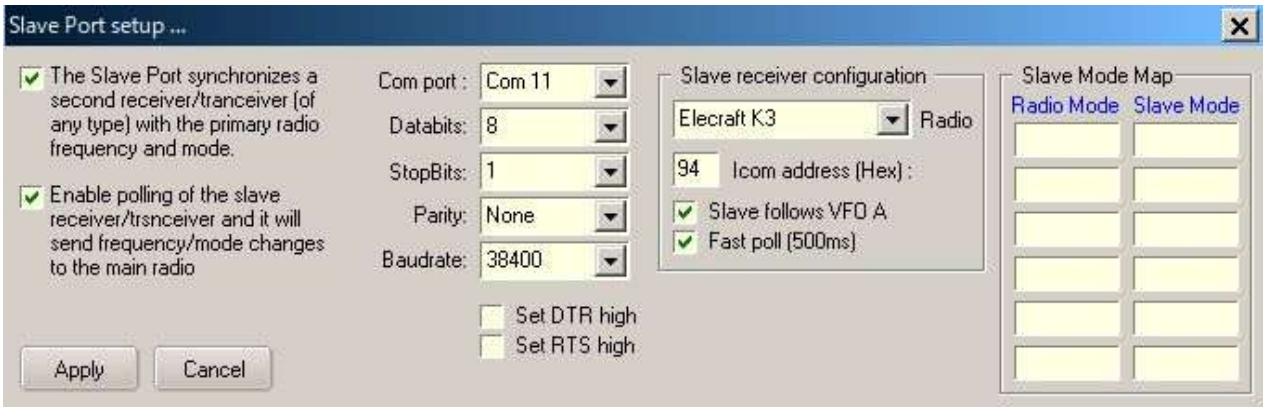


The righthand side of the slave port setup screen shows how radio modes are converted between the main and slave radios. The slave baud rate is shown as 1200 (factory default). The IC-735 can be set to 9600 baud by changing an internal jumper.

Hinson tip: I don't know what the change entails. Maybe the jumper becomes a cardigan?

12.7.3 IC-7300 as main radio and K3 as slave radio

Here, the slave K3 is connected to COM11 ▼



12.8 Use a cheap SDR RX as a panadapter

A cheap SDR RX and app, controlled via a slave port, makes a handy panadapter/band-scope.

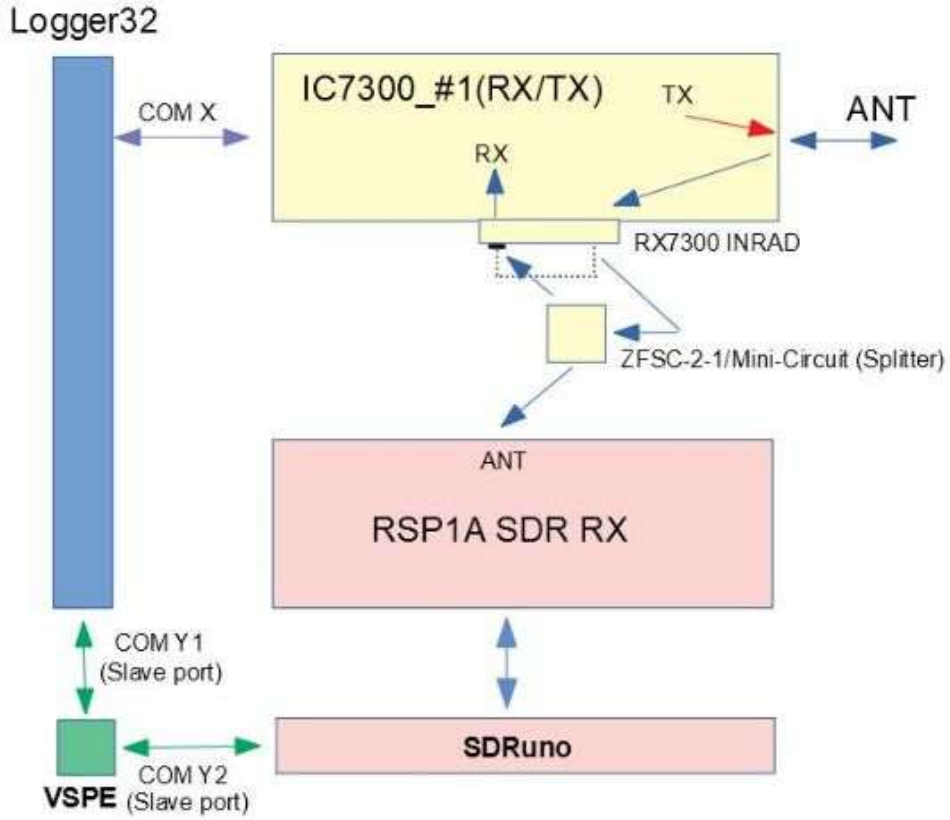
We used SDRplay RSP1A as SDR RX, SDRplay SDRuno V1.22 and SDR-Radio SDR Console V3.B2 as the SDR app for testing. These apps can be downloaded from www.sdrplay.com/downloads/

IC-7300 and IC-706MK2G do not report both VFO A and VFO B frequency/mode in real time, however Logger32 commands RSp1A/SDRuno (or SDR Console) frequency and mode when:

- Clicking a DX spot.
- Entering a frequency in the log entry pane or changing frequency by mouse wheel.

Copyright © 2026 Robert C. Furzer K4CY

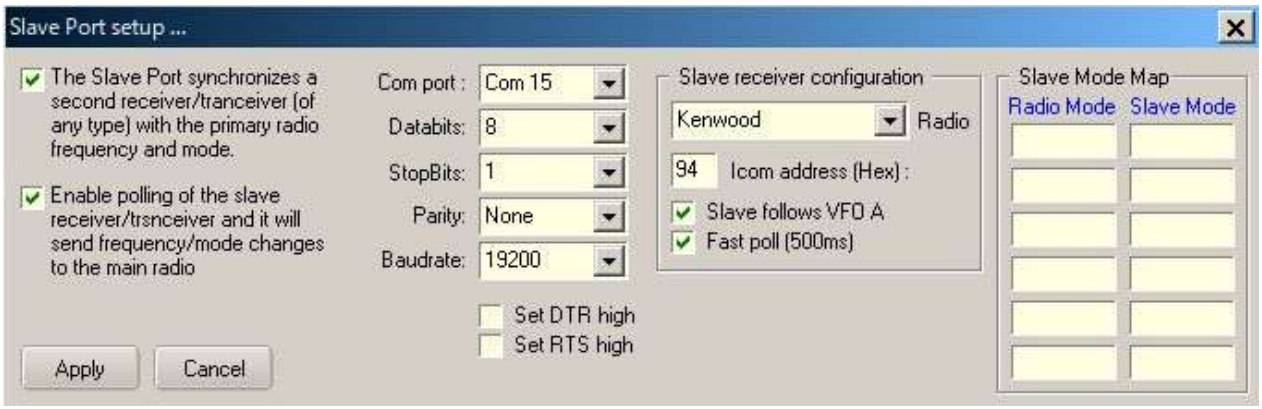
- Turning the IC-7300 VFO knob (VFO A or VFO B).
- Changing frequency and mode in SDRuno (or SDR Console).



◀ Basic setup

Logger32's slave port does not command VFO B in SDRuno. Even if you select VFO B in SDRuno, it returns to VFO A when you change frequency or mode on the IC-7300.

Using VSPE, create a pair of ports to share a slave port, one for Logger32 and another for SDRuno. COM 15 and COM 16 are used in this example ▼



Use your particular COM port and settings, obviously, and select <Kenwood> as the slave receiver radio. Tick both <**The Slave Port synchronizes a secondary receiver …**> and <**Enable polling of the slave receiver …**>.

Copyright © 2026 Robert C. Furzer K4CY

SDRuno works with the default settings provided you select the correct COM port for your device. In this case, it is COM 16. For details, see the SDRuno User Manual.



▼ Typical SDRuno displays ▶



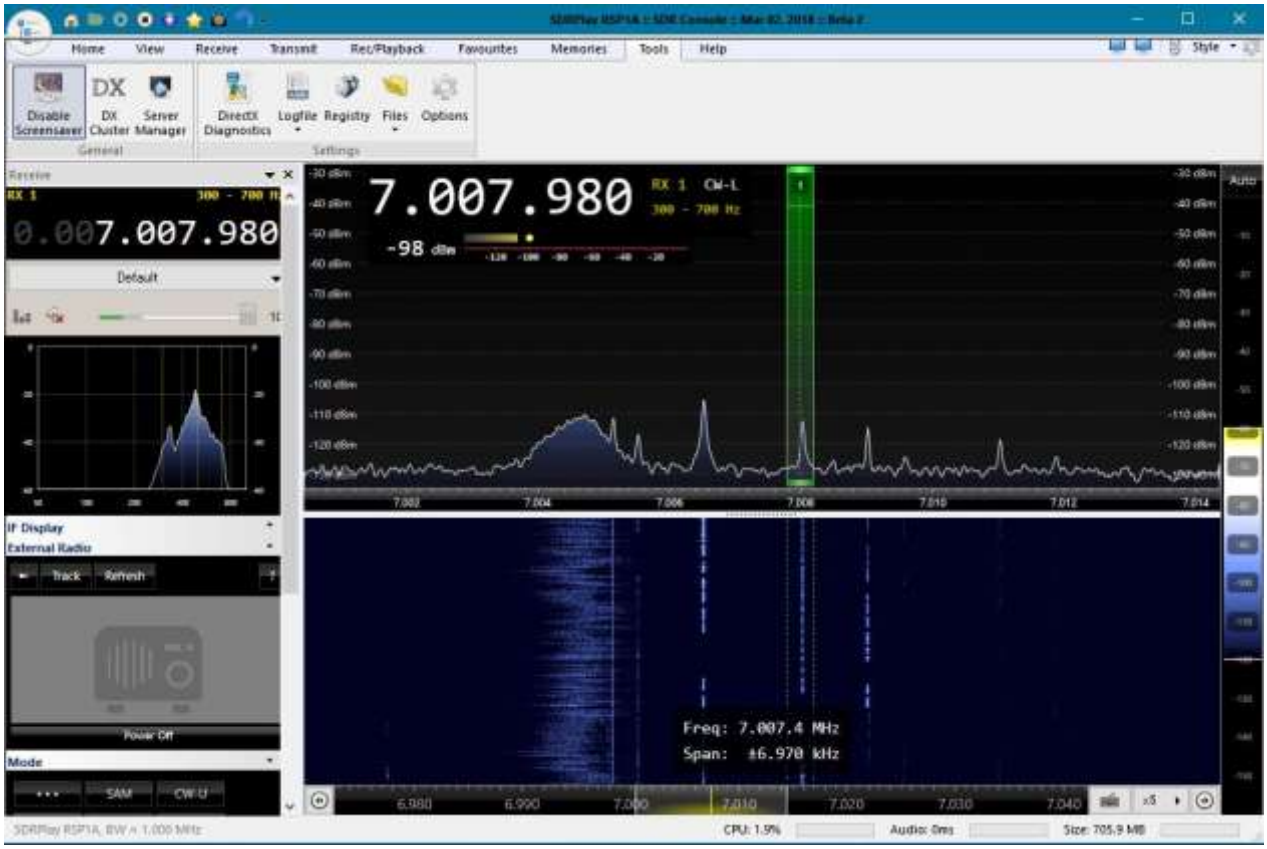
Copyright © 2026 Robert C. Furzer K4CY

12.8.1 SDR Console setup

1. Open the SDR Console program and click **Tools** ⇨ **Options**.
2. Click the **Controllers** folder, then the **CAT(Serial Port)** folder.
3. Scroll the window to find the other half of the Virtual serial port pair that you created earlier. Tick to select that port (COM16 in this example ▼) and click <**OK**>.



4. You may need to restart SDR Console for things to work.
5. After opening the echo port in Logger32, open SDR Console and the radio setup in Logger32. You should be able to control the frequency and mode in SDR Console. Clicking the waterfall or scrolling the frequency or scrolling in the waterfall will update the radio in Logger32.



The frequency is not updated in Logger32 until the frequency updates in SDR Console *e.g.* when scrolling through the waterfall, the frequency is not updated until you stop scrolling.

Copyright © 2026 Robert C. Furzer K4CY

12.9 CAT FAQs

Q. My radio/CAT connection is playing up. What's wrong?

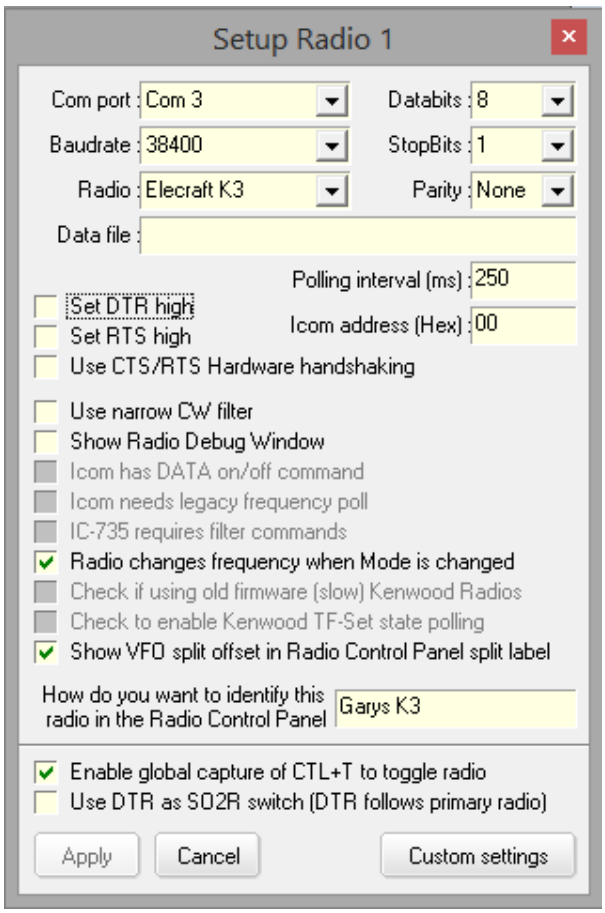
A. There are several possibilities.

First, get ready to diagnose the problem/s:

- The radio debug window is a good way to fault-find CAT connections. When working, it shows the polling from Logger32 to the radio, and the radio's responses back to Logger32. It helps to understand the command and response formats for your radio, so blow the dust off your radio instruction manual. We have included only a few examples in this Logger32 manual.
- The **Freq** panel at the top of the log entry pane shows the active radio's main VFO frequency when its CAT connection is working. If you turn the VFO knob or change bands on the radio, the frequency display updates accordingly.
- The <**Radio**> panel on the status bar has blue text when the CAT port is open, or red text if closed.

Now work your way systematically through the following checklist, starting from the top, checking and rechecking things at each stage:

- Your **radio** may be turned off. If you have several radios, possibly for SO2R, are you trying to use Radio 1 or Radio 2? Which one is which? It *is* confusing. Are you confused?
- Your radio's **CAT cable** may not be plugged in. Properly. Pushed fully home. At both ends. For the correct radio.
- In Logger32, you may not have **configured** the radio COM port correctly ▼ Check:
  - The COM port number for the serial port your radio is currently using (it *may* have changed number if you are using USB: check it in Windows Device Manager).
  - The Hex address for your model of ICOM radio, if applicable.
  - The remaining communications configuration, especially the settings in the top half of **Setup ⇨ Radio ⇨ Radio 1|2 configuration** ▶ Look up the relevant chapter of this User Manual for your radio and/or check the radio instruction manual for clues.
  - The radio type. If your model of radio isn't available on the drop-down selection, try something similar from the same manufacturer.



- Double-check the settings for the CAT comms options in your radio's built-in config menu. They should match Logger32's config form ▶

Copyright © 2026 Robert C. Furzer K4CY

- Try reducing the **baudrate**, especially if you are using an old radio and/or PC. Be sure to set the same speed on both the radio *and* Logger32! There is very little to be lost by slowing down the CAT line to, say, 9600 or 4800 baud – in fact you are unlikely to even notice any difference in practice, except that the connection will hopefully be more reliable.
- The **polling interval** may be too fast for the radio to respond to polls from Logger32. Try increasing the polling interval by 50 or 100 ms at a time, rechecking it at each stage. Longer intervals make Logger32 a little less responsive to radio VFO changes *etc*. but, again, more reliable. You will probably get used to it anyway. If it is annoyingly slow, it's time to update your radio and/or PC.

> **Hinson tip**: I found that my creaky old FT847 would take a minute or so to wake up and start communicating with Logger32 after I turned it on. Meanwhile, clicking spots achieved nothing: the radio debug window showed CAT commands being sent to – but ignored by – the sleepy radio. The log entry pane showed a VFO frequency of 0 … until eventually it sprang to life and behaved itself. Increasing the polling interval from 100 to 150 ms cured this little annoyance.

- The CAT COM port's USB hub may have nodded-off. See the next FAQ.
- Try closing the **Slave and Echo ports**, if used. There may conceivably be an issue in the way CAT messages are being duplicated, perhaps a timing problem or (shock! Horror!) a bug.
- Your **radio** may have lost the plot. Turn it off and on again to see whether it finds it.
- **Logger32** may have got itself in a muddle. Stop and restart Logger32 to see if that sorts it out.
- **Windows** may be playing up. Windows updates have been known to mess up audio and other settings, resetting defaults and levels, that sort of thing. Download and apply any current Windows updates (especially "critical" ones), reboot and check your audio settings.
- You *may* have experienced a **hardware** fault in the radio, cable/interface or PC serial port – perhaps the result of a lightning/static zap or component failure. Has the cat chewed the CAT cable?

> **Hinson tip**: you can rule out the possibility of such faults if you can successfully command the radio through a CAT connection from some other software such as N1MM+ or the radio's configuration utility. Try it now. If that works, use the same COM port settings in Logger32, and be sure to disconnect the other software (*e.g.* close it) so that it is not hogging the port. Check the settings carefully. Watch Logger32's radio debug window to check the polls and responses.

- You may have found a **bug** in Logger32. Update to the current release of Logger32. Try closing Logger32, renaming *C:\Logger32\Logger32.INI*, restarting Logger32 with its defaults, and reconfiguring the radio from scratch. Does it work now?
- Last resort: ask for help on the Logger32 reflector, patiently explaining what you have tried and found please. Tell us what radio you are using and anything else that may be relevant.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why is there a noticeable delay after I click a DX spot before the radio responds?

A. Launch the radio debug window to check the CAT comms. What happens when you click a spot? You should see Logger32 sending a QSY command for the relevant VFO, and a response from the radio confirming the new frequency. You may see additional commands being sent to the radio (in which case, open the **Radio Control Panel**, right-click the vertical strip below the corner X and open the Setup DX Spot Macros pane to check the commands defined there), and comments from Logger32 *e.g.* "QSY Marker Set" ▶ For radios that change VFO frequency when you change mode, *two* QSY commands are sent. If it all seems in order, try reducing the poll interval under **Setup ⇨ Radio ⇨ Radio 1|2 configuration**.



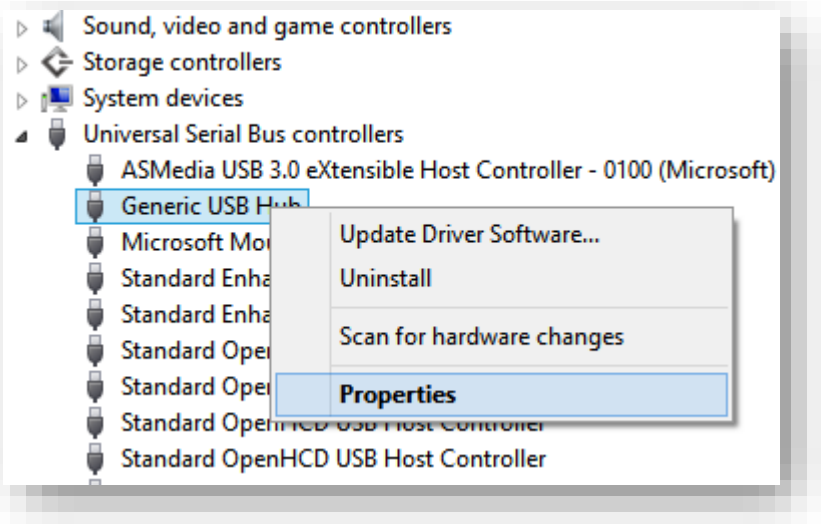
Q. After a while, the frequency stops updating and the radio becomes unresponsive to DX spot clicks *etc.* What's going on?

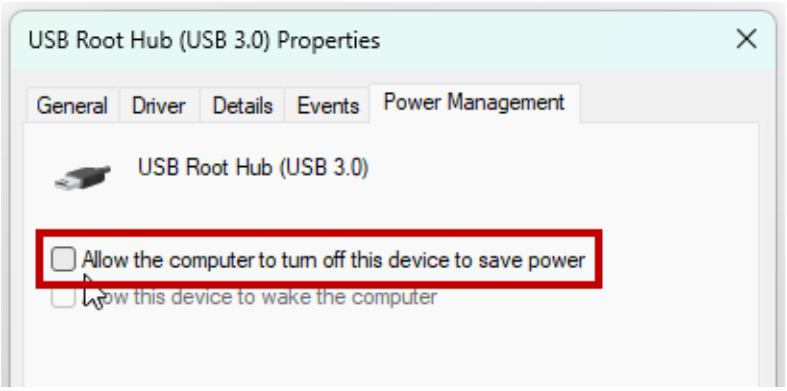
A. That sounds like your USB port is powering itself down: by default, Windows automatically powers down USB ports that *appear* inactive. This 'feature' can affect USB to serial converters used for CAT.

You can disable this annoying power-saving 'feature' through the **Windows Control Panel** ⇨ **Device Manager**.

Expand the **Universal Serial Bus controllers** at the bottom of the devices listed by clicking the little outlined triangle ▶

There may be several USB hubs. Double-click one, click <**Properties**> and then the <**Power Management**> tab ▼





Click to de-select (un-tick) <**Allow the computer to turn off this device to save power**> then click <**OK**> … and repeat for all other USB hubs listed, unless you are able to figure out which one is driving your CAT.

This *should* be a set-and-forget setting unless, maybe, a naughty Windows update silently re-enables USB power saving.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: particularly if you are using a battery-powered computer, you may want to let Windows save a few joules by powering-down *some* USB hubs *except* those used to CAT-control your radio or other peripherals that need to remain powered-up. Good luck figuring out which one is which though. The age-old process of trial-and-error may work for you. I wish I could be more helpful.

Q. Why don't my CAT macros work while I'm running JTDX?

A. Logger32's CAT macros don't work because JTDX – *not* Logger32 – has to control your radio via CAT in order to read and set the VFO frequencies and trigger the radio to transmit generated audio.

When you click **Start ⇨ JTDX** on the UDP BandMap menu, Logger32 closes its CAT port and hands-over CAT control to JTDX. Therefore, Logger32 cannot command your radio as it could when its CAT port is open. You can't change bands, change filters, update the rig's clock *etc*. directly through Logger32 until you close JTDX, whereupon Logger32 re-opens its CATs port and resumes control of your radio.

Well, OK, that's not entirely true: clicking buttons on Logger32's JTDX Control Panel tells *JTDX* to command the radio to change bands and/or modes via *its* CAT connection.

Q. What if my radio model isn't available in the list of radios in Logger32?

A. Look for a similar radio model in Logger32 - a model from the same manufacturer and model family, or one of about the same vintage – and try that.

If "nothing works" *e.g.* Logger32 does not display and cannot change the radio frequency or mode, check the basic CAT communications settings first *i.e*. the COM port number, baudrate *etc*. Logger32's settings must match the radio's settings:

- Read what the manual for your radio says about CAT port connections, settings, commands and parameters. Seriously: dig it out, dust it off and crack it open. You never know, it *might* be more useful than this Logger32 User Manual.
- Make sure you have physically connected the PC to the radio with the correct CAT lead. Is it plugged in to the correct sockets, pressed fully home at both ends?
- Use the **Windows Control Panel ⇨ Device Manager ⇨ Ports (COM & LPT)** to identify which COM port is connected to the radio. If it is not abundantly clear from the description, it is the one that disappears from the list when the CAT cable is unplugged and reappears when plugged back in.
- Use the radio's configuration menu to find/adjust the baudrate and other values for the CAT port, so they match the settings in Logger32. It *may* not be using the defaults.

If the CAT connection is intermittent, playing up, try adjusting the baudrate and polling interval to see if that helps. Generally, slowing things down gives the system more time to react.

If the CAT connection is sort-of working but a particular CAT feature, function or parameter is not, you have several choices:

- Try selecting another similar type of radio in Logger32 in the hope that *its* CAT command set suits your radio.
- Use the radio debug window to check the CAT commands and responses flowing back and forth between your PC and radio. Any clues there?

Copyright © 2026 Robert C. Furzer K4CY

- Explain your problem/what you are trying to do, and ask for help on the Logger32 reflector: *maybe* someone has a simple solution for you, perhaps a different way to achieve the same ends.
- Experiment using macros though the **R**adio **C**ontrol **P**anel to determine the specific CAT commands needed to achieve whatever it is you are trying to do.
- If the RCP approach works, can you make-do with macros on RCP buttons?
- Through the Logger32 reflector, ask for a program change (another radio type) to support your radio's command set and parameters fully.

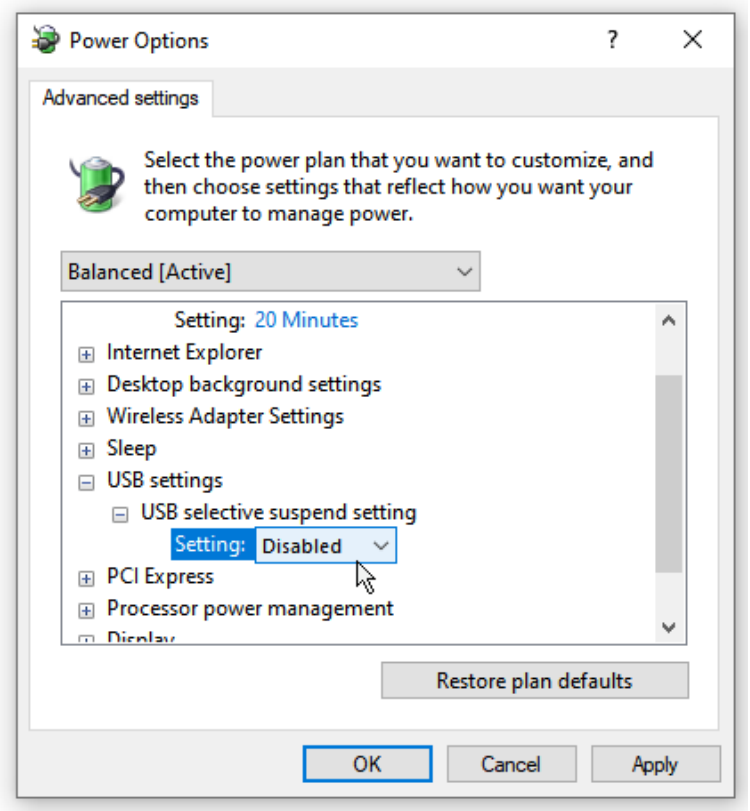
> **Hinson tip**: if Bob agrees to modify Logger32 to support your specific model of radio, he will need to know exactly what all the commands and parameters are, so your chances of success are much greater if you provide a link to the relevant page/s of the radio manual, or at least readable images of the section/s. Also, please offer to test the changes: Bob can only ensure that the listed CAT commands are generated and sent, while you will need to check that they work as they should on the radio.

- Live without it. Can you control the function from the radio's front panel or menu? If so, you'll just have to lift your hands off the keyboard and mouse to reach over to the radio, occasionally. Do it *energetically* to count as part of your daily exercise regime.

Q. After I wake my PC from sleep, the built-in USB port on my radio that was working just fine before is no longer working, so Logger32 gives an error message. How do I stop this?

A. Windows may be turning off the radio's USB serial port when the PC goes to sleep, and not turning it back on again when it wakes – a power-saving 'feature' of Windows designed to stop active USB devices from waking up a sleeping PC.

On *some* systems, the USB Selective Suspend function can be disabled through Windows Control Panel, although the setting is buried deep under **Hardware and Sound ⇨ Power Options**, then from the left side menu click **Choose when to turn off the display** or **Change when the computer sleeps** then **Change advanced power settings ⇨ USB settings ⇨ USB selective suspend setting ⇨ Disabled ▶**



Be aware, however, that with USB selective suspend disabled, sending the PC to sleep may no longer work as expected if any active USB device keeps it awake, or mysteriously wakes it up.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: my Windows 11 PC does not have the USB settings option as shown here. Maybe that screenshot was taken before I 'upgraded' to Windows 11. Boo hoo. Anyway, instead I have disabled the inactivity power-down setting on my USB hubs.

Q. Where is this mysterious 'Windows Control Panel' of which you speak?

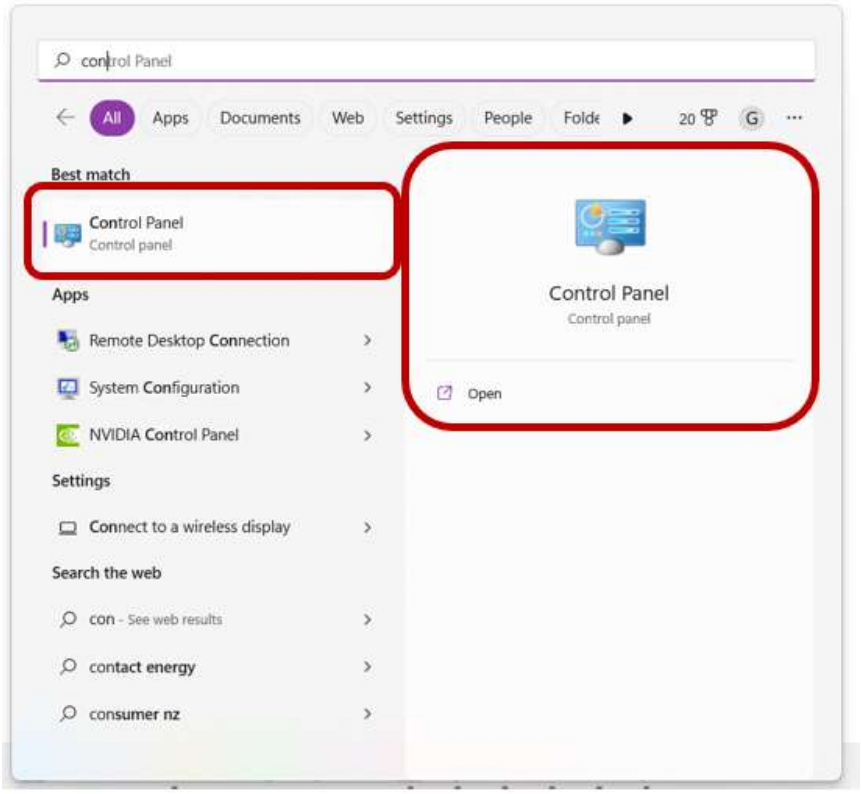
A. The Control Panel used to be dead easy to access in Windows 8.1 and previous versions using [**Win**+**X**] *i.e*. press the keyboard key printed with a Windows icon and tap the **X** key, then click **Control Panel** on the menu.

Starting with Windows 10, Microsoft made it harder to access Control Panel …

Briefly tap [**Win**], the Windows icon key, or click the Windows icon at the bottom of your screen, or swipe to open the Start menu.

Then start typing **control panel** to have Windows search for the Control Panel app.

Look, there it is! ▶



Click <Open> quickly, before it goes back into hiding!



On most versions of Windows, [**Win**+**X**] remains a quick way to access common PC system administration functions, a handy shortcut.

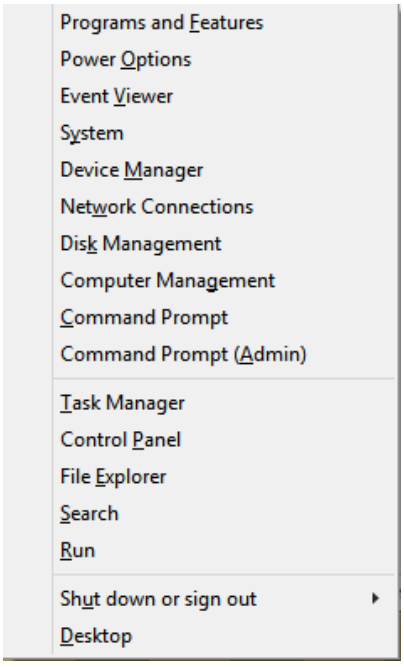
◂ Device Manager, for instance, is right there on the [**Win**+**X**] menu in Windows 11, along with Settings.

Notice the underlined letters on the menu: those are keyboard shortcuts to open the respective functions *e.g.* [**Win**+**X**] followed by **M** opens the Device **M**anager, whereas **Y** opens the S**y**stem function providing technical information about your CPU, RAM *etc*.

Copyright © 2026 Robert C. Furzer K4CY

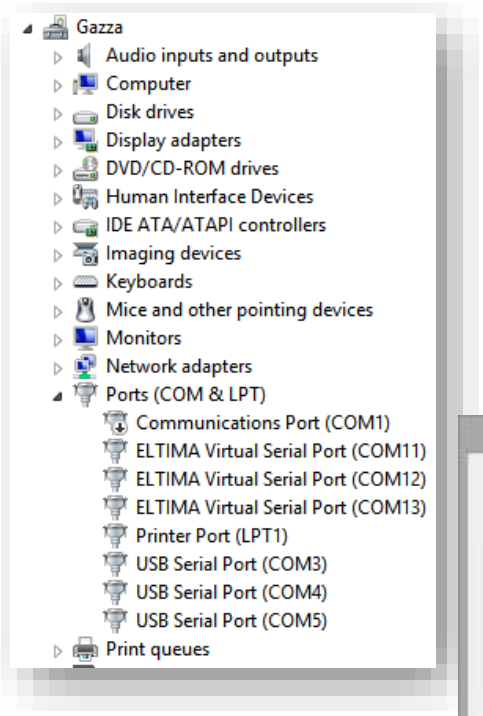
Hinson tip: there is a bewildering set of keyboard shortcuts in Windows. Rather than trying to learn them *all* or printing out a list, just think about the few things you often need to do in Windows. Look them up and start using the shortcuts. You'll soon remember the ones you use most.

Q. My USB devices stopped working and are no longer recognized by Logger32. How can I get them going … and hopefully stop this happening again?



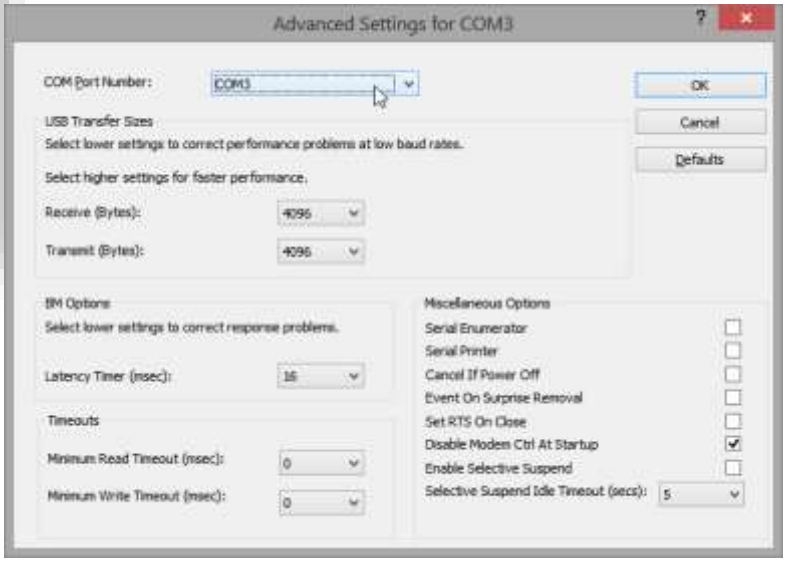
A. First, make sure all your serial port/USB devices, cables and USB adapters are plugged in where you want them.

1. Press <**Win**+**X**> to open the Windows admin menu ▶
2. Click <**Device Manager**> or press "M".
3. Click the little triangle to expand the <**Ports (COM & LPT)**> devices list if it is not already expanded (it's a toggle) ▼



4. Unplug one of your devices and watch as the list of ports changes. Watch again as you plug it back in. You should be able to figure out which port the device is currently using.

5. Right-click the COM port and click <**Properties**>.



6. Click to open the <**Port Settings**> tab, then click <**Advanced**> ▲
7. Select the *desired* COM port number for that device from the drop-down list (it *must* be an unused port!).
8. Click <**OK**> to save the port number and <**OK**> again to close the port configurer.
9. Repeat steps 4-8 for each of the other ports you need to reconfigure.

Copyright © 2026 Robert C. Furzer K4CY

Write down the new port numbers to make it easier to configure the serial ports in Logger32 if it happens again. You can reduce the chances of COM ports changing numbers by always using the same physical USB ports each time, but Windows cannot be relied upon to play the game. Windows updates are notoriously problematic. Physically label the physical ports if that helps, although you may still need to check/reconfigure their COM port numbers through the Windows device manager if they don't work.

Q. When oh *when* will Logger32 support my shiny new radio?

A. It probably already does: the CAT command sets for each manufacturer's radios are generally similar, so try using another model from the same maker.

Occasionally, new commands are added by the manufacturer to support new capabilities or functions on new models. Those won't be supported by Logger32 until the corresponding CAT coding changes are specified, developed, tested and released: you can help move that process along by finding and sharing the manufacturer's specifications for the CAT commands through the Logger32 reflector. Try pointing out any particular new commands that you *need*, and offer to test them (since neither Bob nor the Logger32 beta test team may have that shiny new radio).

Q. My radio is supported and I'm pretty sure I have set it up right in Logger32, so *why* isn't it working? *Why*? I'm tearing my hair out here!

A. Take a breath! Calm down! Here are three potential problem areas to check systematically:

1. **Configuration in Logger32**: double-check the port settings (port number, speed *etc*.), radio model, hex address (for an ICOM) against the example screenshots in this manual if your model of radio is listed in the following chapters. If not, look for a similar model of a similar vintage by the same manufacturer, or check your radio manual for instructions.
2. **Configuration on the radio:** look through the radio's configuration menus for anything to do with CAT control, port numbers, speeds, PTT and audio connections *etc*. The radio and Logger32 settings should match.
3. **CAT connection**: check the CAT cable and connectors (plural! Both ends!). Is the cable plugged in to the correct ports, pressed all the way home? Has the cable (and USB-RS232 adapter if applicable) ever worked properly, or is its first outing? Check continuity between the relevant pins, or try another cable/adapter. Use the radio debug window to check the CAT data for clues (starting with: is there any CAT data flowing? Does it flow both ways *i.e*. commands *and* responses?).

Q. Why isn't my slave port working properly, reliably?

A. Check the port settings, including the polling rate. A beta tester reported intermittent problems polling at 450 ms whereas it worked fine at 475 ms. Those extra 25 milliseconds were necessary on *his* setup … but yours may well be different.

Q. My CAT connection isn't working. Clues please.

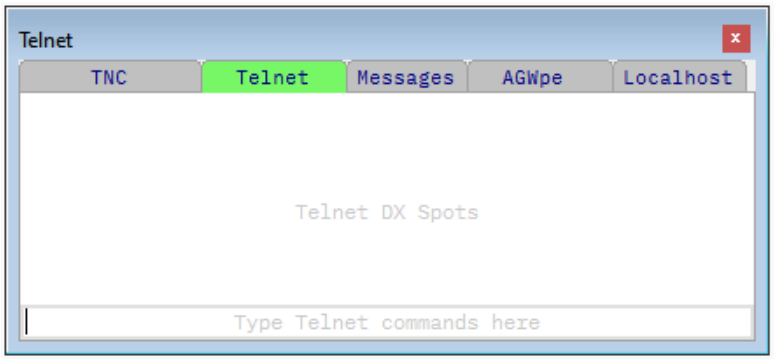
A. That's an unfortunate tail. Study every claws in this chapter, scratch some notes and hopefully soon it'll be working *purrrrrr*fectly fur you, or at least a whisker better. *I know. Dad joke. Sorry.*

Copyright © 2026 Robert C. Furzer K4CY

13 DX cluster

> "Have I done the world good, or have I added a menace?"
>
> *Guglielmo Marconi*

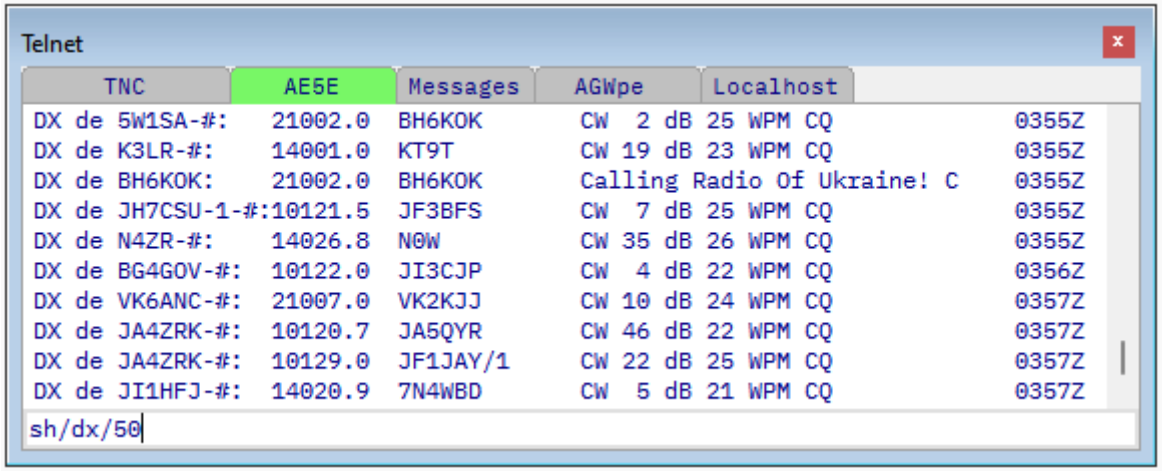
The DX Cluster pane is where you connect to one or more DX cluster nodes by various methods *e.g.* packet radio over a VHF/UHF amateur band or Telnet over the Internet. The DX cluster connection(s)[183] established through this pane feed the DX Spots pane and BandMaps their information. DX spots arriving through the tabs in the Cluster pane are merged together and passed through as one stream.



Use **View ⇨ Show Cluster Window** to open it. Having recently installed Logger32, it starts out mostly blank and intriguing ▶

The pane has five tabs for up to five DX cluster connections[184] via TNC, Telnet, Messages, AGWpe (packet radio) and Localhost, as described below.

Once it is connected to DX cluster, a tab takes on the cluster name and the area below starts filling up as DX spots flood in from around the world, scrolling upwards to make room for new spots appearing at the bottom ▼ These connections remain active even if you close/hide the pane.



[183] Yes, you can connect simultaneously to multiple DX clusters: one per tab. Don't overdo it though as processing DX spots is quite intensive and can melt the pistons in a steam-powered PC.

[184] On the <**View**> menu, panes that are grayed out are already open. If you can't see a pane, it may be hidden beneath another one, or off the screen. Use **View ⇨ Find lost windows** to locate it.

Copyright © 2026 Robert C. Furzer K4CY

Here I have connected via the Telnet tab to AE5E's DX cluster node and logged in. The window is resizable: to avoid the DX spots wrapping, make it just wide enough by dragging the right margin left or right. Drag the lower margin down to show more spots at once. Use the scroll bar on the right to look back through previous spots.

The blank row across the bottom of the pane is where I type DX cluster commands (I'm about to hit <**Enter**> to send sh/dx/50 to AE5E in this example). Simple commands (such as 'show DX' – sent abbreviated as SH/DX) are common to all DX clusters but more complex commands (such as band filtering options) vary depending on the server software used by the DX cluster node and its configuration. The command HELP tells you more about the commands and syntax used by the particular DX cluster to which you are connected.

13.1 What *are* DX spots?



DX spots are reports about amateur stations currently on the air, circulated rapidly through the global DX cluster network and distributed to connected DXers around the world.



In more detail, DX spots are structured plain ASCII text strings circulating between nodes (servers) and DXers (clients).

Spots are originated/created by 'spotters', either manually (typed by amateurs into apps on their shack computers or smartphones) or automatically (*e.g.* extracted from digital mode software or Skimmer-type robots).

DX spot data structures vary according to the DX cluster server software and configurations, plus the intended destinations. Here are three example DX spots intended for a connected cluster user (me!):

```
CC11^7074.0^NE4NE^10-Nov-2021^1846Z^looking for Kansas^NE4NE^226^226^AE5E^8^4^8^4^^^Alabama-K^Alabama-K^^ (GB7UJS)
CC11^14019.5^EA8/SA6ENT^10-Nov-2021^1846Z^CW 23 dB 14 WPM CQ^KM3T-2-#^1^^RBN^36^33^8^5^^NH^EA8^K^    ^FM42^24^28.32/15.85^ (VE7CC-1)
CC11^10115^WD4LZC^10-Nov-2021^1846Z^CW  9 dB 28 WPM CQ^WA7LNW-#^1^^RBN^8^4^6^3^KY^UT^K^K^EN78^DM37^24^38.5/85^ (VE7CC-1)
```

Copyright © 2026 Robert C. Furzer K4CY

The following DX spot fields (even if empty) are each terminated by a caret (^) and sent in turn:

Field	Meaning/use
Message type	Identifies the particular spot structure/format and destination - other DX cluster nodes or users. Type **CC11** spots, for instance, are sent to users of CC clusters (and DX Spider users *if* they have sent the command SET/VE7CC). Messages between DX cluster nodes have types **PCnn** where nn is a number between 10 and 99. PCnn messages are even more succinct with fewer data fields.
DX frequency	Reported frequency of the DX station in kHz with up to 2 decimal places *i.e*. to a maximum resolution of 10 Hz.
DX station's callsign	Reported callsign of the station spotted (who may not be "DX"!).
Date	UTC date (day-month-year) on which the spot was initially posted (sent to the first DX cluster node) *e.g*. 09-NOV-2021^
Time	UTC time (hour and minute) on which the spot was initially posted *e.g*. 2216Z^ (the Z is optional).
Spotter's callsign	Callsign of the station that posted the spot.
Comments	Notes about the DX from the spotter *e.g*. UP 1^
Hops remaining	A count that is decremented each time this spot is forwarded to another DX cluster node. Spots are no longer forwarded once the count hits zero, on the presumption that they must have got to where they should have been by then. They are ex-spots.
RX via node	DX cluster node that sent us the spot.
Origin node	DX cluster node to which the spot was initially posted.
Spotter's ITU zone	Geographical information to help locate both the spotter and the DX station. This info can be used to filter spots quite specifically *e.g*. "Only show me spots for DX stations outside my CQ zone that have been spotted by spotters in my country."
Spotter's CQ zone	
DX station's ITU zone	
DX station's CQ zone	
Spotter's state	
DX station's state	
Spotter's DXCC entity	
DX station's DXCC entity	
Spotter's grid square	
DX station's grid square	
Location	Latitude and longitude of the DX are determined and added to the spot by the last DX cluster node in the chain, using grid squares if known, otherwise states or DXCC entities. This can be used to calculate (or rather approximate) bearings and distances.

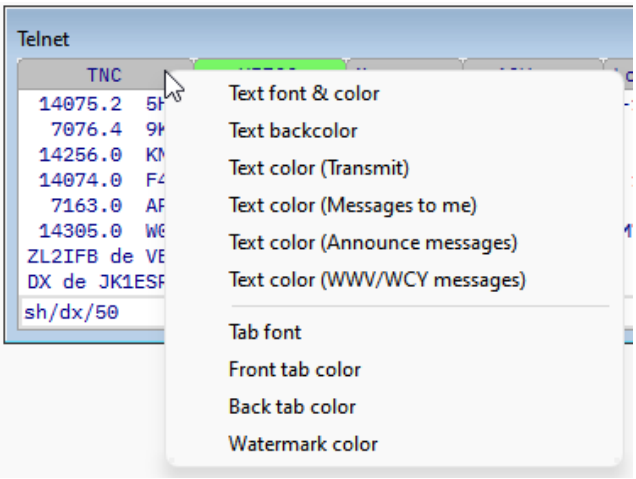
Copyright © 2026 Robert C. Furzer K4CY

Field	Meaning/use
Spot mode	Differentiates manually-entered from automated spots *e.g.* CW or RTTY Skimmer, FT# *etc*. Enables quicker look-ups and responses to SH/DX commands. Skimmer spots are simply passed along and are not stored by the cluster nodes, whereas manually-entered spots are cached for a while. 'Own spots' (DX spots for the cluster user) can be handled differently *e.g.* showing bearing and distance from the user to the spotter, rather than from the user to the spotted DX station as would normally be the case.
Spotter's IP address	Useful for identifying Internet spotters, collating statistics *etc*.
Message timestamp	Used to discard (*i.e.* drop, not forward) stale old DX spots.

Additional message types passing between DXcluster nodes carry status and administrative information such as lists of connected users, geomagnetic and solar activity reports, plus talk messages (similar to SMS/TXT, private chat messages or postcards) and announcements (similar to emails, public notices or spam).

13.2 DX cluster pane appearance

Right-clicking any of the tabs (the actual tabs, not the main body area) opens a menu to configure the pane's fonts and colors ▶



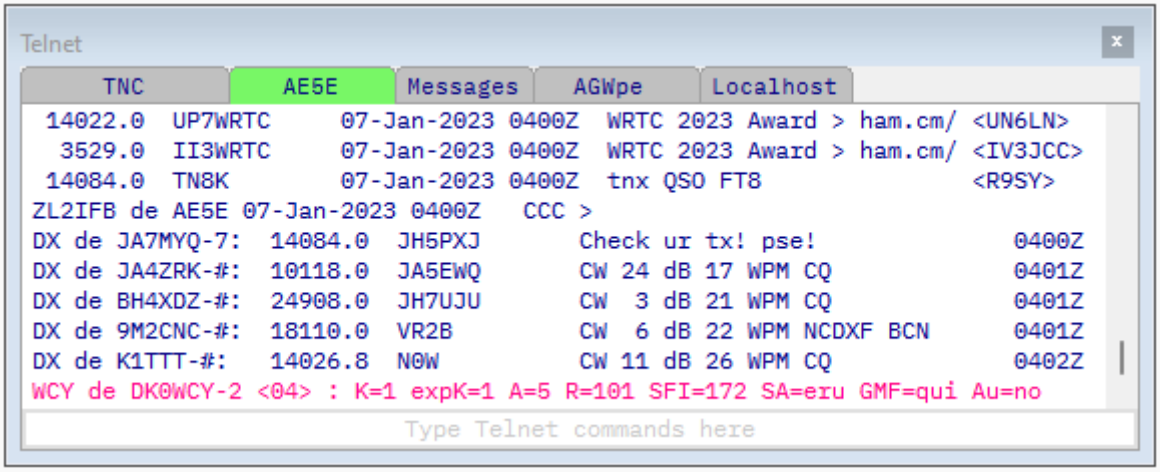
- **Text font & color** sets the text font and [fore]color for all the tabs, including anything you type into the data entry areas.

> **Hinson tip**: a monospaced font such as Lucida console or IBM Plex mono aligns the spot information in neat columns.

- **Text backcolor** sets the background color for all the text panes, including the data entry areas *while* you are entering data.
- **Text color (Transmit)** colors any commands sent to the connected DX clusters in the main body of the pane.
- **Text color (Messages to me)** colors any Talk messages sent to you by other cluster users (provided they are forwarded by the cluster node/s to which you are connected).
- **Text color (Announce messages)** colors DX cluster announcements broadcast to all connected users.

Copyright © 2026 Robert C. Furzer K4CY

- **Text color (WWV/WCY messages)** colors the solar and geomagnetic data received from WWV or WCY, circulated via the DX cluster network every few hours.



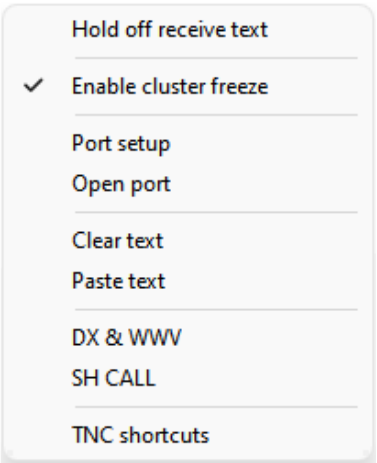
Hinson tip: ▲ a distinctive color makes the solar reports stand out from the humdrum flow of DX spots, a handy reminder that propagation conditions are constantly changing.

- **Tab font** sets the font used for the labels/names of all five tabs.
- **Front tab color** sets the color of the front (clicked, open, foreground, selected) tab. Lime green suits me.
- **Back tab color** sets the color of the other (background, unselected) tabs *e.g.* boring gray.
- **Watermark color** colors the boilerplate reminders *e.g.* for the "Type Cluster commands here" text on the empty data entry panel of the TNC tab, or where Announce and Talk messages would appear on an empty Messages tab.

13.3 TNC tab

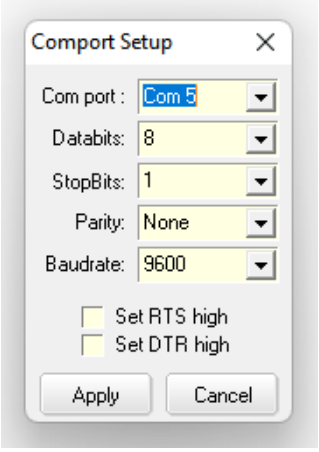
The TNC tab can receive DX spots from a VHF/UHF packet cluster station with a **T**erminal **N**ode **C**ontroller physically connected to your PC using a serial connection (RS232 or USB). It can also be used to connect to another shack computer to receive spots through *its* cluster connection.

Right-clicking the TNC text area (*not* the tab) opens a configuration menu for the TNC connection ▶

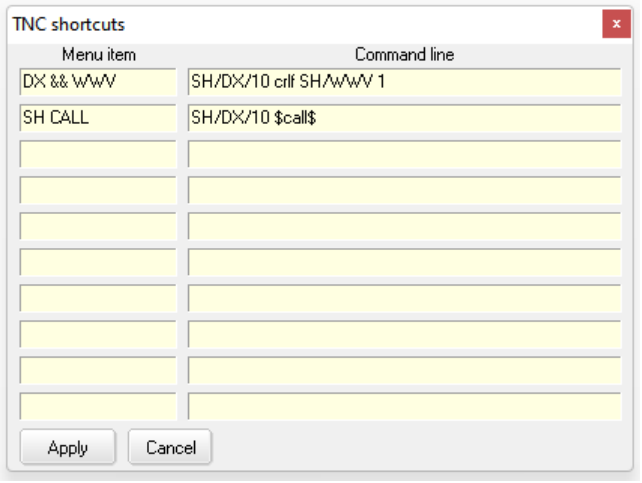


- **Hold off receive text** is a toggle: when selected (ticked), the received text pane stops updating *indefinitely* … giving you a breather to read it. Click it again to release the hold and resume the glowing updates.

Copyright © 2026 Robert C. Furzer K4CY



- **Enable cluster freeze**: stops the display updating *temporarily* while the mouse cursor is on it. Simply move the mouse cursor away to resume updating.
- **Port setup**: set the serial (COM) port parameters for your TNC ▶ Either mess around in the hope of finding settings that work … or cheat by reading the TNC manual for instructions.
- **Open|Close port** is a toggle controlling whether the TNC serial port is open for communications or shut for a rest.
- **Clear text:** wipes all text (except the boilerplate) from the TNC panel.
- **Paste text:** puts the Windows clipboard content (*e.g.* having copied an interesting DX cluster command string) into the TNC panel at the current cursor position.



- **"DX && WWV"** [185] and **"SH CALL"** are *examples* of named cluster shortcuts that I have set up using the next menu item …
- **TNC shortcuts:** up to 15 shortcuts (meaning DX cluster commands or TNC command strings/scripts) can be set up here. Each shortcut row has the menu label in the left column and the command/s[186] on the right ▶

  Multi-line command scripts are supported: separate them with **crlf** (meaning **c**arriage **r**eturn, **l**ine **f**eed) into the command string *e.g.* **SH/DX/10 crlf SH/WWV 1** grabs ten DX spots *plus* the most recent solar/geomagnetic data[187].

Macros $**call**$ (inserts the DX callsign from the log entry pane, if any) and $**band**$ (inserts the band you are currently using) are supported in the TNC shortcuts. Be sure to use *lower case* for these macros though: they are CaSe-sEnSiTiVe here.

Hinson tip: use something like **SH/DX $call$** to display DX spots for the DX station you are currently stalking, hoping to figure out perhaps where the DX might be listening. Use **SH/DX $band$** to see who else has been spotted lately on whichever band you are currently using – perhaps including variant callsigns for a DX station that might have been wrongly spotted (*e.g.* the DX station spotted as "P5DX" and may in reality be "SP5DX" or "PY5DX" or something else entirely).

[185] The ampersand is doubled-up because a single ampersand is a control character telling Logger32 that a hotkey follows: a single ampersand is not actually displayed.

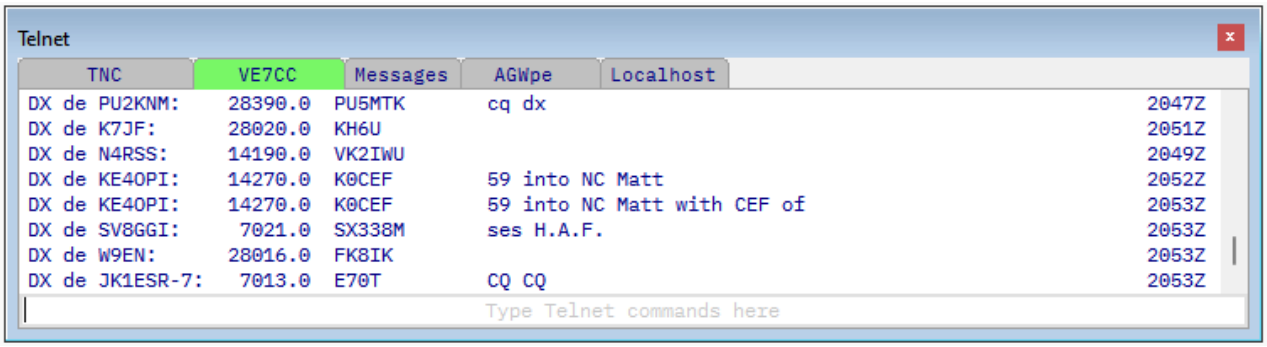
[186] Cluster commands are not case-sensitive. CAPITALS stand out for me. You may prefer lower or mixed case, or not be bothered. See if I care.

[187] Plain **SH/WWV** gets *several* WWV reports. Logger32 shows the last report received in the status line … but if the cluster sends the reports in descending order (the most recent one first), that will be the oldest of the batch, not the latest. Adding the '1' tells the cluster to send *just* the very latest one. Got that?

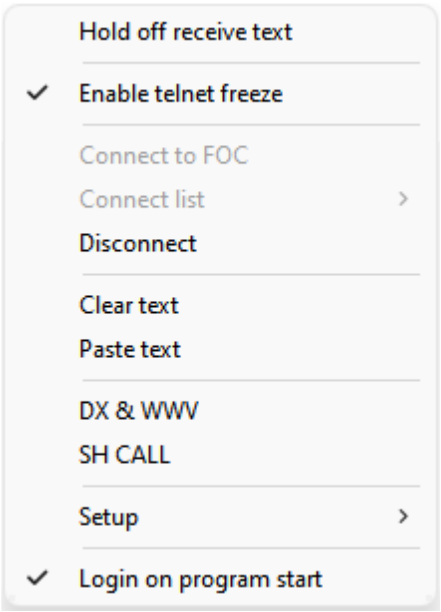
Copyright © 2026 Robert C. Furzer K4CY

13.4 Telnet tab[188]

Use the Telnet tab (the second tab) to connect to an Internet-based DX cluster node using the Telnet data communications protocol ▼



Right-click the main text area to configure and use a Telnet DX cluster connection ▶



- **Hold off receive text**: a toggle to stop and resume updating the *display* of received text.
- **Enable telnet freeze**: *temporarily* stops updating the display whenever the mouse pointer is positioned within the pane. Incoming spots are buffered and resume flowing when the display thaws. Even if this option is *not* ticked, mouse wheel movement[189] or a click freezes incoming spots for 10 seconds – long enough hopefully to browse the history and maybe double-click a line (which also thaws it). The cluster pane's caption shows 'Freeze …', the dots disappearing one by one until fully thawed. Meanwhile with the cluster display frozen, incoming spots continue flowing normally on the DX Spots pane.
- **Connect to [**cluster name**]**: if there is no Telnet connection currently open, click this to connect to the named DX cluster.
  - If you are already connected to a cluster via Telnet, this menu item is grayed out and disabled until you disconnect (using the option further down the menu). You can see the connected cluster name in the menu, as well as the tab.
  - To connect to a *different* cluster, you first need to select it using the next menu item …
- **Connect list**: displays a drop-down list of DX clusters that have been defined using **Setup** ⇨ **Setup remote hosts** in the right-click setup menu (see below). If you are already connected to one, <**Connect list**> is grayed out and disabled until you disconnect ...
- **Disconnect**: drops the current Telnet connection (if any), logging you out of the DX cluster ready to select and connect to another, or indeed select and reconnect to the same one.

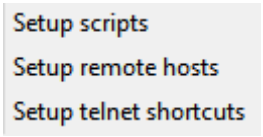
---

[188] The tabs take on the names of the DX clusters when connected (*e.g.* "AE5E") - more accurately, the names are whatever names we have defined for the cluster nodes through **Setup** ⇨ **Setup remote hosts**.

[189] On some systems, Windows is ignorant of mouse wheel events … and some deprived mice have no scroll wheels anyway. Too bad. Simply click to freeze the display. Click again to thaw. Or upgrade your mouse.

Copyright © 2026 Robert C. Furzer K4CY

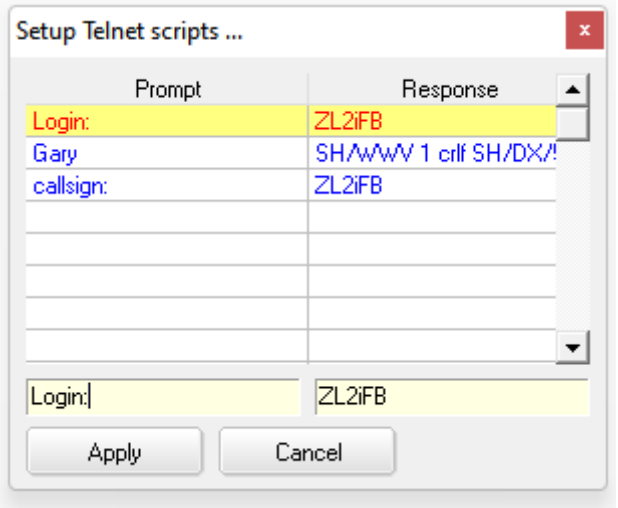
- **Clear text**: wipes all text from the Telnet panel. Clears the decks. Scrubs the floor.
- **Paste text**: pastes the content of the Windows clipboard (*e.g.* a DX cluster command copied from somewhere) into the "Type Telnet commands here" section, ready to [edit and] send.
- **"DX & WWV"** and **"SH CALL"** are *examples* of DX cluster shortcuts *you* may have added using the next menu item ▼. Once defined, up to ten custom shortcuts (cluster command scripts) will appear here on separate lines.
- **Setup**: opens an important little submenu ▶



  - **Setup scripts**: when configured, Logger32 can automatically respond when the DX cluster sends certain text, passing back commands or parameters. Scripting is useful to login to the cluster (*e.g.* sending your callsign as your user ID) and issue commands (such as SH/DX and SH/WWV 1) that you might normally have entered manually after logging in. More below.
  - **Setup remote hosts**: define the Telnet addresses and names for up to 20 Internet-based DX clusters that you may wish to connect to. More below.
  - **Setup telnet shortcuts**: up to 15 shortcuts (cluster commands) can be configured to appear on a menu, so you don't have to remember the exact syntax each time you want to run them. More below.
- **Login on program start**: when launched, Logger32 can automatically connect to whichever cluster you have identified as the default using **Setup ⇨ Setup remote hosts ⇨ Set as default shortcut**. If the cluster needs a user ID (typically your callsign, optionally adding an SSID to make it unique for each cluster connection) and maybe a password, you must either type it in manually each time or set up a simple script to login automatically …

13.4.1 Setup scripts

By specifying the prompts sent by the Telnet host (on the left), and your corresponding responses (on the right), your normal DX cluster logins can be automated[190] ▶



Logger32 continuously monitors the text streaming in on the cluster link for the Prompt strings in this form. If a listed **Prompt** string is received *exactly* as shown, Logger32 automatically sends the corresponding **Response** text. Therefore, prompts should be entered *exactly* as they are received from the cluster, in the same case, complete with any punctuation and spaces. To automate the cluster login process, copy the relevant login prompt from the screen as you log in manually, and paste it directly into the setup form Prompt field, then type your normal reply into the Response field.

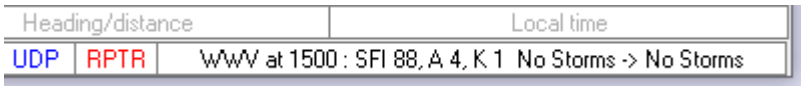
[190] Notice that I auto-respond to both "Login:" and "callsign:" with my callsign. This is because the login prompt texts vary between clusters that I normally use. With *both* options listed here, I can connect and login automatically to either type of cluster without having to enter my callsign manually … unless I come across one which uses a yet another login prompt anyway.

Copyright © 2026 Robert C. Furzer K4CY

Responses may contain multiple commands on the same line separated with <space>crlf<space> *e.g.* I use the response **SH/WWV 1 crlf SH/DX/50** to grab the latest solar/geomagnetic data from WWV *and then* show 50 recent DX spots to prime my BandMaps with a few juicy DX morsels, when the cluster sends me a welcome message containing my name after I login.

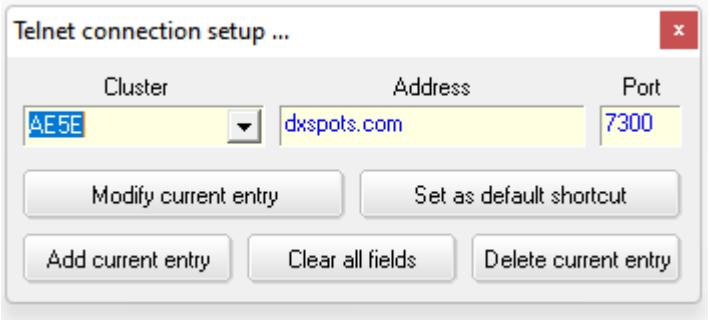
When you're done configuring, click <**Apply**> to save the configuration and close the form.

Hinson tip: incorporating the SH/WWV 1 command as part of the cluster login means the latest solar and geomagnetic information is displayed automatically in the cluster pane and in the status bar at the bottom of the screen ▼ when Logger32 starts up and connects to the cluster, without me having to wait until the solar information is next circulated by the cluster (possibly hours later).



13.4.2 Setup remote hosts

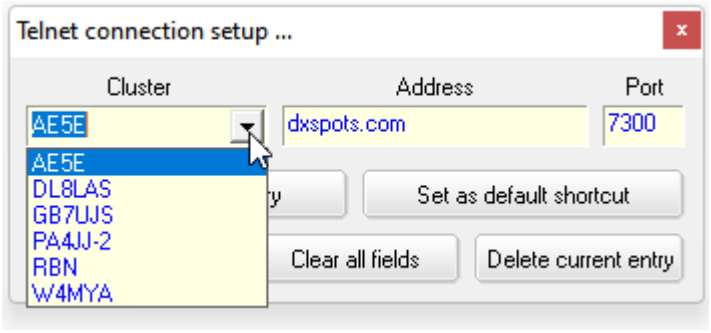
Use this confusing little form to configure up to 20 DX cluster nodes that will appear in your <**Connect list**>, and specify which one of them is the default. Luckily, this is not a frequent job.



- **Modify current entry**: edit the cluster info, having edited something in the yellow data entry boxes ▶

  The "Cluster" field is your name for a cluster connection as used in the tab [191] and the connect list. The "Address" can be either the server's IP number or its domain name. The "Port" is the server's Telnet port number: you'll need to find out the address and port from whoever runs the DX cluster (the sysop) or find it somewhere online *e.g.* from DXcluster.info/telnet/

  Click the ▼ to see the list of DX cluster nodes already defined ▶ and click any one to see/update its details.



- **Set as default shortcut**: the currently selected DX cluster will be named in the <**Connect to xxxx**> menu item, and will be connected automatically when Logger32 starts-up *if* you have ticked <**Login on program start**>.
- **Add current entry**: actually **saves** the DX cluster information currently shown in the data entry boxes on the form to Logger32's internal cluster node list. It also writes the information out to the *C:\Logger32\TelnetAddresses.INI* data file to be read-in when Logger32 is next started up.
- **Clear all fields**: start afresh for the currently-displayed cluster. Empty the fields shown.
- **Delete current entry**: get rid of it permanently *e.g.* when an expired DX cluster node finally topples off its perch. It is no more. It is an ex-cluster.

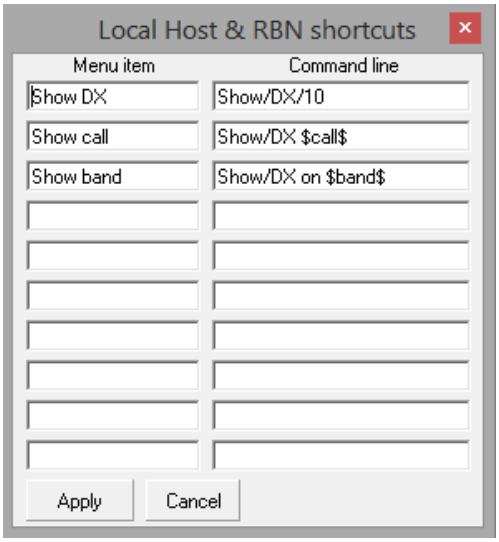
[191] The cluster name appears on the tab label during connection: if you change a cluster's name, you can disconnect and reconnect to update the tab label immediately.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: Logger32 lists *up to* **20** clusters on the connect list. If **20** is not enough for you, I suggest choosing your top most favoritest **20**, the very best and most useful **20**, the **20** star performers …

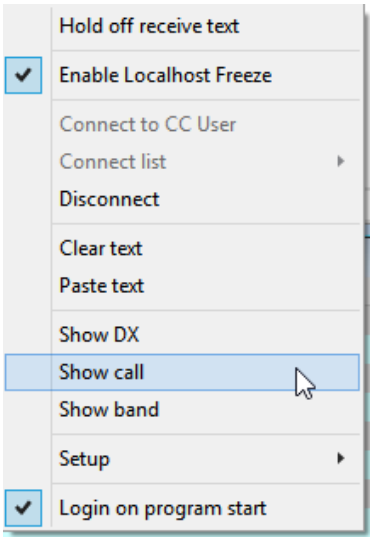
13.4.3 Setup telnet shortcuts

Use this menu option to define up to 15 DX cluster command strings. Each entry consists of text in the left column that will be displayed as an item in the menu and, in the right column, the corresponding command to be sent via Telnet to the DX cluster when you click that item.



◀ Macros $**call**$ and $**band**$ are supported in the Telnet window (*note: lower case*), using the current call and frequency shown in the log entry pane.

The 'Menu item' names for these shortcuts appear in the right-click menu of the Telnet window ▶

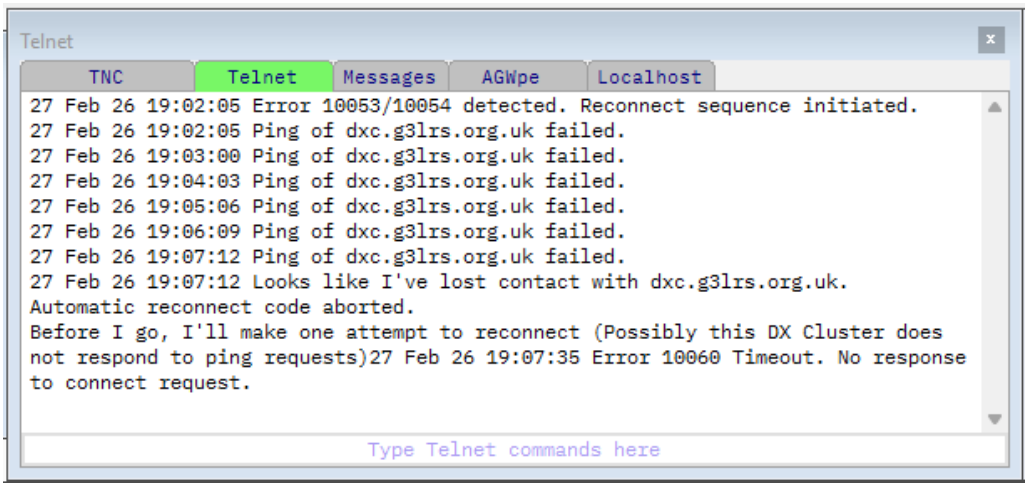


Logger32 allows command lines of up to 200 characters (although specific DX clusters may impose different limits). As with scripts, shortcuts also support multi-line or a sequence of command strings. Simply add **crlf** between the commands to send them in sequence.

13.4.4 Automatic connect *and* reconnect

With <**Login at program start**> enabled, shortly after Logger32 is launched, it attempts to connect and then login to the selected default DX cluster node via Telnet. Whereas normally the login step involves us typing our callsign in response to a succinct cluster message such as "Login:" or "Callsign:", scripting our response let's Logger32 send our callsign automatically.

If Logger32 receives Windows errors indicating the Telnet connection attempt failed, or worked but then dropped out, it attempts to reconnect automatically. It pings the DX cluster's IP address to find out if it is reachable ▼



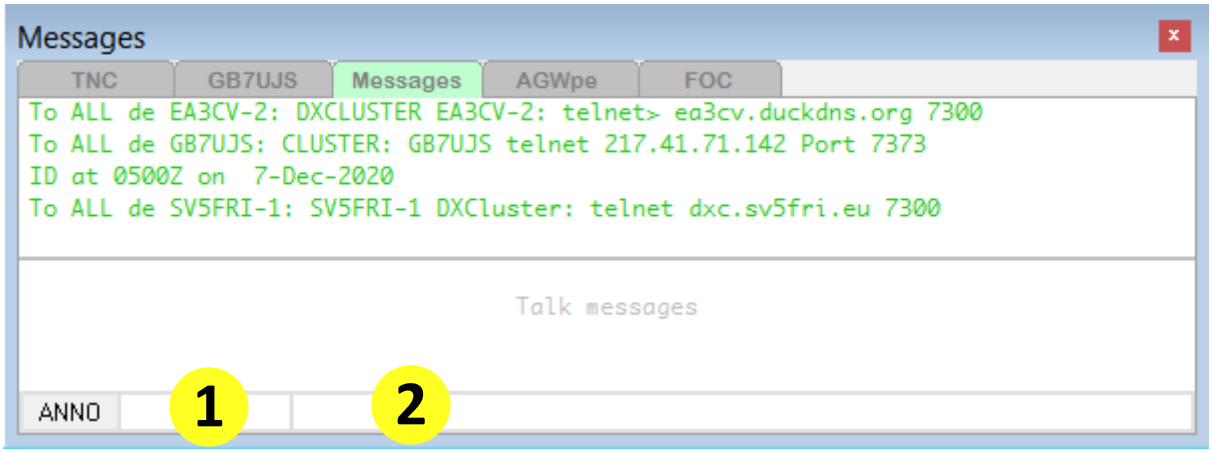
Copyright © 2026 Robert C. Furzer K4CY

- If a ping echo reply is received, Logger32 then tries to connect via Telnet. If that Telnet connection fails, no further attempts are made so we must manually reconnect.
- If no ping echo reply is received, Logger32 waits one minute and pings again, repeating this sequence up to five times.
- After five minutes and six pings without response, Logger32 makes one final desperate attempt to connect via Telnet anyway since some DX clusters have ping echo disabled. If the connection fails yet again, Logger32 finally gives up in disgust, leaving us to reconnect manually. It tried, it really tried. It failed.

Hinson tip: DX Clusters share information about logged-in users among themselves, and object to finding the same user logged in to more than one cluster node at once … so in order to be able to connect to multiple clusters (for resilience or other reasons), we normally append an SSID (a dash and number) to our callsign (*e.g.* ZL2IFB-1, ZL2IFB-2 …) to generate a unique login userID for each cluster connection.

13.5 Messages tab

The Messages tab of the DX cluster window is used for both Announcements (broadcast messages sent to all connected users) and Talk messages (incoming texts sent to your callsign[192]) ▼



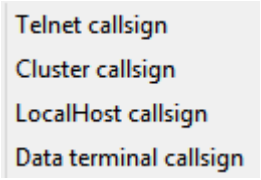
Talk messages are private[193] text messages sent between users of the DX cluster network. For example, you and a friend may be waiting patiently for a band opening on 6 meters. If you find an opening, you can send a private Talk message directly to your friend about it, without other users on the cluster seeing it as they would if you had sent an Announce message or DX spot.

[192] To capture anything from either section, click, hold and drag the cursor to highlight the required text then release the mouse button to copy it to the Windows clipboard. It can then be pasted wherever you like *e.g.* logbook notes or emails.

[193] They may be private as in 'only for the intended recipient' but not secret: there is no encryption of the Talk message content as it flows through that part of the DX cluster network that links sender to recipient. Think of it like sending someone a postcard, or talking to them on-air.

Copyright © 2026 Robert C. Furzer K4CY

Logger32 uses the callsigns defined on the DX Spots pane to determine who you are ▶



To send a Talk message, type the recipient's callsign into the empty field to the right of ANNO (1), type your message into the wider empty field to the right of that (2), then hit <**Enter**> to send it.

Hinson tip: if someone sent *you* a Talk message, click anywhere in the message to prepopulate the callsign field (1) ready for you to type your response to them into field (2).

To send an Announce message, type the message into field (2) then click the <**ANNO**> button. Any callsign in (1) is irrelevant since announcements are broadcast to everyone, although many of us configure the DX clusters to ignore them, reducing unnecessary distractions (cluster spam!).

Hinson tip: *if* you have configured the DX cluster/s to which you are connected *not* to send you announcements and/or talk messages, there will of course be nothing to show in either or both areas of this tab. This is not the place to tell you how to configure clusters, though: consult the cluster help or Google. Phone a friend. Ask the audience. Send me a talkie if I'm QRV.

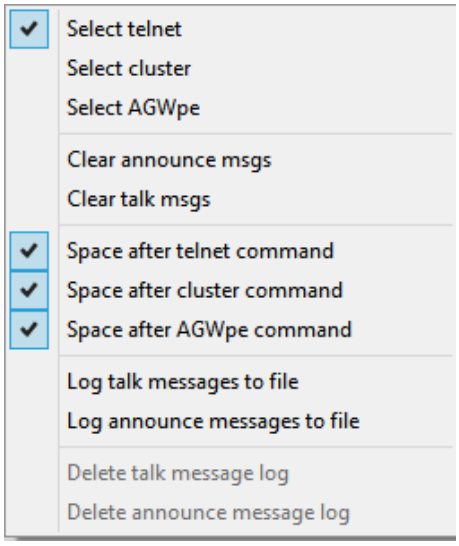
Announce and talk messages are often much longer than DX spots, so if the pane is only just wide enough for spots, announce and talk messages will wrap at the right margin to the next line. If you have a wide enough screen, simply stretch the pane out wider to avoid wrapping by clicking and



dragging the left or right margins ▼

As usual in Logger32, right-clicking the Messages window opens a configuration menu ▶

- **Select telnet|cluster|AGWpe**: which tab do you want to receive and send messages through?
- **Clear announce|talk msgs**: wipes either part of the pane.
- **Space after telnet|cluster|AGWpe command**: sends a space character at the end of the respective messages. This PreventsSuccessiveMessagesBeingConcatenated, confusingly.
- **Log talk|announce messages to file**: if you want to record Talk or Announcement messages to disk, these options save them as *C:\Logger32\TalkMessages.txt* and/or *C:\Logger32\AnnounceMessages.txt*, respectively. Although there is no Logger32 function to open/edit these files, you could create a desktop shortcut or define a utility program entry (using, say, *C:\Windows\Notepad*



Copyright © 2026 Robert C. Furzer K4CY

followed by the filename to edit) if you find yourself accessing the files frequently for some reason.

- **Delete talk|announce message log**: having created one or both log files, you can delete them with these options – or simply navigate to the folder in File **E**xplorer and delete them manually.

The Messages tab goes bright yellow ▼ to alert you *if* you receive a cluster talk message while the tab is not currently selected. The tab reverts to its normal color when you click it to open and read the message.



13.6 AGWpe tab

Use the AGWpe panel to connect to a DX cluster via **packet radio** using the AGWpe Packet Engine. The right-click options are similar to those for the Telnet tab.

13.7 Localhost (and RBN) tab

The Localhost tab provides a simple[194] TCP socket to connect – by default – to 127.0.0.1 (the IP address of the PC running Logger32, also known as Localhost) for a connection to a VE7CC Cluster User pseudo cluster node, CW Skimmer system or similar software running on the same PC.

This tab can also be used as a second Telnet tab. In this role it provides some additional capabilities, notably de-duping the spot stream coming from Telnet DX clusters that provide high-volume **R**everse **B**eacon **N**etwork feeds from the distributed global network of Skimmers.

DX spots received through the Localhost tab populate the DX Spots pane and BandMaps in the same way as those received via the Telnet tab.

Right-click the Localhost pane to configure and operate it ▶

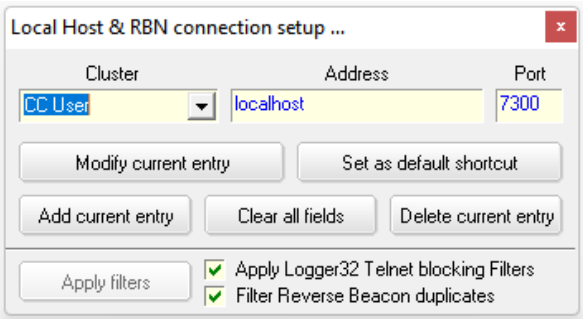


- **Hold off receive text:** toggles stop/resume the display of received text.
- **Connect to …:** lets you connect to the named DX cluster node (*i.e*. the default shortcut - see below).
- **Connect list:** shows the DX clusters you have configured.
- **Disconnect:** does what it says! Drops the mike. Lets go the string.
- **Clear text:** clears all text from the Telnet panel.
- **Paste text:** pastes in any text from the Windows clipboard at the current cursor position.
- **"Show DX"** here is an *example* of a DX cluster command you might have defined.

[194] "Simple" means there is no error recovery, a single address and port, and single prompt/reply messages only. Nothing fancy - crude you could say, straightforward and basic, limited, barely adequate.

Copyright © 2026 Robert C. Furzer K4CY

- **Setup:** is where you set up each cluster's name, address and port number that will appear in the menu ▶, plus the response to be sent when the specified login prompt is received from the DX cluster.

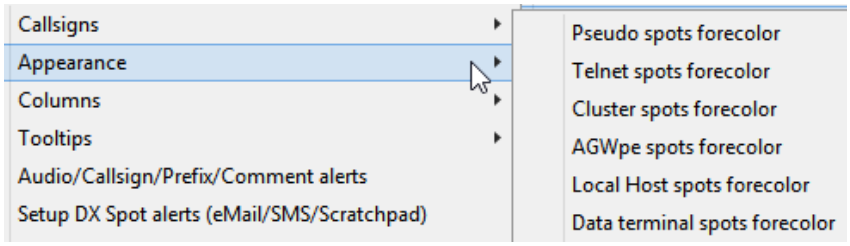


  - **Apply Logger32 Telnet blocking Filters**: you can choose whether to apply the main Telnet blocking filters here (*e.g.* to filter out FT# or SSB spots if you are a hardened CW addict). With this option off (*i.e.* do not apply Logger32 blocking filters to the localhost DX spots), you can apply Logger32 DX spot blocking filters to (say) the G3NPA connection, and DX cluster specific filters to (say) G3NPA-1. For example, on G3NPA, block all bands except 6m and all originating continents *except* Europe, whereas on the G3NPA-1 connection block 6m DX spots but pass those for the other bands.
  - **Filter Reverse Beacon duplicates**: if a new DX spot has "-#" in the originator's callsign (indicating that it came from RBN), Logger32 will check the previous 100 DX spots to see if the same DX callsign has been spotted within 1 kHz. If so, setting this toggle causes the new information to overwrite the older spot in place on the screen rather than appending a new spot to the bottom of the list. Watch the screen carefully to see this happen.

Hinson tip: don't forget to set all your callsigns & SSIDs if you are connected to several clusters on multiple tabs with various user IDs, so that incoming talk messages sent to any of your callsigns ring the bells. Right-click the DX Spots pane then use **Setup ⇨ Callsigns**.

Spots in the DX Spots pane and BandMaps can be color-coded to indicate which tab and hence which connection and which DX cluster/source they originated from.

Right-click the DX spots pane, then click **Setup ⇨ Appearance** and click to choose the forecolors[195] (text/font colors) for DX spots from each of the tabs ▶



- **Login at program start**: a toggle tells Logger32 whether to connect and login automatically to the default Telnet cluster when it starts up, or not.

13.7.1 Using the Localhost tab with CW Skimmer

Get CW Skimmer loaded and running to your satisfaction first.

[195] For example, visually distinguish human-originated DX spots coming via a Telnet cluster in, say, red text, from the stream of Skimmer robotic spots coming via a Localhost RBN connection in blue – assuming the spot comments are not distinctive enough already.

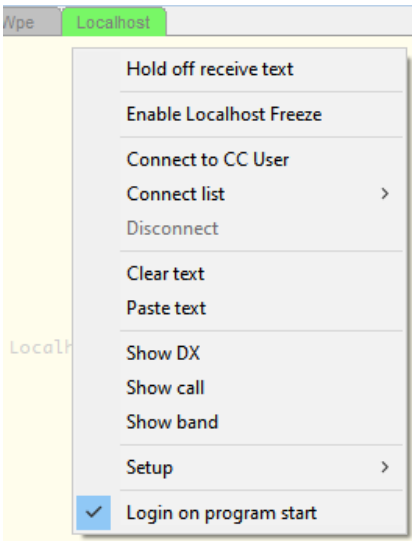
Copyright © 2026 Robert C. Furzer K4CY

You need a virtual port splitter to feed the radio frequency information obtained via CAT to *both* Skimmer *and* Logger32. For example, a secondary CAT port is available in the *micro*HAM router software to drive Skimmer, and VSP Manager is another possibility.

To see the output from Skimmer in the DX Spots pane, configure Skimmer's Telnet server using port 7300.

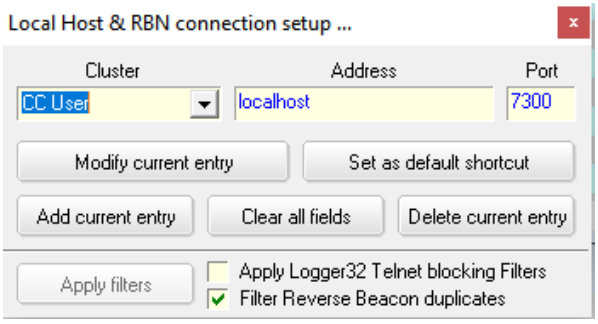


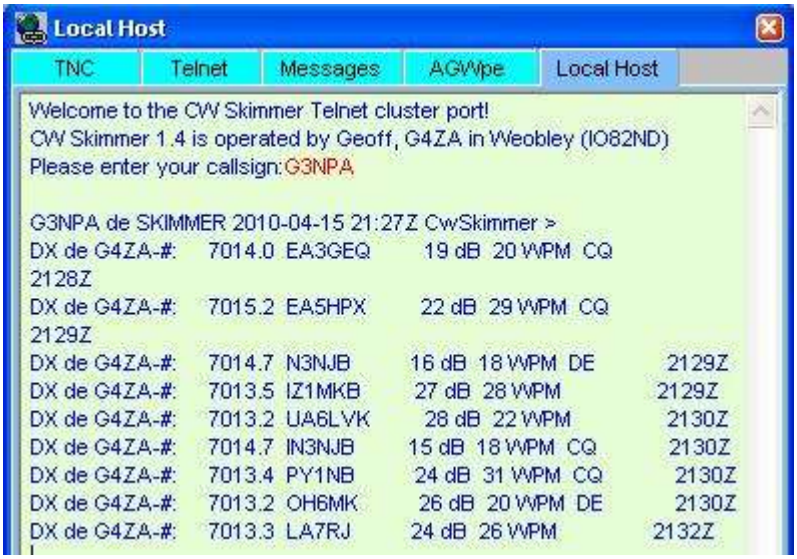
In Skimmer under
View ⇨ Settings ⇨ Telnet
tick <**Enable Telnet Server**>
and select the same port 7300 ►



◄ In Logger32, right-click the Localhost panel of the DX cluster pane and click <**Setup**>.

Set the IP address for the Skimmer server (or "localhost" if on the same PC as Logger32) and its Telnet port number ►





◄ Here, the Localhost tab is connected to a CW Skimmer.

Here is the DX Spots pane ▼

Copyright © 2026 Robert C. Furzer K4CY



In this example, spots originating from the Localhost (Skimmer) are in blue text on the DX Spots pane, whereas those coming from Telnet (DX cluster) are black. The colors are defined under the respective tabs.

Furthermore, blue spots coming from G4ZA's Skimmer setup have the origin "G4ZA-#", and the comments show the relative signal strength in dBs, Morse speed in WPM and a snippet of the CW text decoded *e.g*. CQ.

13.8 Using Telnet to connect to other computers

Although the Telnet tab is primarily intended to connect to DX cluster via the Internet, it can also be used to connect to other computers via Telnet (*e.g.* other PCs on your shack LAN). However, Logger32 does not provide a full-featured Telnet terminal, so the remote host may 'complain' by emitting garbage before the initial prompt, perhaps scrambling the prompt as well while trying to negotiate with Logger32 about how the Telnet link will be used. After you have connected, you may get messages from the remote host such as *Unknown terminal "network"* or *WARNING: terminal is not fully functional*. Hence some programs may not run and behave as you might normally expect.

Hinson tip: if your login to the remote system involves you entering a password, then, because Logger32 echoes keyed input in a different color, Logger32 will display your password on the screen. If this bothers you, check for snoops or spooks watching over your shoulder.

So, given the limitations, why would you want to use the Telnet window of Logger32 in this way? A PC on your LAN may be running a DX cluster node on the global cluster network, or it may have radios connected to it that you can remotely use to start a packet radio session with your local DX cluster, you may conceivably want to do some remote system administration through Logger32 for some curious reason, or it may satisfy your inner geek.

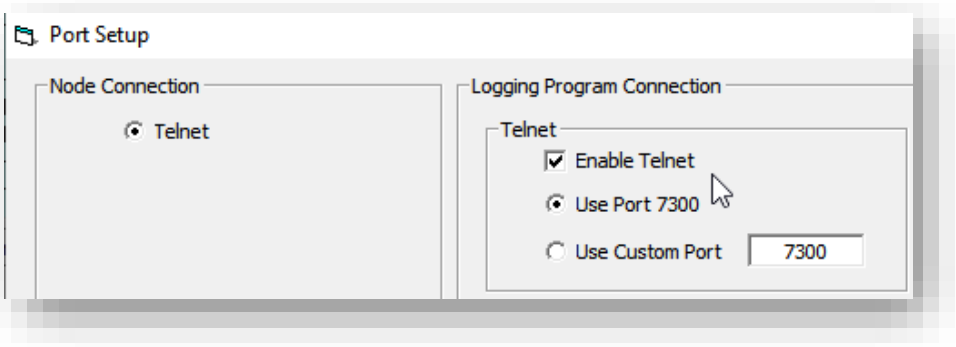
13.9 Pseudo DX clusters with VE7CC's CC User software

VE7CC's **CC User** program is a cool DX cluster client with point-n-click menus that simplify connecting to and configuring DX cluster nodes (*e.g.* setting up your band and mode filters): no more struggling with those obscure DX cluster commands! It can also act as a pseudo DX cluster node, aggregating and forwarding a stream of DX spots received from one or more remote cluster nodes to logging software running locally on the same computer or LAN - Logger32 for instance, and N1MM+.

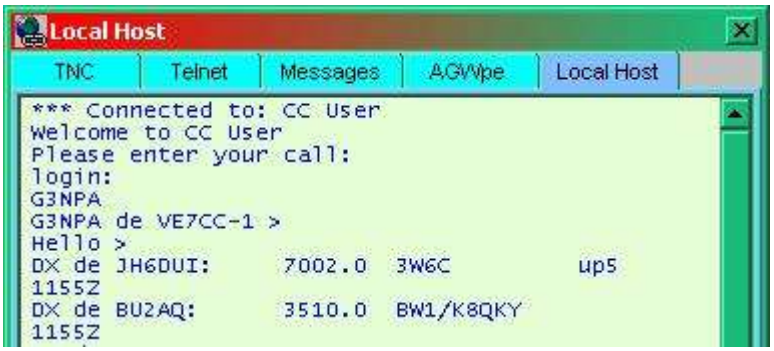
First get the CC User software loaded, running and configured to your liking. Connect to your favorite DX cluster node or nodes and tweak those filters until you are happy with the flow of DX spots.

Copyright © 2026 Robert C. Furzer K4CY

In CC User, under **Configuration ⇨ Ports/Logging program**, enable Telnet to start the pseudo DX cluster node running ▶



The default Telnet port number 7300 is probably fine unless by chance it is already being used by some other network program on the same computer, in which case pick another vaguely similar yet unique number. Either way, you will need to configure the system's address (normally 127.0.0.1 or "Localhost:") and port number in Logger32's DX Cluster pane Localhost tab or Telnet tab …



◀ Then set up Logger32 to connect via the localhost tab to the address 127.0.0.1[196] port 7300 (or rather the port you configured in CC User). Right-click the Localhost pane, then click <**Connect to …**> to connect to the pseudo DX cluster node, in the same way as you would connect directly to a regular DX cluster node.

In this example of Logger32's DX Spots pane, entries in black text came from a Telnet connection directly to a real DX cluster node, whereas the entries in blue text came from the pseudo cluster provided by CC User ▶



Note:

- If any DX spot blocking filters have been set, they can apply equally to all the data feeds, or not: there are setup options to apply or ignore the filters per tab.
- *Exact* duplicate spots are ignored … but spots for the same DX station made by different people, or with different spot comments, or at different times or frequencies, or even with different callsigns (due to copying errors and typos) are *not* exact duplicates. When a major DXpedition comes on the air, you are likely to see numerous DX spots as DXers and around the world excitedly share the news via the DX cluster network. Some will have busted DX callsigns, and some are unhelpful messages (such as "Nil here" or "When PSK?") rather than genuine spots. Logger32 lacks the brains to identify and treat them all as identical spots. Too bad. Blame the inept and over-excited spotters.

[196] 127.0.0.1 and "Localhost" both mean 'this very computer' to geeks like me. They are used for 'loopback testing' of network apps.

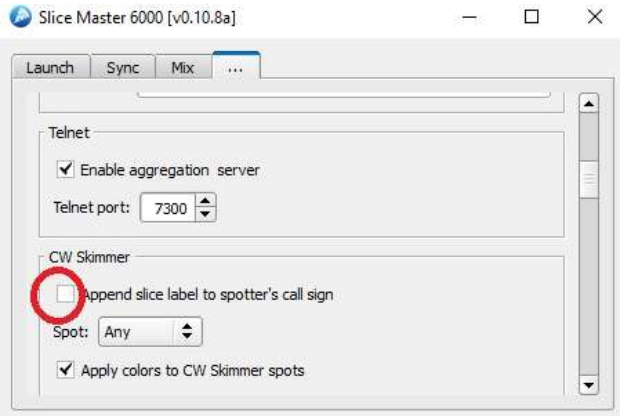
Copyright © 2026 Robert C. Furzer K4CY

13.10 Pseudo DX clusters with Slice Master 6000 software

Slice Master 6000 by K1DBO can launch and configure various applications to process data and metadata from Flex Signature SDR radios. As with Cluster User, it can:

- Collect DX spots from one or more sources such as CW Skimmer and JTDX processing slices on your Flex radio, plus your choice of Internet-based DX cluster/s;
- Aggregate them into a single stream;
- Send them onto the shack LAN using the Telnet protocol, localhost IP address and your choice of port number, emulating an Internet-based DX cluster node.

Logger32 can connect to the pseudo cluster node by Telnet in the same manner as it would to a genuine Internet-based DX cluster node. See the previous section for details, with one wrinkle …



◀ When processing DX spots from CW Skimmer on the Flex, Slice Master's spot aggregator can optionally append the slice name to the spotter's callsign … which confuses Logger32 … so don't enable that option unless you particularly *want* to cause confusion.

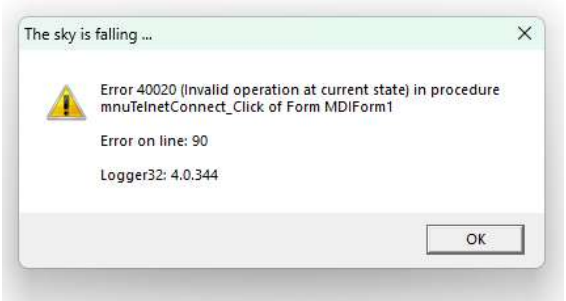
13.11 DX cluster FAQs

Q. What happens if DX clusters are offline?

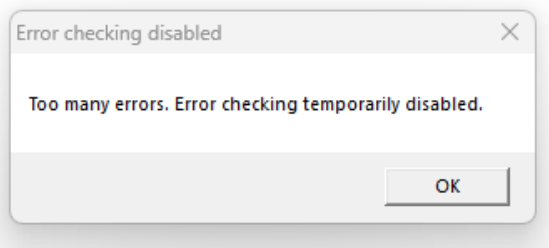
A. Many DX cluster nodes are running on consumer-grade computers in someone's shack. They are not as reliable (or as costly!) as virtual web servers on pro-grade platforms in stacks of racks in hyperscale data centers with UPSs, automated round-the-clock monitoring and failover, overlapping shifts of highly-qualified tech professionals … and all that jazz.

So, in practice, cluster nodes go down from time to time. High RF levels in the shack can add to the problems, along with consumer-grade networking equipment and "other priorities" for cluster sysops, such as DXing, contesting and 'having a life'.

When we try to connect to a DX cluster node that is currently unresponsive, Logger32 now gives us clues that it is trying hard to connect. It attempts to establish a Telnet connection … repeatedly … until eventually it gives up in disgust. Meanwhile, if we try to connect to a different DX cluster *while* it is still busy trying to connect to a dead one, we may see this ▶



Copyright © 2026 Robert C. Furzer K4CY



◀ If we persist in impatiently trying to connect, Logger32 expresses its frustration like this.

The solution is to **slow down**. Having tried to connect to a cluster node, wait patiently before trying another. Each connection attempt takes just 5 seconds or so. Chillax!

Q. Where are the other DX cluster FAQs?

A. They have been merged with the closely-related FAQs for DX spots at the end of the next chapter.

Copyright © 2026 Robert C. Furzer K4CY

14 DX spots and BandMaps

> "You miss 100% of
> the shots you don't take"
>
> *Wayne Gretzky*



Open the DX Spots pane using the <**View**> menu ◀ or the binoculars icon #3 on the toolbar.

The DX Spots pane takes the raw stream of DX spots from the DX cluster window, processing, filtering and displaying them rather like this ▶

2020: DX Spots (All op. LoTW+QSL)

?	DX Spot	Freq	Comment		Time	Origin
	E21YDP	7039.90	CW 12 dB 20 WPM CQ	+	16:47	VU3KAZ-#
	K1NPT	14074.00	FT8 -22 dB RI	+	16:47	W7HR-#
	IT9CKA	7031.10	CW 22 dB 20 WPM CQ	+	16:47	CT7AHV-#
	Z35Y	7010.40	CW Ukrainian DX Contest	+	16:47	R3OR
	HA1BF	3573.00	FT8 +19 dB		16:47	S50U-#
	5P2A	3573.00	FT8 - 3 dB	+	16:47	S50U-#
	IW1RGP/QRP	10136.00	FT8 -19 dB		16:47	S50U-#
	R3BB	3573.00	FT8 - 5 dB	+	16:47	S50U-#
	UA9CGL	3573.00	FT8 -19 dB	+	16:47	S50U-#
	WB8ERJ	7180.00	OH POTA K-1946 OH		16:47	K4CAE

DX spots can be visually highlighted and sound audio alerts if they are new ones (DX stations not yet worked and/or confirmed ever, not yet this year, or not on this band and/or mode), based on information derived from the open logbook, for the selected operator and QSL types. Select the operator and QSL type from a menu within the worked/confirmed table. The DX spots and worked/confirmed table captions display that selection ("All op. LoTW+QSL" in this case).

Clicking a DX spot on the pane sends your CAT-connected radio to the spotted DX frequency *and* sets the relevant mode if that is specified in the spot, otherwise Logger32 determines the mode as best it can according to your bandplan.

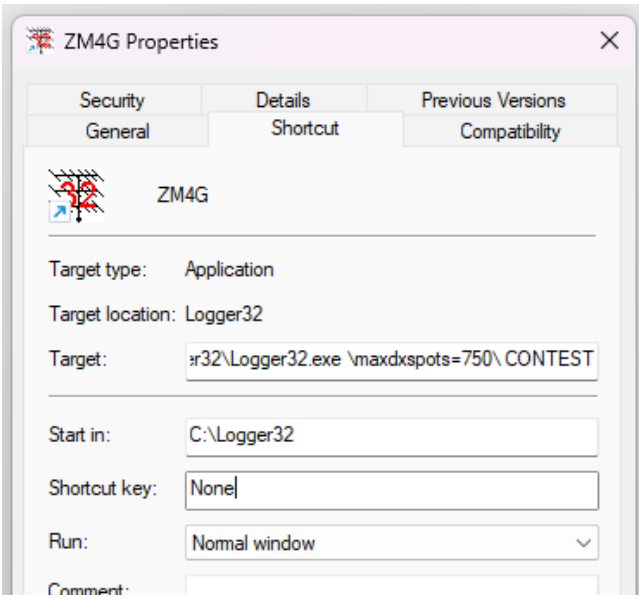
Right-clicking the DX Spots pane then clicking <**Reset radio frequency**> sends your radio back to where it was *before* you clicked the DX spot: when some exotic DX station is spotted, you can quickly click the spot to check whether you can hear him and, if not, <**Reset radio frequency**> to continue whatever you were doing before *e.g.* tuning around and listening hard, or CQing.

Copyright © 2026 Robert C. Furzer K4CY

The DX Spots pane has the following columns:

Header	Content
?	'Worked before' markers (X's) and LoTW user indicators (green blobs) appear in this column as applicable
DX spot	Callsign of the spotted station, the spottee, the DX[197]
Pfx	The DX station's prefix or country[198]
Freq	Frequency: the frequency format[199] is configurable using **Setup ⇨ DX spots**
Comment	Free format comments of a maximum length determined by the DX cluster network (longer comments are truncate …
Time	UTC time that the spot was originally posted
Origin	Callsign of the originating station, the spotter who spotted

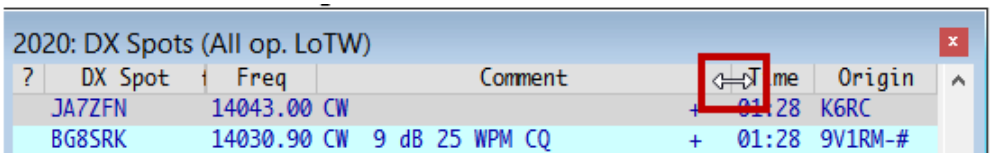
Hold <**Shift**> or <**Ctrl**> or <**Alt**> while clicking a DX spot on the DX spots window or BandMaps to send the frequency and mode to the *other* VFO (VFO B) of SO2V-capable radios, unless you are operating SO2R in which case it moves VFO A of the *other* (non-focus) radio instead.



The **\maxdxspots=**nnnn command line parameter determines the maximum number of spots that can be tracked and displayed in the DX Spots pane. nnnn must be between 300 and 1,000 spots[200].

◀ The shortcut shown here launches Logger32 with the capacity to track and display up to 750 DX spots at a time (when the 751st spot arrives, the 1st one is discarded … and so on).

Hinson tip: increasing **maxdxspots** consumes some working memory and increases CPU load, but can be useful on a decent PC, especially during a major contest when spots are flowing fast, filling the band maps with potential QSOs.



Alter column widths in the DX Spots pane by mousing-over the divider between column headings. When the cursor changes to a double-ended arrow ▲ click and drag the column divider left or right to narrow or widen the column to the left.

[197] *Please* remember always to **verify the DX station's callsign** before logging him/her. A fair proportion of DX spots circulate busted calls, typically due to someone sending CW ineptly (*e.g.* withthewr ongspa cing) or garbling their callsign on voice (*e.g.* using "funny" phonetics or speaking with a strong accent), or due to copying errors and typos by the spotters.

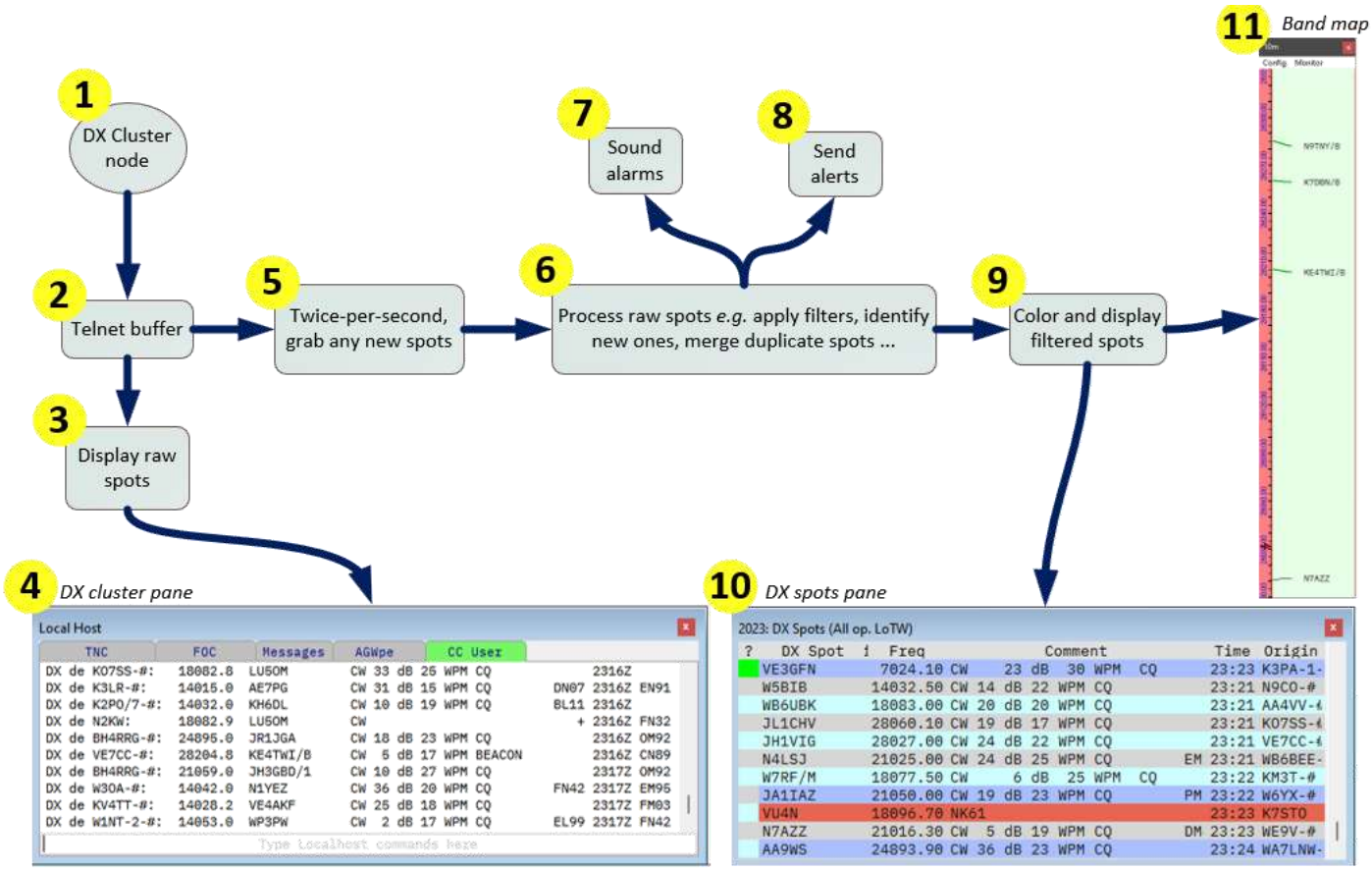
[198] The Pfx column is not displayed in the DX Spots pane example above, because I can generally figure out the prefix for myself! Even with two monitors, screen space is a scarce and valuable resource.

[199] The same format is used to send spots to DX cluster.

[200] Spot processing is quite demanding on the CPU. On a slow PC, 300 may be the most it can handle.

Copyright © 2026 Robert C. Furzer K4CY

14.1 DX spot processing overview[201]



1. DX spots stream in through Telnet connections from one or more DX cluster nodes.
2. The DX spots are temporarily buffered as they arrive[202], pending further processing.
3. Raw DX spots are sent to the DX cluster pane (if displayed).
4. New spots are added to the bottom of the list in the DX cluster pane, shifting the currently displayed lines up accordingly.
5. Every 500 ms, Logger32 grabs any newly-arrived raw spots from the buffer.
6. The raw spots are processed[203] in various ways, such as applying any band/mode filters defined by the user, identifying new ones (*e.g.* DX stations that would be a new DXCC if worked on the spotted band or mode) and merging duplicate spots for the same DX station spotted by multiple spotters on the same frequency at the same time.

   Spot processing is fairly CPU-intensive and can build up a backlog when spots are arriving thick and fast *e.g.* during major contests.
7. Audio alarms are sounded for new ones if so configured.
8. Email or SMS alerts are sent for new ones if so configured.

---

[201] This is how *incoming* DX spots are processed. Read about *sending* DX spots here.

[202] By default, the buffer holds up to 300 spots. If the buffer fills up, older, stale spots are simply discarded without being processed to make way for fresh spots. To extend the buffer, see above.

[203] Processing of spots on the list pauses during higher priority activities *e.g.* if the user mouses-over one of the panes, right-clicks a pane to configure something or click 'hold off', or while JTDX is busy decoding the latest batch of FT# messages received, while the software CW Machine is transmitting, or while loading/unloading MMTTY TX/RX buffers.

Copyright © 2026 Robert C. Furzer K4CY

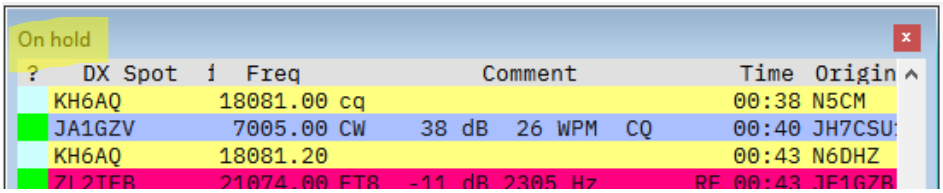
9. Processed spots are sent for display on the DX Spots pane and BandMaps (if shown) with flags indicating new ones if applicable.
10. New spots are added to the bottom of the list in the DX Spots pane, shifting the currently displayed lines up accordingly. They are also colored appropriately if they are new ones.
11. New spots are added to the BandMap/s for the relevant band, shifting others to make space if necessary. They are also colored appropriately if they are new ones.

14.2 DX Spots pane right-click menu

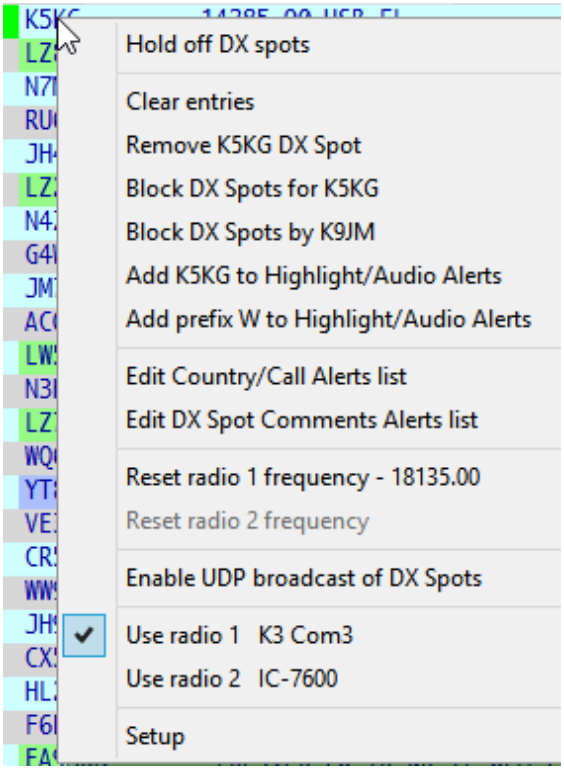
As usual in Logger32, right-click the DX Spots pane for an *extensive* menu of configuration options:

- **Hold off DX spots**: temporarily suspends the *display* of incoming spots[204]. The caption shows "On hold" ▶

  Click the menu item again to release a little rash of spots buffered in memory while on hold.



- **Clear entries**: wipes the DX Spots pane.
- **Remove [**callsign**] DX spot***: remove the selected spot from the DX Spots pane and the maps.



Having right-clicked a DX spot for K5KG on the 17m BandMap to bring up the menu, clicking <**Remove K5KG DX spot**> would wipe all currently visible spots for K5KG from the BandMaps, the DX Spots pane *and* the DX spot map ▶

- **Block DX Spots for [**callsign**]***: *future* spots for the selected DX callsign will not be shown. Any currently visible spots, however, remain until they timeout, unless you manually remove them. See also the paragraphs on blocking filters.
- **Block DX Spots by [**callsign**]***: *future* DX spots posted by whoever spotted the selected spot will not be shown. If someone's DX cluster spots are annoying you (*e.g.* they are promiscuously spotting non-DX stations, or making too many errors with the DX callsigns), simply block them. Again, their currently visible spots remain until they expire. See also blocking filters.
- **Add [**callsign**] to Highlight/Audio Alerts***: a simple way to highlight and sound an alert whenever this DX station is spotted in future.

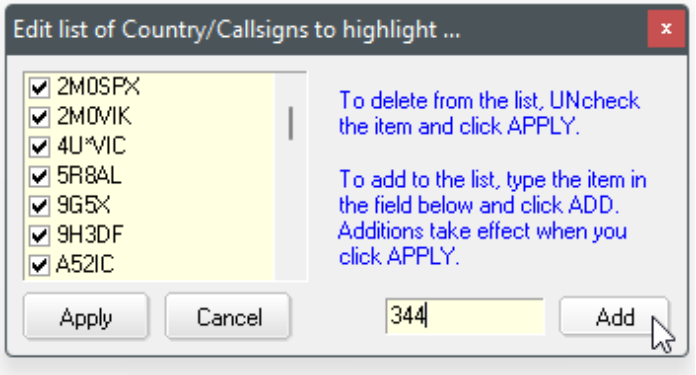
[204] As well as freezing the DX Spots pane, Logger32 also freezes the BandMaps and audio alerts.

Copyright © 2026 Robert C. Furzer K4CY

- **Add prefix** [DXCC prefix] **to Highlight/Audio Alerts***: a simple way to highlight and sound an alert whenever anyone from this DXCC entity[205] is spotted in future.

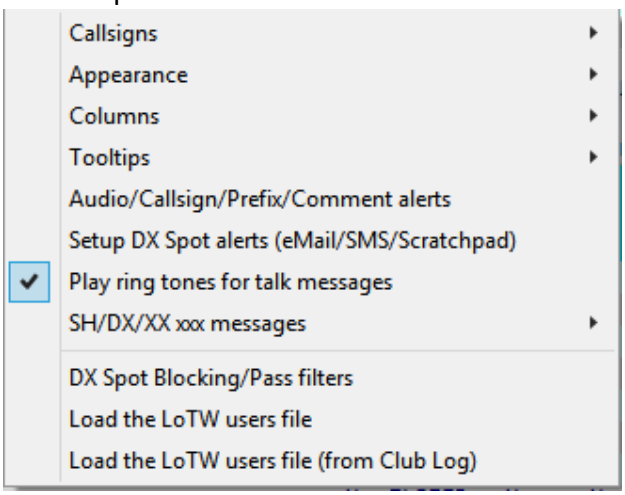
* *The call/country specific entries on this menu only appear if you right-click a line on the DX Spot window that has a DX Spot, so Logger32 has a callsign and country to work on. Right-click a blank line and those asterisked-entries are missing from the menu.*

- **Edit Country/Callsign Alerts list**: review and perhaps adjust the list of DXCC entities and callsigns for which DX spots trigger alerts[206] ▶



> **Hinson tip**: this form accepts wildcards (such as 4U*VIC for 4U1VIC, 4U2VIC …), DXCC entity numbers (such as 344 for North Korea) and continents (AF, AN, AS, EU, NA, OC or SA) in the same manner as the DX spot blocking filters.

- **Edit DX Spot Comments Alerts list**: a similar alerting function, triggered by strings such as IOTA, SOTA or QRP in DX spot comments.
- **Reset radio 1|2 frequency** - [frequency]**:** returns your CAT-connected radio's VFO to the specified frequency, as it was *before* you clicked a DX spot[207].
- **Enable UDP broadcast of DX Spots**: DX spots are broadcast onto your local network via UDP port 12061 for use in other programs.
- **Use Radio 1|2**: swap to either radio in an SO2R setup.
- **Get DXpedition info**: *if* this option is shown bold, clicking it takes us to the DXpedition's information page at DXWorld. More below.
- **Setup**: this opens a menu of configuration options, with a further five submenus ▶ Read on for more.



[205] That's just the primary prefix for an entity, taken from the official DXCC list: others may be used on air.

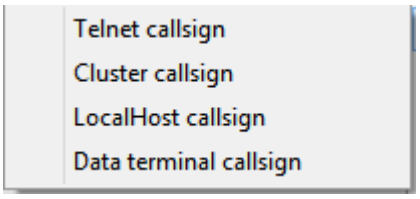
[206] This is a convenient way to catch your attention when someone specific turns up. 4U1VIC, for instance, would not normally be highlighted since I have worked Austria, the corresponding DXCC entity, many times … but 4U1VIC counts separately for the DX Marathon so I'd like to know when it is spotted.

[207] Handy, this, to slip away from calling CQ to chase some spotted DX, then return to the CQ frequency. There is a pair of RCP macros for this now: if you would prefer to click an RCP button rather than access the right-click menu on your band map, check out **$ResetRadio1$** and **$ResetRadio2$**.

Copyright © 2026 Robert C. Furzer K4CY

14.2.1 DX spots *setup* submenus

- **Callsigns**: set the callsigns or login IDs you use in the Telnet, Cluster, localhost and/or Data terminal windows ▼



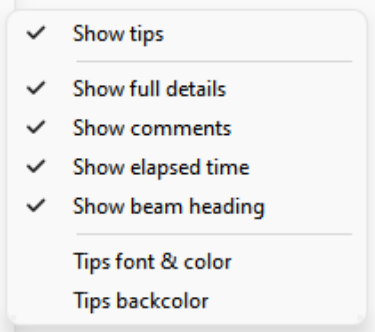
Logger32 uses these callsigns/IDs to recognize and alert you when someone sends you a talkie (DX cluster talk message) through any of the DX cluster tabs.

- **Appearance**: customize the look and functionality of your DX Spots pane using this *extensive* suite of options. Read about the <**Appearance**> submenu below.

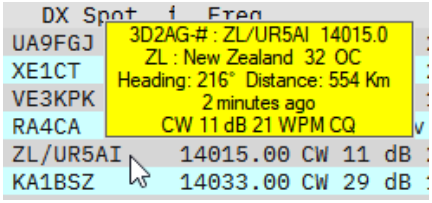


- **Columns**: choose the columns shown for each DX spot line ▶ Most are self-evident. <**Show full country name**> changes the Pfx (prefix) column header to read "Country", displaying the names of the corresponding DXCC entities instead of the WPX prefixes.
- **Tooltips**: choose what (if anything) appears in the pop-up info box when you mouseover a DX spot, plus the font and colors ▼

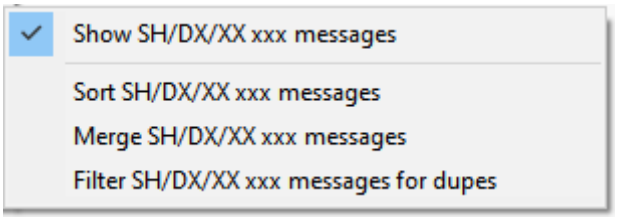
e.g.



gives



- **Audio/Callsign/Prefix/Comment alerts**: audio alerts are explained below.
- **Setup DX Spot alerts (eMail/SMS/Scratchpad)**: see below.
- **Play ring tones for talk messages**: tick this to play *RING.WAV* on your default audio system when someone sends you a DX cluster talk message.
- **SH/DX/XX xxx messages** opens a submenu ▶



  - **Show SH/DX/XX xxx messages**: DX spots returned by 'show DX' commands are sent to the DX Spots pane. Disable this option to collect information from the Telnet window in the background, *without* updating the DX Spots pane. If this is the only option enabled, a SH/DX command appends DX spots to the DX Spots pane in whatever order they are received from DX cluster. It can however be used in combination with one of the three following options …
  - **Sort SH/DX/xx xxx messages**: after you send the SH/DX command, DX spots are sorted in chronological order and appended to the bottom of the DX Spots pane, making it obvious that the SH/DX command was received and processed by the cluster.

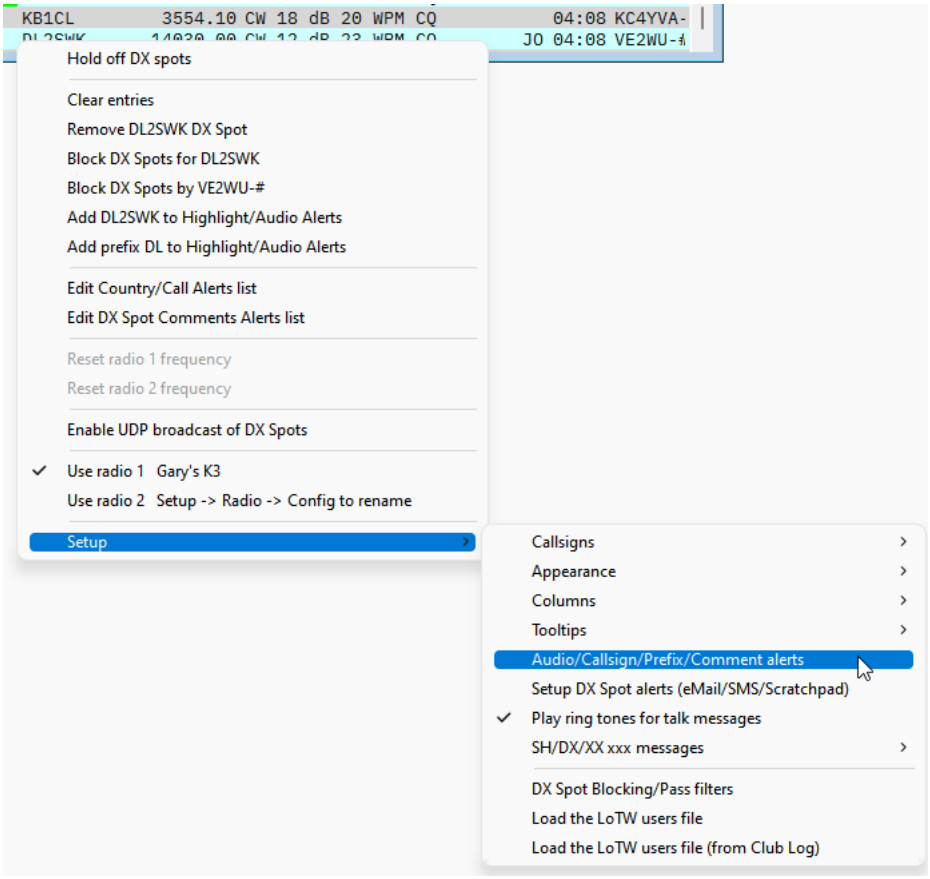
Copyright © 2026 Robert C. Furzer K4CY

- **Merge SH/DX/xx xxx messages:** after you send the SH/DX command, DX spots are merged with entries already in the DX Spots pane.
- **Filter SH/DX/xx xxx messages for dupes**: duplicate messages are filtered out without appearing on the DX Spots pane at all.

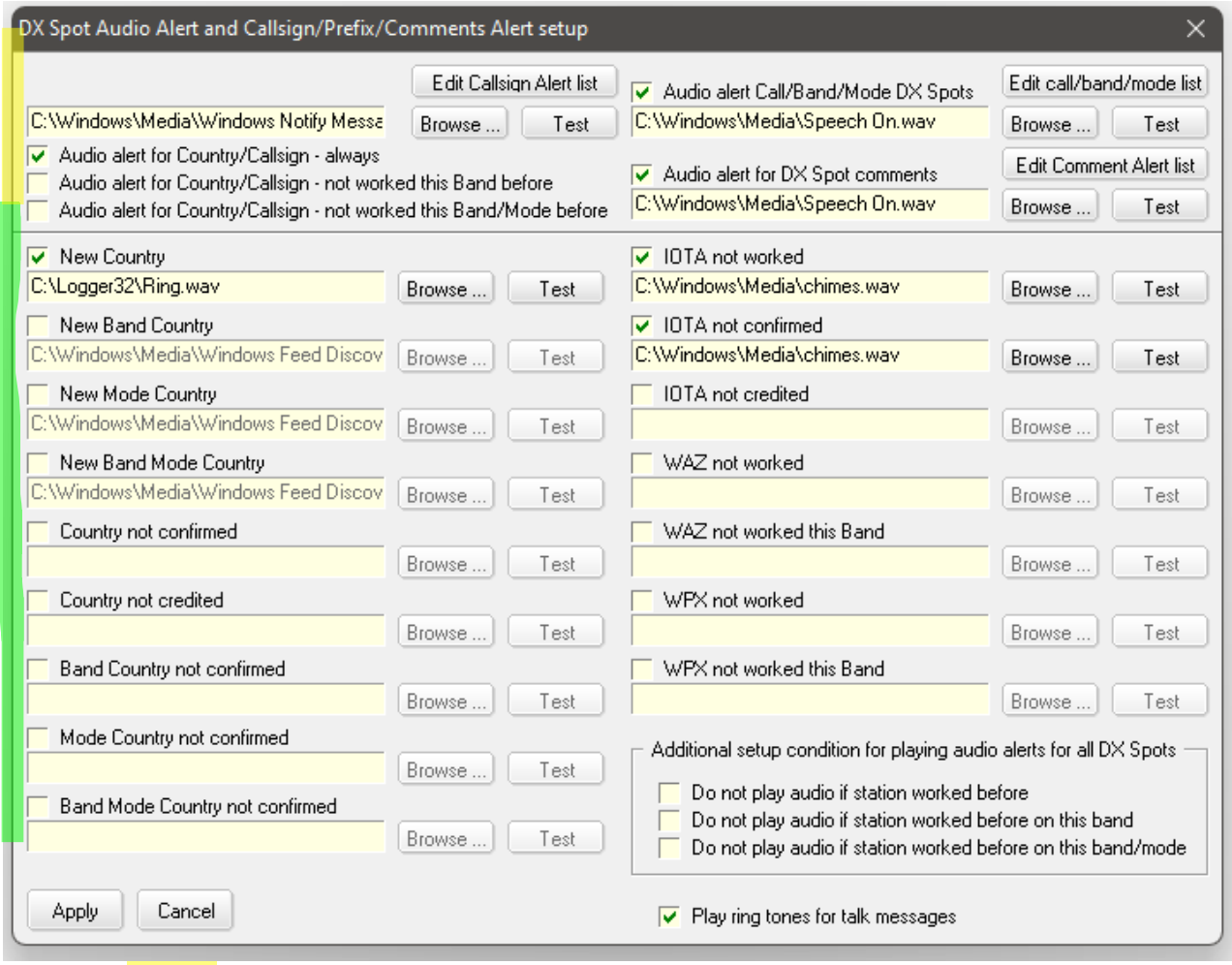
- **DX Spot Blocking/Pass filters** opens a pane with several tabs to configure the rules for determining whether various types of DX spot are shown or not. See below.
- **Load the LoTW users file [(from Club Log)]**: Logger32 can highlight callsigns known to use Logbook of The World and Club Log's **O**nline **Q**SL **R**equest **S**ervice, or to belong to a club. See below.

14.2.2 Audio alerts for spotted DX stations and comments

Right-click the DX Spots pane, then click **Setup ⇨ Audio/Callsign/ Prefix/Comment alerts** to open a form with which to configure whether and how Logger32 audibly alerts you to various 'new ones' ▶



Copyright © 2026 Robert C. Furzer K4CY



Use the upper section of this form ▲ to set 'conditional' audio alerts for specific callsigns, countries (DXCC entities), even specific bands and modes, or particular DX spot comments. These narrow selections take precedence over the lower section's more general alerts.
For most options on this forbidding form, the configuration process is much the same *i.e.*:

- Decide what it is that you want to be alerted about – maybe activity from a country you haven't worked yet on a particular band or mode, or from a friend, or a new prefix …
- Find the section on the form for whatever it is you want to be alerted about (*e.g.* New Country) and click to select (tick) that section;
- Click the associated <**Browse …**> button, then navigate through the audio files on your PC to find and select an appropriate sound for that alert[208];
- Click <**Test**> to sound the selected alert[209]. Is it what you expected? If not, resume browsing and selecting sound files. Otherwise, move on to the next type of alert;
- When you are all done, click <**Apply**> to save and start using the alerting settings, or click <**Cancel**> to discard any changes you were thinking of making, having changed your mind.

[208] Various sound files (typically under *C:\Windows\Media*) are used for Windows alerts, warnings and notifications. You *can* select any of them for Logger32 but it may be confusing if Windows uses the same sounds. Others can be found on the web, or you can make/record your own.

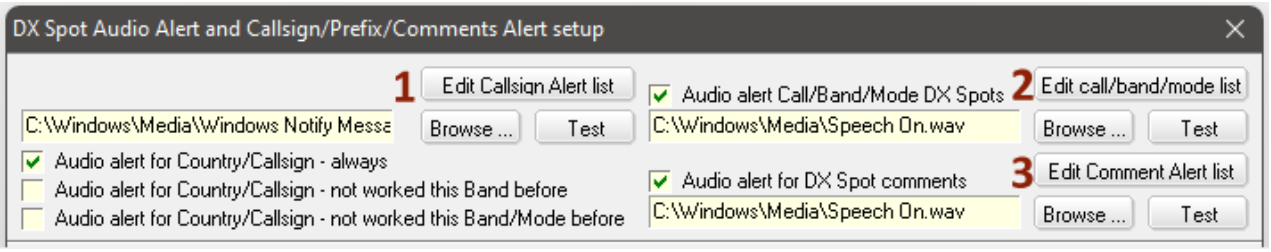
[209] The <**Test**> buttons are only active and hence only play the sounds *if* the alerts are selected (ticked).

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: although Logger32 can play different sounds for each of 19 distinct alerts on this form, you would probably find such variety extremely confusing in practice. Personally, I can only cope with a handful before I lose the plot, forgetting the association between particular sounds and the corresponding alerts. Spoken cues might work though – recordings of someone reading out the alerts[210], perhaps for a white-stick DXer keen not to miss out on anything? Or Morse characters?

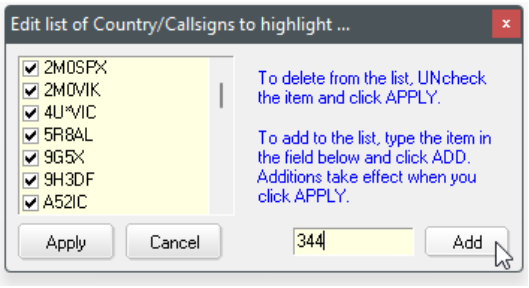
The audio alerts sound whenever[211] the relevant stations are spotted on DX cluster *or* appear on the UDP BandMap *e.g.* if P5DX ever shows up on FT# and is decoded by our stations, our 'New Country' or 'New Mode Country' alarm bells will jangle merrily[212], even before someone spots him on the DX cluster network.

Hinson tip: reserve the most intrusive, annoying, loud sounds for relatively rare, important events, using more subtle, quiet, discreet sounds for commonplace trivia. Gentle ticks and quiet dings or tones can still inform you about band-fillers *etc*. without constantly annoying others in your household, while something deliberately ear-catching such as loud jangly bells, sirens or a barking rottweiler might be appropriate for an all-time-new-one drop-everything *urgent* alert such as a New Country – a DXCC entity you have not yet worked on any band or mode[213]. Start off with just a few – the most important alerts – and give it a while before updating your selection, so you have time to get used to them. If you try setting too many alerts at once, trust me, you'll be confused.



The upper section of the form ▲ has 3 buttons to edit (define) the *specific* alerts that interest you.

1. <**Edit Callsign Alert list**> lets you sound alerts when particular callsigns are spotted or decoded ▶ Simply type a callsign into the blank rectangle, then click <**Add**> to add it to the list. Use asterisks for wildcards. To remove a callsign from the alert list, click to un-tick it then click <**Apply**> to save the settings when you are done, or click <**Cancel**> or the corner X if you chicken-out.



[210] In the Logger32 Groups.IO files area is "Logger Audio Alerts.zip", a set of spoken alerts as .WAV files.

[211] To avoid transmitting them on-air, audio alerts are silenced while you are transmitting through the sound card window … although you should really be using a dedicated sound card *just* for the radio, separate from the one used for alerting *etc*. so there is no conflict.

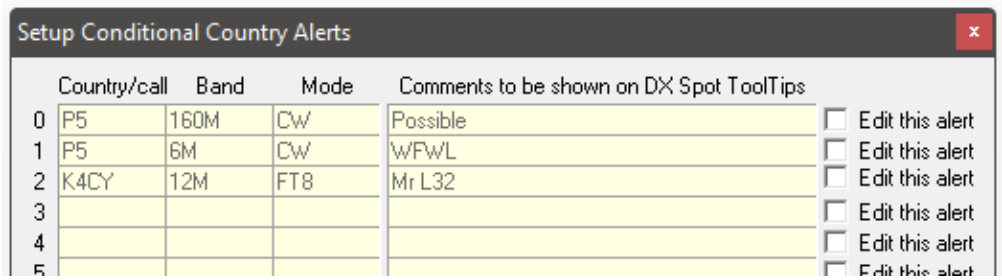
[212] If several alerting conditions coincide, Logger32 only generates a single alert for the highest priority condition. The system determines priorities using a secret algorithm known only to K4CY's shrink.

[213] Clicking the top left corner cell of the worked/confirmed table toggles back and forth between "ALL" (meaning *all time* new ones) and the current year (new ones just this year). The audio alerts and visual highlighting of DX spots follow suit.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: add your own callsign to the callsign alerting list in order to be notified whenever someone spots you. Maybe you have been calling CQ for a while without replies and are just about to give up in disgust when a kind soul spots you, hoping to generate some action …

2. <**Edit call/band/mode list**> lets you set up to 20 alerts for combinations of a particular country or callsign appearing on a given band and/or mode ▼

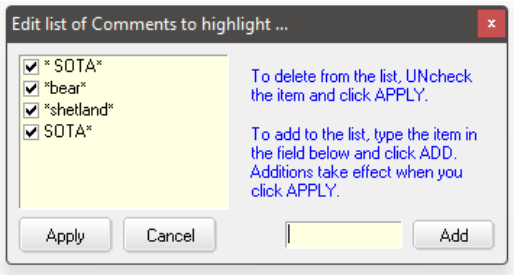


◀ Click to tick <**Edit this alert**> in order to add, modify or delete any row.

When you click <**Apply**> at the bottom of this form, the alert settings are written out to the file *C:\Logger32\Conditional Country Alerts.txt*[214] to be read-in when Logger32 next starts-up.

Hinson tip: *Conditional Country Alerts.txt* is a plain ASCII text file that you can edit directly with MS Notepad or WordPad, or TED Notepad if you prefer the quick approach. Create and <**Apply**> at least one entry in Logger32 first to generate the basic file, then in your text editor copy the format and structure of any existing entries *i.e*. information from each of the four columns in the table, separated by vertical bars, with each table row on a separate line.

3. Use <**Edit Comment Alert list**> to configure alerts when various interesting DX spot comments appear, for example if someone kindly spots stations from Shetland, North Cook or Bear islands with those words in the spot comments (since their prefixes tend not to indicate those particular locations) ▶



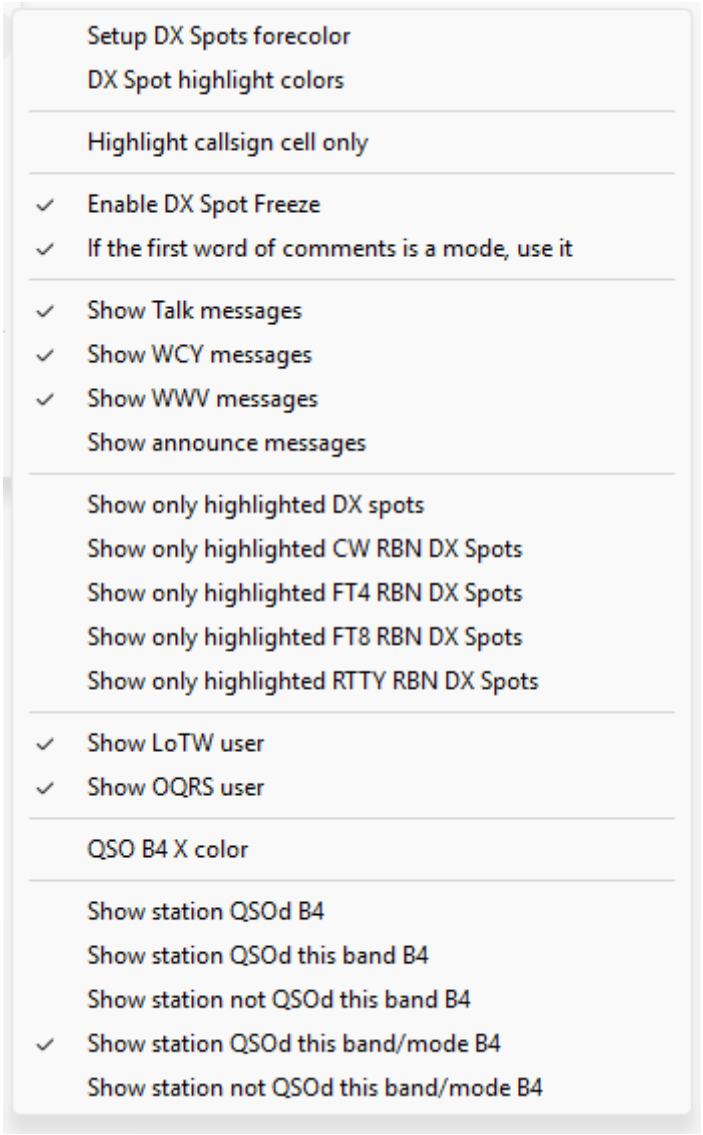
Hinson tip: some events such as the DX Marathon use an extended country list *i.e*. 340 current DXCC entities *plus* five extras from the **W**orked **A**ll **E**urope list *and* African Italy. Logger32 lets us configure 'new one' alerts for the extended list of 346, including stations using special callsigns, so we shouldn't miss out if they are spotted – miss out on the opportunities, that is: we still need to work and log them to claim the points!

[214] … and removed from the *C:\Logger32\Logger32.INI* file where they were stored prior to v4.309, so now the same alerts are active regardless of which configurations you use.

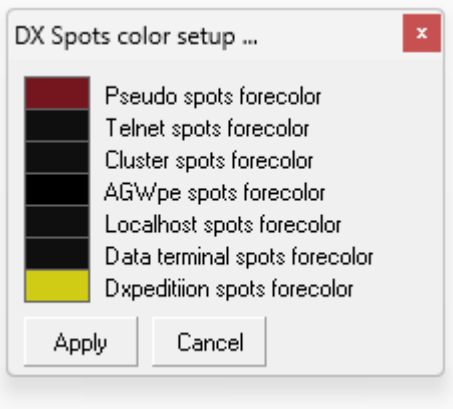
Copyright © 2026 Robert C. Furzer K4CY

14.2.3 DX spots right-click *Setup* ⇨ *Appearance* submenus

Right-click the DX Spots pane or a BandMap then click **Setup ⇨ Appearance** to open this *extensive* set of spot options ▶



- **Setup DX Spots forecolor**: opens yet another submenu ▼



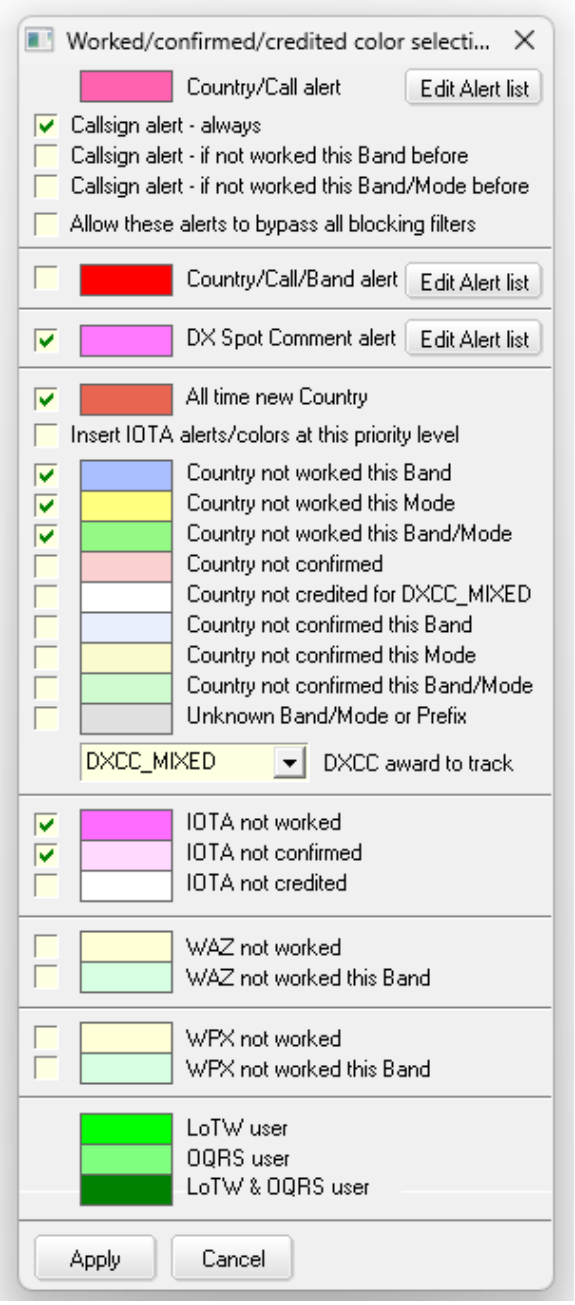
  - **Pseudo spots forecolor**: if you are in the habit of bookmarking interesting stations privately by clicking <**Ctrl**+**B**> to generate pseudo-spots, you can give them a distinctive text color so they stand out from regular public DX spots in the DX Spots pane and on your BandMaps.
  - **Telnet|Cluster|AGWpe|Local Host|Data terminal spots forecolor**: similarly, if you have connections to more than one DX cluster through different tabs in the DX cluster pane, you can distinguish the DX spots visually using different text colors for spots arriving through each tab's connection. For instance, *special* DX spots received from club members via your club's private DX cluster through one tab can be shown in a distinctive color (red in my case), whereas *ordinary* spots circulating on the public DX cluster network and received through another tab are shown in a plain/dull color (black or blue for me).
  - **DXpedition spots forecolor**: this sets the color of the callsign of known DXpedition stations when they are spotted – that is, stations on the DXpedition timeline.

> **Hinson tip**: DXpeditions are more likely than most to be on your wanted lists, hence they are more likely to be highlighted with a distinctive background color than ordinary DX spots. Bear this in mind when selecting the forecolor: if it lacks contrast against your background highlighting, it will be difficult to read and you might miss your opportunity for a new one.

  - Click **<Apply>** to save the settings as shown, or <**Cancel**> to discard recent changes and revert to the settings before we started meddling.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: the colors defined here apply to DX spots displayed in the DX Spots pane and most BandMaps (apart from the UDP BandMap). It can be tricky to find a combination of colors that is legible both with and without the highlighting of new ones. Good luck Jim.



◀ **DX spot highlight colors:** this is where we setup the highlighting and visual alerts for various 'new ones', plus those colored blobs showing LoTW users *etc*., aside from any audio alerts.

Again, there are *lots* of things you can configure here but take care: it's all too easy to set up a kaleidoscope of garish colors with highlights for things you don't really care about, distracting you from your operating priorities.

So, a more sensible approach is to start by figuring out your operating and award-chasing priorities. Pick out the *few* things that really matter to you, and come up with a color scheme to suit (*e.g.* a gradation from subtle pastels for the lowest priorities through to intense "day-glo" colors for your top priorities). Both the colors and intensities can be varied using the color picker (click the colored rectangles).

Then configure Logger32 to your scheme and try it out for a while. Give it a good run for at least a week or two before adjusting anything. You will discover the value of not having too many distinctive colors (since they are used all over the place in Logger32, and can get very confusing on busier windows such as the BandMaps). If your scheme is complex and confusing in practice, it can hinder rather than help your activities … so review, simplify and redesign. Lather, rinse, repeat.

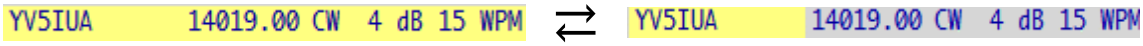
Hinson tip: believe me, after a few years of this you'll *eventually* be reasonably happy with it - or just exhausted by the quest for perfection! Be sure to backup your lovely *.INI* files containing all these configuration settings from time to time, unless you relish the thought of slogging through the laborious configuration process again, from scratch, after a disk failure or some other utter disaster.

- Most items on that menu are self-explanatory.
- Tick <**Allow these alerts to bypass all blocking filters**> so Logger32 pokes little holes in your normal DX spot blocking filters to *pass* spots for 'new ones'. For instance, if you are routinely blocking all spots from spotters outside your continent, but someone on the far side spots a rare DX station that you desperately need, ticking this option means you *will* be alerted anyway. You may or may not work him but, hey, that's DXing.

Copyright © 2026 Robert C. Furzer K4CY

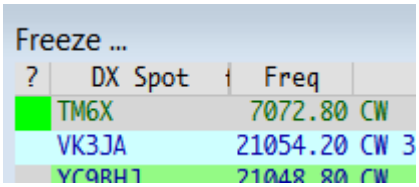
- Tick <**Insert IOTA alerts/colors at this priority level**> if new IOTAs are as just important to you as all-time new countries.

- **Highlight callsign cell only**: whereas normally the whole DX spot line has a brightly-colored background for new ones▼, the callsign column alone ▼ may be highlighted instead.



Hinson tip: this has the advantage of retaining the zebra stripes (alternating background colors) for the remaining spot information, making it easier to read along each line, quickly.

- **Enable DX spot Freeze**: mousing, wheeling or scrolling in the DX spot pane freezes incoming spots for a short while - long enough, hopefully, to browse the history, find an interesting spot and click it. The window caption ▶ shows Freeze …, the dots disappearing rapidly as it thaws. While the pane is frozen, incoming spots are buffered and appear as soon as it thaws. Click the mouse in the pane to thaw it instantly.



- **If the first word of comments is a mode, use it**: a DX spot for an FT8 fox-and-hound station is outside the normal FT8 band segment, so the highlight colors and tooltips may not indicate FT8 - maybe RTTY or something depending on the bandplan settings. However, if the first word of the comments field is FT4, FT8, FOX or HOUND, and if you have this option checked, Logger32 disregards the mode from the bandplan, using the mode from the spot comment instead.

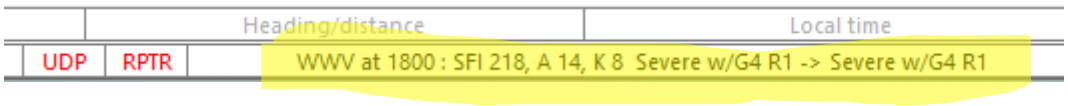
  Clicking a DX spot for an FT8 station operating in the RTTY band segment should result in:

  - The DX spot window's tooltips and status bar showing FT8 wants/needs.
  - The log entry pane mode field showing FT8 (even if WSJT-X/JTDX is not running).

  If there is a bandplan entry for FT8 on this band, the radio switches to the appropriate radio mode for FT8. Otherwise, the radio should switch modes using the first FT8 entry it finds in the bandplan. If there is no FT8 entry anywhere in the bandplan, the radio defaults to SSB.

- **Show Talk messages**: DX cluster "talk" messages addressed to you appear in the DX spot pane.

- **Show WCY|WWV messages**: display the solar/geomagnetic messages circulated on the DX cluster network in the DX spot window, *and* update the figures shown on the status bar ▼



Hinson tip: tick *both* options to use the latest data received from *either* WWV *or* WCY.

Each WCY/WWV message states:

- The source and time (in UTC) of the circulated report.
- The **S**olar **F**lux **I**ndex - a measure of radio noise emitted from the sun at a wavelength of 10.7cm (a 100 MHz-wide band centered on 2800 MHz) averaged over an hour and expressed in **s**olar **f**lux **u**nits[215].

[215] As you are no doubt aware, 1 sfu is, of course, 10^{-22} W/m^2/Hz, whereas stfu is short for something decidedly nonscientific, cutting and very rude.

Copyright © 2026 Robert C. Furzer K4CY

> **Hinson tip**: SFI values above 100 generally indicate visible sunspots and good HF conditions, whereas levels down to about 50 imply few if any sunspots and poor propagation on the high HF bands. At the top of the ~11-year solar cycle (solar maximum), SFI values peak around 300, the highly ionized ionosphere allowing worldwide propagation on 15, 12, 10 and 6m, even for those of us running QRP to crude antennas, in between the frequent and often severe solar flares and proton storms anyway.

  - The **A**-index – a linear measure of geomagnetic activity indicating disturbances in the magnetosphere. Values above about 15 suggest poor HF communications, particularly over polar paths with auroral displays likely.
  - The **K**-index – a logarithmic (exponential-scale) measure of geomagnetic activity.

> **Hinson tip**: the K-index resembles Richter units for earthquakes. Rather like a K-index of 5, a strength 5 earthquake is noticeable but benign, whereas above strength 7 the damage can be substantial. A strong geomagnetic storm at a K index of 8 wipes out the high bands, leaving few if any signals. A K index of 9 indicates *catastrophic* levels of geomagnetic disturbance, potentially disrupting power grids and satellite communications.

  - **WWV messages** also reveal current and predicted solar activity. Significant events such as solar flares and **C**oronal **M**ass **E**jections (sun farts) can cause blast shockwaves followed by streams of energetic particles that perturb the Earth's magnetosphere and ionosphere, known as 'solar storms'.
  - **WCY messages** also display the **R**elative **S**un**S**pot **N**umber (a.k.a. Zurich or Wolf number), indicating the number of individual sun spots plus those in groups (where a group counts as ten spots), modified by a fudge-factor intended to even-out observers' differing instruments and interpretations.

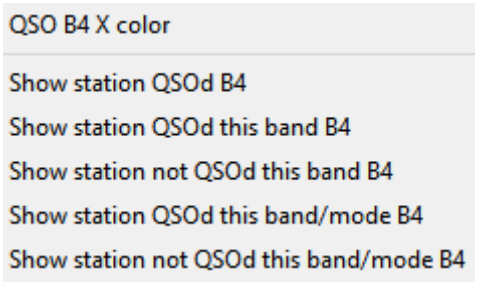
- **Show announce messages**: display DX cluster **ann**ouncements (broadcasts to all connected users) in the DX Spots pane.
- **Show only highlighted DX spots**: when the DX cluster networks are abuzz with DX spots (*e.g.* on busy weekends, during major contests or when DXpeditioners are *everywhere*), only showing 'new ones' avoids them being crowded-out by hordes of humdrum spots[216]. This option affects the DX Spots pane *and* the BandMaps.
- **Show only highlighted CW|FT4|FT8|RTTY RBN spots**: these four options let us filter out and ignore *non*-highlighted spots from the **R**everse **B**eacon **N**etwork for the specific modes. They are ideal if, for example, you are a CW or sideband nut who might be tempted onto the digital modes occasionally, but *only* in order to chase a new one (*e.g.* a band-filler) before scurrying back to your normal mode, your happy place.
- **Show LoTW|OQRS user**: indicate whether spotted stations are known to use LoTW and/or Club Log's OQRS (**O**nline **Q**SL **R**equest **S**ystem) by showing colored blobs in the (leftmost) column of the DX spot pane, and on the BandMaps.
- **QSO B4 X color**: Logger32 asks, politely, "What color shall I paint the X indicating a station you have worked before?" and shows you the usual color palette from which to pick.

[216] It is analogous to choosing to see "Mults only" rather than "Mults & Qs" in N1MM+.

Copyright © 2026 Robert C. Furzer K4CY

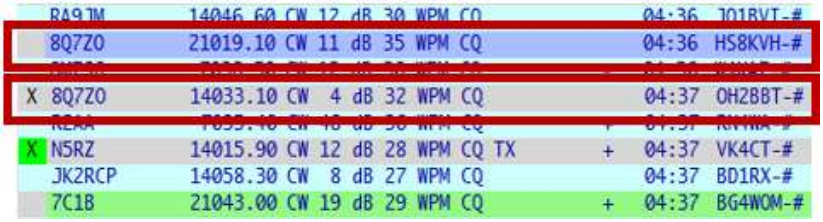
- **Show station QSOd before**: if the station has been worked and logged already, an **X** (in your choice of color) marks the row in the leftmost "?" column[217]. You can mark:

  1. Stations you have worked before on *any* band and mode (not necessarily those on which they are now been spotted);
  2. Stations you have, or have not, worked before on the spotted band; or
  3. Stations you have, or have not, worked before on the spotted band *and* mode.



  4. None: un-tick all the options to leave the "?" column empty, unused, vacant.

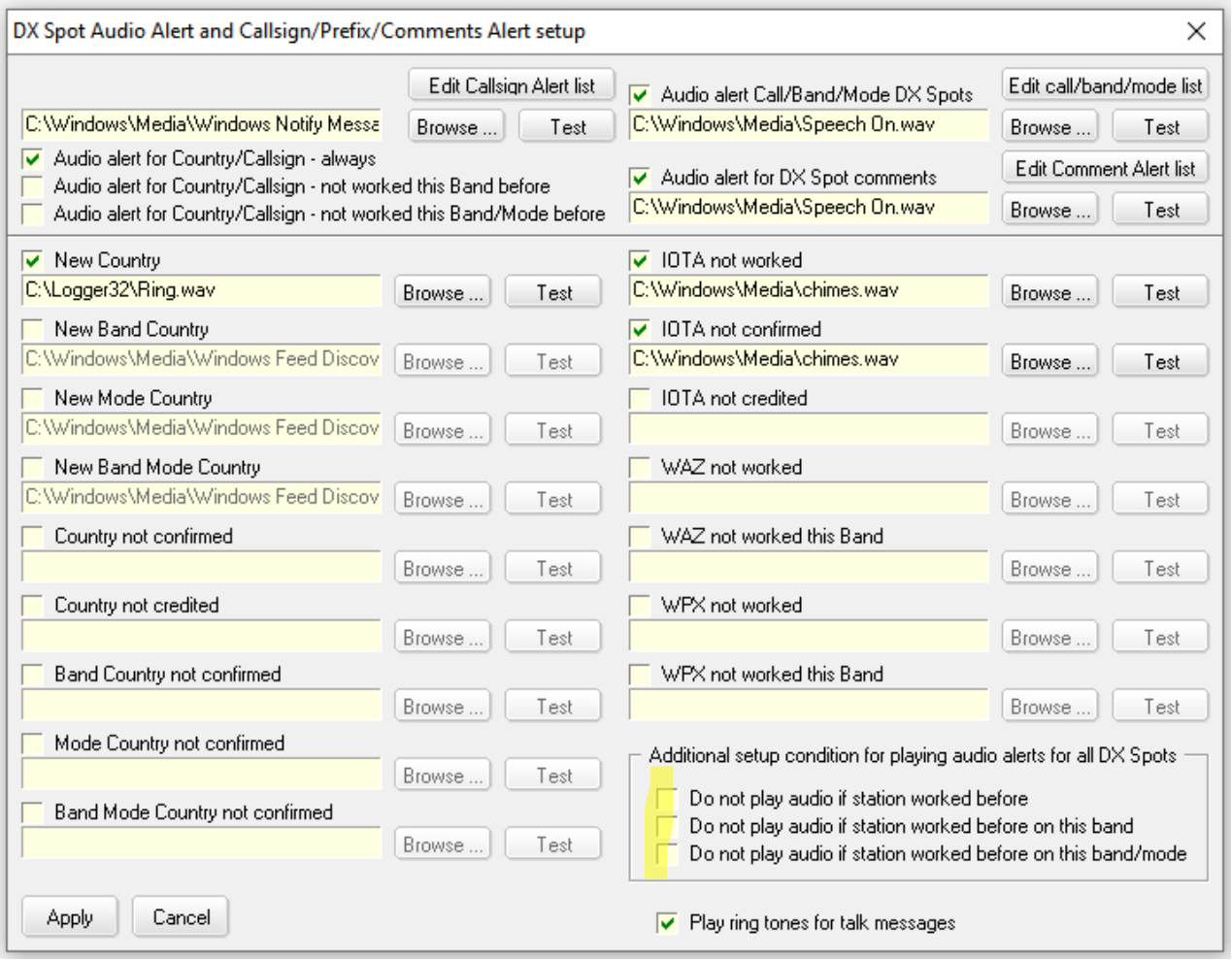
In this example, the **X** tells me that I have worked 8Q7ZO on 20m but not on 15m … where the blue background means it would be a new DXCC ▶



[217] Logger32 uses your Bands & Modes table to determine the mode a spotted station is *probably* using on the subband/frequency. In major contests, pileups and other situations where band plans are widely disregarded, the **X** markers might be missing or incorrectly shown. **X**'s are shown on the BandMaps too.

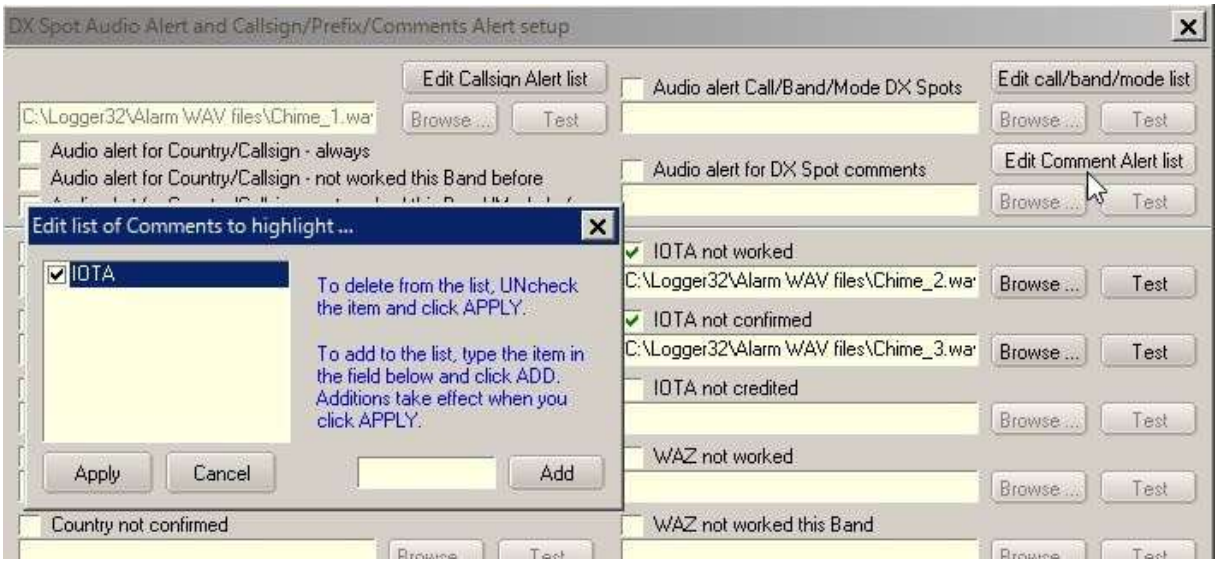
Copyright © 2026 Robert C. Furzer K4CY

- Audio alerts can be reduced by ticking one of the options listed in the lower right-hand corner of the alerts configuration form ▼



Most audio alerts *except* for callsign and comment alerts will be suppressed. These are (generally) overriding priorities so their alerts continue.

- **Audio alert for DX spot comment**: if you want to be alerted whenever DX spot comments mention IOTA, POTA, SOTA, special, Shetland, North Cook, Pyongyang, Bear Island *etc*. click the <**Edit Comment Alert list**> button and add the relevant strings ▼



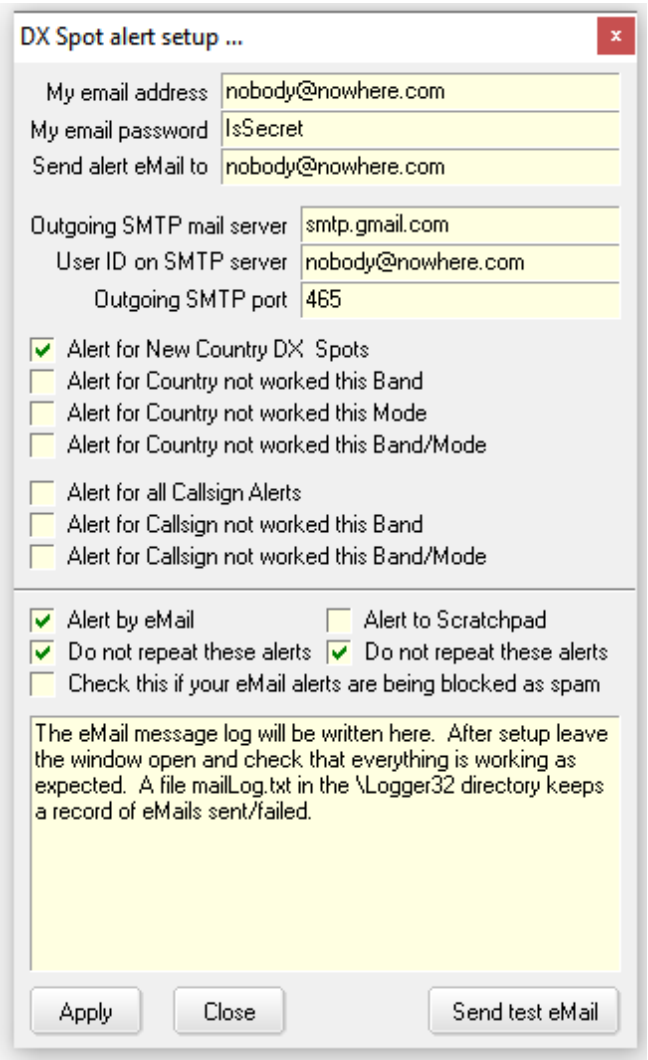
Copyright © 2026 Robert C. Furzer K4CY

- **Audio alert for IOTA**: three IOTA alert categories are available: (1) island reference not yet worked; (2) worked, but not yet confirmed; or (3) worked and confirmed, but not yet credited for the IOTA award. If it is worked, confirmed *and* credited, there's no need for an alert.
- **Audio alert for WPX**: two WPX alert categories are available: (1) prefix never worked before on any band; or (2) prefix worked but not yet on this band.
- **Additional setup condition for playing audio alert for all DX spots**: if you are getting too many audio alerts and annoying the neighbors, these three options can reduce the aggravation.
- **Play ring tones for talk messages**: tick this to play *RING.WAV* on your default audio system when someone sends you a DX Cluster talkie.

14.2.4 DX alerting via email and text messages or scratchpad

Logger32 can send you an email or (using an email-to-SMS service) a text message whenever selected DX stations are spotted on DXcluster.

Open the configuration form ▼ by right-clicking the DX Spots pane, then clicking **Setup ⇨ Setup DX spot alerts (eMail/SMS/Scratchpad)**.



Since Logger32 does not support TLS for email alerting, you'll need to find an outgoing email service that still accepts SMTP connections on **port 465** (for SSL) or **port 25** (unencrypted) – assuming you are not fussed about someone snooping on your stream of email alerts in transit, that is. Will you accept the risk?

Note: **Gmail** users must enable 2-step verification on their Google accounts *and then* create an app password to get these alerts.

Click to tick <**Alert by eMail**> and/or <**Alert to Scratchpad**> (sending the spots to the Scratchpad, ready to chase at your leisure).

Decide which spots/alerts you want to receive:

- **Alert for New Country DX Spots|Country not worked this Band|Mode|Band/Mode** [218] sends the specified 'new country' spots.
- **Alert for all Callsign Alerts|Callsign not worked this Band|Callsign not worked this Band/Mode** sends the callsign alerts.

Click <**Apply**> to save the alerting configuration and set it in motion, keeping an eye on the diagnostic section at the bottom of

[218] "Band/Mode" means this band *and* mode. If you have already worked P5DX on 160m RTTY, do you really want to be notified *every* time he is spotted on 160m RTTY? 160m CW, now that's a different matter.

Copyright © 2026 Robert C. Furzer K4CY

the form as well as your email system. Email alerts are shown in the lower part of the configuration form as they are sent … or fail. They are also written to an audit trail at *C:\Logger32\mailLog.txt*

If your selected DX stations are rarely spotted, click <**Send test eMail**> to send a fake alert right away, hopefully proving that the current configuration works.

Even if you tick <**Do not repeat these alerts>** to block duplicates, emails containing alerts for relatively common DX may be consistent and frequent enough to trip anti-spam controls. If the mail server or email software categorizes the alerts as spam, tick <**Check this if your eMail alerts are being blocked as spam**> to add some junk text to the messages that – hopefully – evades the anti-spam control ▼

```
Some free text to fake out the spam blockers. Here is the DX Spot:

DX de F1TRE:      7086.7  N6AR         WPX RTTY                       0312Z JN37
```

Finally, when you are happy that the email alerting is working OK, click <**Close**>.

Hinson tip: email alerting settings are independent of the credited/confirmed/worked color on/off settings and the DX spot audio alert on/off setting … but the spot filtering is the same. Get your DX spot highlighting working sweetly before enabling email alerts.

Hinson tip: HamAlert is a flexible alternative. It can hook into your Club Log account to identify 'new ones' just for you, sending you alert messages by email, SMS or HTTP GET/POST.

Email and SMS alerts of course rely on Logger32 running 24/7, especially if you want to be alerted when you are not in the shack. This is where Parallel Logging comes in useful for me. My 2nd PC is a virtual PC with my entire logbook on it. That PC is connected to the cluster 24/7. As it is a Virtual Machine it can stay on 24/7, headless (no screen or keyboard). Parallel logging keeps the logbook up-to-date. The VM is configured to use a commercial Email-to-SMS service and my alerts are sent as texts to my mobile phone wherever I am. The company I use for the Email-to-SMS service charge just under £0.03 per SMS. They give £2 of free credit when you register with them and you can put as little as £10 credit on when that runs out: your £10 will last quite a while!

Rob G4UJS

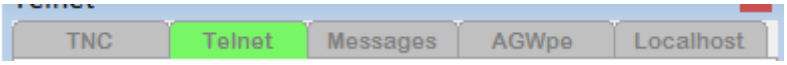
Copyright © 2026 Robert C. Furzer K4CY

14.2.5 DX spot blocking filters

Using the configuration options described below, Logger32 can apply a set of rules to block (filter out, hide, drop, ignore) DX spots that you don't want to see or use, *i.e.* spots that were:

- Posted by stations in a given continent (*e.g.* block all spots *from* spotters in North America).
- For DX stations in a given continent (*e.g.* block all spots *of* North American stations).
- On a given band or mode (*e.g.* block topband or SSTV spots).
- From a given spotter (*e.g.* block inaccurate/inept/excessive spotters).
- Separately per connection tab if you have multiple DX cluster connections (*e.g.* pass everything on the TNC or Localhost tabs to keep an eye on your club's private DX cluster or your own Skimmer, but block unhelpful stuff from the public DX cluster arriving through the Telnet tab).

  As a reminder, these are the default names for the DX cluster tabs ▼ until connected, when they take on the cluster names.



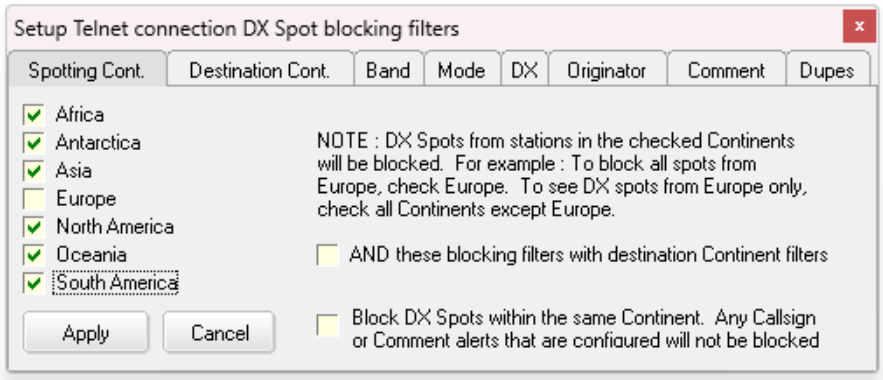
Logger32 can bend the rules (passing spots that would otherwise be blocked) **if** those spots are configured to trigger your country/call alerts, and **if** you select the option. For instance, imagine you don't enjoy using FT# so your system blocks all FT# spots normally: if P5DX was spotted on FT8, and if P5* (or, better still, DXCC entity code 344) was listed in the country/call alerts, seeing that highlighted spot might convince you to break your own rule and give FT8 a go just this once in order to fill an *ultra*-rare slot and triumphantly carve another notch in your shack desk.

Hinson tip: the blocking filters are applied to *DX spots*, specifically. Despite looking much the same as regular BandMaps, callsigns shown on your UDP BandMap are from decodes, *not* true DX spots. However, there is an option on the UDP BandMap's menu **Config ⇨ Show cherry-picking blocked callsigns, text of DXCC #** to apply the same DX spot blocking filters there too, if you wish.

To configure the filters, right-click the DX Spots pane, then click **Setup ⇨ DX Spot Blocking/Pass filters**. These are all the DX spot blocking filter tabs and their options:

- **Spotting Cont. tab**: click to choose the continent/s of spotters from whom you do *not* want to see any spots.

  With these 6 ticks blocking all others, *only* spots posted *by* European stations are passed ▶



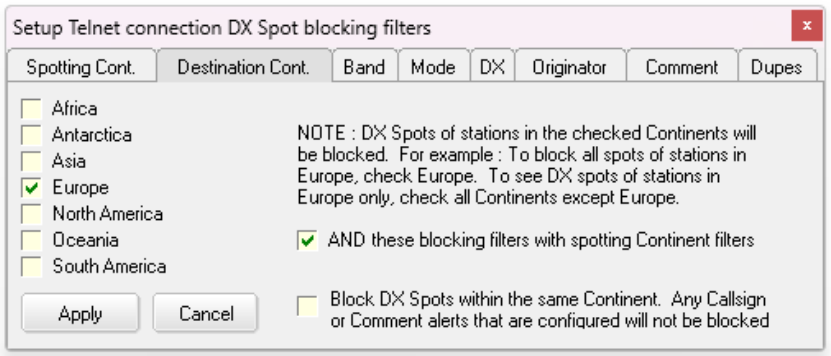
  <AND these blocking filters with destination Continent filters> does a logical **AND** with the **Destination Continent** tab's filters, linking the two sets of filters together. So, with the 6 ticks shown above and a single tick for Europe on Destination Cont., only spots posted *by* Europeans *for* DX stations *outside* Europe are passed. No longer will you see Portuguese spotting Poles, Finns spotting the French, or Belgians spotting Belgians. If this option is not ticked, Logger32 applies a logical **OR** for the two filters: spots are blocked if they are caught by *either* or *both* tab's filters.

Copyright © 2026 Robert C. Furzer K4CY

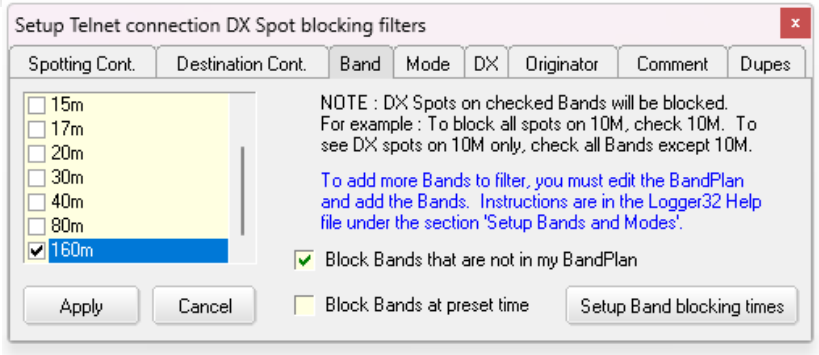
<Block DX spots within the same Continent …> achieves the same thing but for all continents *e.g.* it will block spots *for* European stations posted *by* Europeans AND spots *for* Africans *by* Africans *etc*. Notice that (with this option) spots for any stations for which you have explicitly configured Callsign or Comment alerts are passed, even if they would otherwise be blocked *e.g.* if you have an alert for C3 Andorran stations, any C3 spots by Europeans will get through the EU-blocking filter and may sound your alert – you won't miss out on them!

After configuring the filters on one or more tabs, click <**Apply**> to enable them all and close the form.

- **Destination Cont. tab**: works the in much same way as Spotting Cont. except that it applies to the spotted DX stations' continents. If you don't want to see spots for DX stations in, say, Europe, tick Europe on this tab ▶



- **Band tab**: similar again, this tab lets you block all spots on selected bands ▶



  If the particular bands you want to block are not listed, you can add them individually to your Bands & Modes table[219] and then tick them, or simply tick <**Block Bands that are not in my BandPlan**>.

Hinson tip: better still, apply filtering at source by configuring filters for the relevant bands directly on whichever DX cluster/s you are using. This has the advantage of reducing unnecessary network traffic and reducing the load on your CPU caused by Logger32 identifying and dropping unwanted spots. If the cluster band-filtering commands are too tricky for you, VE7CC's Cluster User software makes selecting/deselecting bands as easy as ticking or unticking bands on a form and clicking the <**Tell Cluster**> button. Point-n-click.

[219] If you do *not* want your award statistics to show the added bands and/or modes (*i.e.* you are *only* adding them for DX spot filtering purposes), put "N" in the Statistics column of the Bands & Modes table for the new entries.

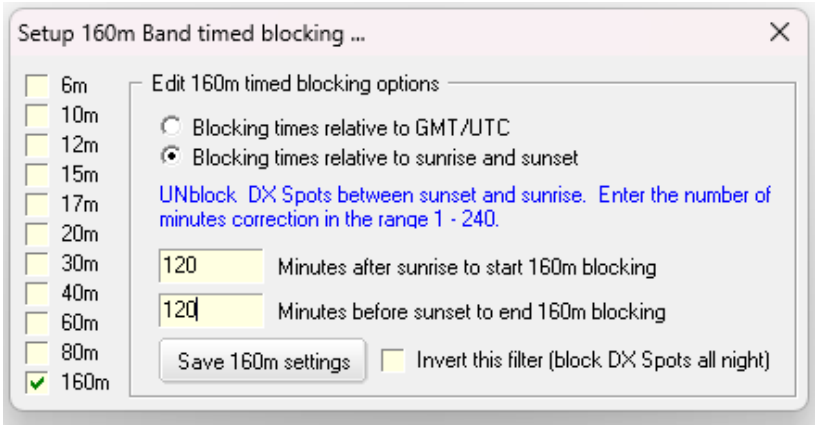
Copyright © 2026 Robert C. Furzer K4CY

<**Setup Band blocking times**> lets you, for instance, block topband spots during the middle of the day when solar absorption makes the band useless for DXing.



◀ First tick the band to enable the options, then set the times either in UTC or relative to your sunrise and sunset (which Logger32 calculates, every day).

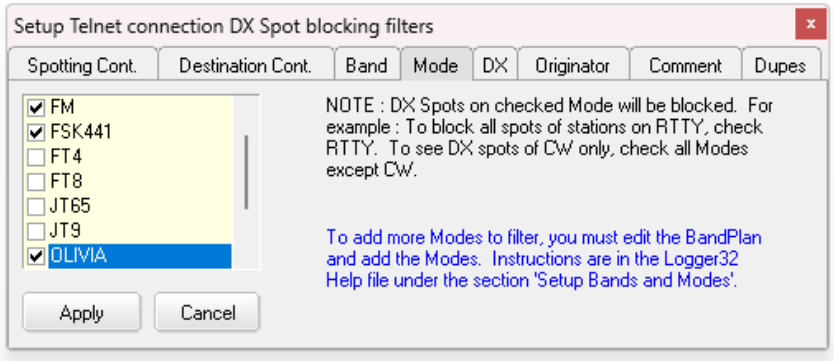
Here I am blocking topband spots from two hours after sun-up until two hours before sun-down ▼



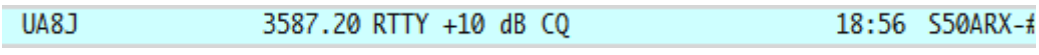
Click <**Save [**band**] settings**> to save the configuration for the present band and then, if you wish, repat the process to configure timed blocks for other bands too.

<**Invert this filter (block DX spots all night)**> might be useful if you are a nocturnal topband DXer: you could block distracting nighttime spots on all the other bands in order to focus solely on 160m all night long if you like.

- **Mode tab**: lets you block DX spots for modes you don't use or simply don't care about right now ▶

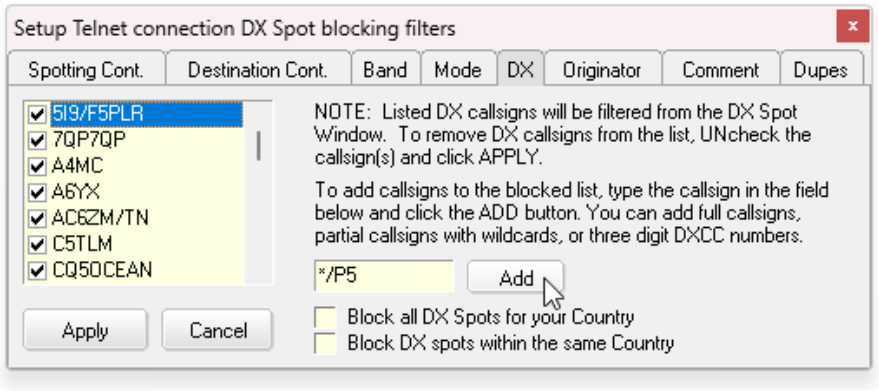


You can add modes to the list by editing the Bands & Modes table that Logger32 uses to determine the mode for any DX spots that do not specify the mode at the start of the spot comment, like this ▼



Copyright © 2026 Robert C. Furzer K4CY

- **DX tab**: Logger32's filter has a fine enough mesh to block spots for individual DX stations, for instance certain busted or misleading callsigns that circulate repeatedly on the DX cluster network[220] ▶



  Type a callsign into the white box then click <**Add**> to add it to the blocked list. Click a callsign in the list to un-tick it (it will be removed from the list when you click <**Apply**> on any of the blocking filter tabs).

  You can use one or more wildcard asterisks in the callsign field *e.g.:*

  - */B blocks stations *spotted* with the /B suffix (whether or not that is part of their transmitted call: Logger32 adds the suffix to spots for stations it identifies as beacons).
  - */MM blocks spots for maritime mobiles who identify correctly as such (not all do).
  - K4CY* blocks DX spots for any callsign *starting* with K4 (passing ZS/K4CY *etc*.).
  - *K4CY* blocks all DX spots for callsigns *containing* the contiguous sequence "K4CY" such as P5/K4CY and K4CYA.
  - K*CY blocks spots for 4-character callsigns starting with a K and ending with a CY – such as K1CY, K4CY and KZCY. In this case, the asterisk is replaced with a single character.
  - AA**XX blocks/passes/highlights callsigns/comments that begin with AA and end with XX.

  You can also enter the 3-digit numeric "Entity Code" from ARRL's DXCC list to block spots for DX stations in an entire DXCC country, without having to specify all the prefix variants (*e.g.* entity 291 means mainland USA where K, W, N and AA-AK prefixes are normally used, with several exceptions).

Hinson tip: the same filtering logic and syntax applies to the highlighting of wanted callsigns.

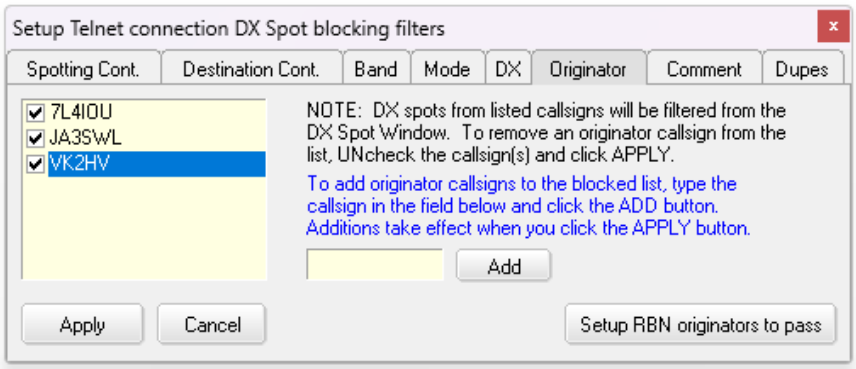
  <**Block all DX spots for your Country**> blocks spots for DX stations in the same country (DXCC entity) as you. Logger32 determines your home country from the prefix of your callsign, assuming that you are sensibly using your callsign as the Operator. If you doggedly insist on using something else (such as "Remote" or "Bob") as the Operator, don't be surprised if this function can't figure out where you are. It is clever, but not *that* clever.

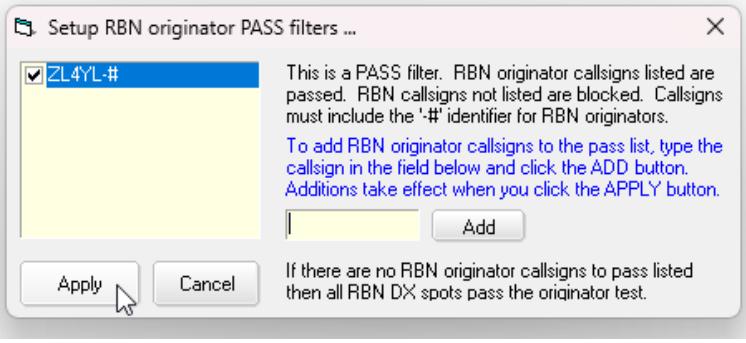
  <**Block DX spots within the same Country**> deals with Italians spotting other Italians, egomaniac self-spotters *etc*.

[220] Some inept operators evidently have trouble sending their own callsigns accurately, or have slow TX/RX changeover relays that truncate the first character, or *deliberately* send some exotic but spurious version of their own callsign occasionally in order to trigger alerts and draw DXers to them, like moths to a flame.

Copyright © 2026 Robert C. Furzer K4CY

- **Originator tab**: is handy to block spots from inept spotters, cluster trolls, prolific spotters and other such annoyances ▶

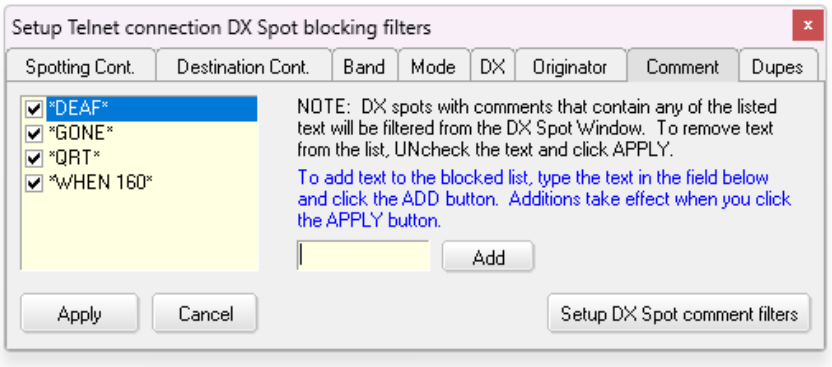




◀ **<Setup RBN originators to pass>** lets you *pass* [221] spots posted by specific RBN/Skimmer nodes (*e.g.* those located in your general area, and maybe a handful of others around the world), **while blocking spots posted by all other RBN/Skimmer nodes**. Callsigns are *shown* with "-#" at the end but you enter just the base callsigns without that. Un-tick the list completely to pass everything.

- **Comment tab**: if inane and unhelpful comments on DX spots annoy you, this is your chance to fight back.

  Type the inane word or phrase into the white box then click <**Add**> ▶



  Comment filtering is case-insensitive, spaces are allowed and wildcards may be used[222], but, commas, may, not. Wildcarding syntax works like this:

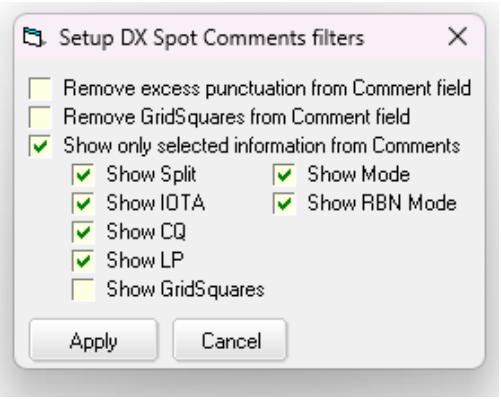
  - **text** blocks spots with comments consisting *entirely* of the text (regardless of case).
  - ***text** blocks spots with comments *ending* with the text, such as Blah blah **tEXt**
  - **text*** blocks spots with comments *beginning* with the text, such as **Text** blah blah
  - ***text*** blocks spots with comments containing the text as a contiguous string at any point – start, middle or end – regardless of case and any spaces or punctuation immediately before or after. Spots with the comment Blah **TeXt** blah for instance would be blocked, also Blah**text** blah and blah **teXT**!! Blah.

[221] In contrast to the rest of the DX spot filtering, spots posted by the listed Skimmer callsigns are passed through, not filtered out. Logger32 pokes holes in the filter mesh for these Skimmers.

[222] If you specify the * symbol on its own, *all* DX spots will be filtered out, letting nothing through as if your DX cluster connection was broken. Doh!

Copyright © 2026 Robert C. Furzer K4CY

▼ Click <**Setup DX spot comment filters**> for yet more filtering options[223]:

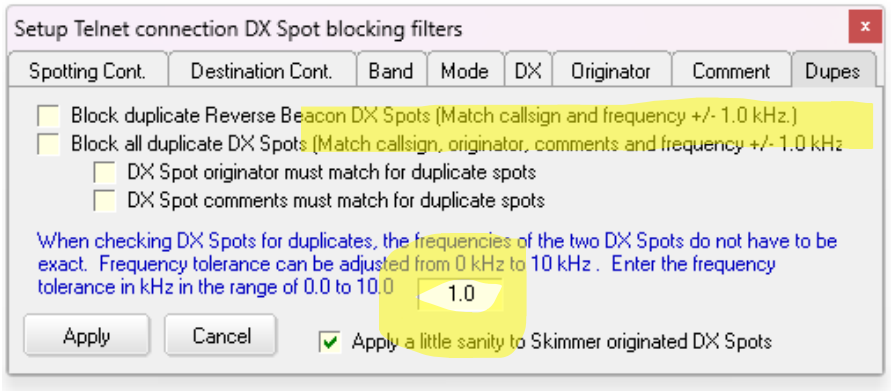


Remove excess punctuation from Comment field: trims down (((((,))))), !!!!!, ????? *etc*. for the grammar police among us[224]. CQ CQ CQ becomes CQ.

Remove GridSquares from Comment field: does what it says on the tin.

Show only selected information from Comments: if selected, the options above are disabled leaving you to specify what kinds things you are willing to see displayed in spot comments. Tick the ones you want.

- **Dupes tab**: today's DX cluster and **R**everse **B**eacon **N**etworks are being hosed continuously with CW and digimode spots[225] generated and circulated indiscriminately by robotic Skimmers. This tab lets you reduce[226] unnecessary duplicate or repetitive DX spots ▲ <**Block duplicate Reverse Beacon DX spots**> seems the obvious thing to do … except that RBN's Skimmers often receive and spot the exact same DX at slightly different times and frequencies. Due to calibration errors, the spotted frequencies can vary a little, while the spot comments typically indicate the signal-to-noise level at each Skimmer, which naturally varies. Logger32 is clever enough to identify and block genuine dupes, without blocking multiple spots for the same callsign on markedly different frequencies or modes (*e.g.* when a large DXpedition or special event station runs CW and digimode pileups simultaneously on the same band). You can define the frequency tolerance value in the white box[227]: multiple spots for the same DX station on frequencies within the tolerance range are treated as duplicates.



Hinson tip: small tolerance values may help you keep up with a DX station jumping around to out-fox an unruly pileup, but poorly-calibrated Skimmers may generate near-duplicate spots that are treated as different spots. It's a tradeoff. By all means experiment with the setting.

[223] Whereas the DX spot filtering function mostly concerns passing or blocking spots, *these* settings affect how spot comments *appear* in the DX spot window, BandMap tooltips *etc*.

[224] The grammar police can't help but notice the Spurious Capitalization littered throughout Logger32. Too bad! Suck It Up or by all means submit a claim under Logger32's full money-back guarantee.

[225] Just wait until Skimmer technology advances to the point of decoding phonetics on speech modes. VFO knobs and headphones will become redundant, microphones too for DXers with DVKs.

[226] Duplicate spots are not simply discarded. They overwrite older entries in the DX spot window. As the DX spot window is sorted chronologically, the most recent DX spots generally remain visible.

[227] Type a value, then click <**Apply**>. The *next* time you open this tab, the top two options will show it.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: you may *want* to see duplicate spots for a given DX station on a quiet band, where the station is patiently CQing for some time on the same frequency. Once the original spot times-out and is removed from the BandMap, further spots for the same guy are ignored and he does not reappear on the BandMap until/unless he QSYs beyond your QSY tolerance value.

<**Block all duplicate DX spots**>[228] applies the same approach to human as well as Skimmer spots, since some over-excited spotters just cannot resist spotting and re-spotting the same DX station, over and over. Worse still are the inept spotters who fail to copy or type DX callsigns correctly, leading to the circulation of busted spots like the game Chinese whispers.

<**Apply a little sanity to Skimmer originated DX spots**> does three things:

- In order to sift out false decodes, no action is taken on the first two Skimmer spots received for a given DX station: they are ignored. Only the *third* one is passed through for further processing.
- After the third Skimmer spot for a given DX station is processed, any further Skimmer spots for the same DX station are ignored for 60 seconds.
- This reduces the load on your CPU which is otherwise wasting cycles and energy.

14.2.6 Determining whether DX spots are duplicates

According to veteran DXer Bob K4CY, DX clusters work something like this: *"Any time a radio ham activates a shipping hazard such as BS8 (Scarborough Reef), there is an unwritten rule among DXers that everyone should instantly post a spot to the global DX cluster network proudly asserting that they: can hear the station … or they can hear other hams calling or working the station … or they somehow get the impression that the station is, may soon be, or has been but is no longer, on the air … or they are growing desperate for the station to appear on the spotted band and mode … or on some other band and mode … possibly at dawn next Tuesday; have at least a rough idea of the station's callsign[229]; think the station or pileup is, or at some point should be or possibly, allegedly, once emitted a short burst of RF, on or in the approximate vicinity of the stated frequency[230].*

Furthermore, a sizeable and still growing army of **S**oftware **D**efined **R**adios, fully armed with automated CW and digimode decoders, has its little robotic ears peeled to the short waves 24x7, generating and feeding spots into the **R**everse **B**eacon **N**etwork, PSK Reporter and/or the regular

[228] Processing DX spots imposes quite a load on the CPU, hence when spots are flowing thick-n-fast (*e.g.* during CQ WW CW contest), a backlog can build up, delaying the display of fresh spots and so devaluing them. Duplicate blocking *substantially* reduces the flow, yet still passes plenty of spots to chase. Enabling both <**Block duplicate RBN Spots**> *and* <**Block all duplicate DX spots**> merely wastes CPU cycles, so choose either one: unless you specifically want to *block* duplicate RBN robotic spots while *passing* duplicate human-generated spots, go for <**Block all duplicate DX spots**>.

[229] On CW, any callsign containing lots of dots is liable to be sent and/or received incorrectly, especially if sent too fast to count-the-dots. A significant proportion of DX spots in circulation are, in fact, totally bogus (fakes, jokes, trolls …) while countless others have the wrong callsign, frequency or mode, and sometimes all the above. It's a wonder the DX cluster system hasn't collapsed under its own weight.

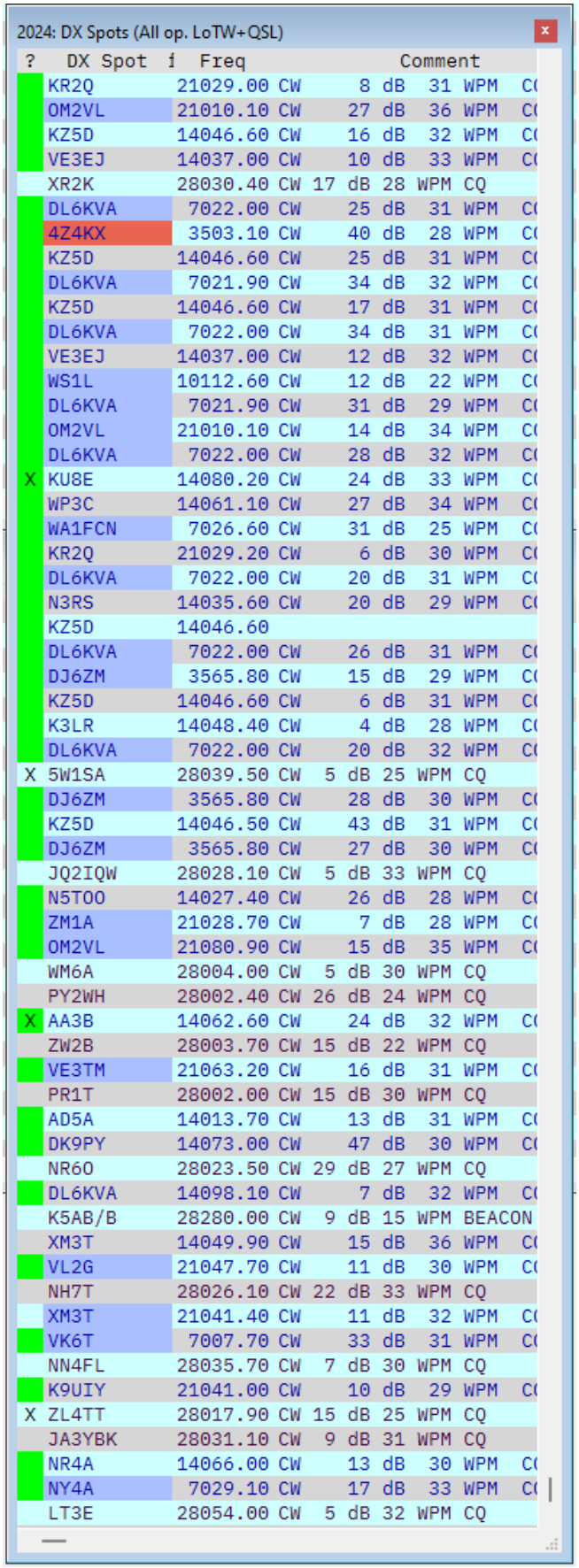
[230] DX stations working 'split' are using *at least* two frequencies (generally a relatively stable TX frequency and a range of RX frequencies depending on the number and desperation of the callers in the pileup), any of which are liable to be spotted. It is not unknown for spotters to post spots on the wrong band, let alone the wrong frequency, for reasons that we can only guess at.

Copyright © 2026 Robert C. Furzer K4CY

DX cluster network. The robots *can* be programmed and configured to apply basic filtering logic, for example weeding out 'clearly busted' spots that don't even resemble legitimate callsigns and straight duplicates (some of which result from poor image rejection in cheap SDRs) … but whether they actually are so programmed and configured in reality is another matter.

In practice, a howling gale of DX spots flows constantly around the globe on the DX cluster and related networks, day in, day out. During major contests such as WPX CW, the gale becomes a veritable *blizzard*, a spot tsunami that threatens to overwhelm the Interwebs.

This screenshot shows what *remained* of the avalanche *after* Logger32 had dutifully applied some sanity and removed the dupes during WPX CW contest ▶

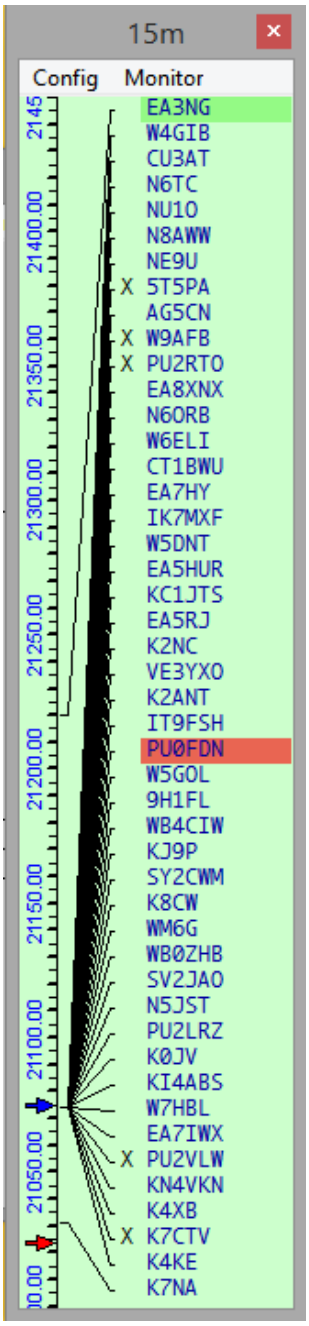


In Logger32, you *can* turn off duplicate DX spot filtering. Successive DX spots arriving in short order for the same shipping hazard will then simply be appended to the bottom of the DX Spots pane in whatever sequence they reached your computer[231]. On the upside, there's little chance of you missing the madness when P5DX is spotted, but if you have better things to do, the annoying spurt of me-too spots might just drive you to drink.

[231] The times shown against the spots were when they were originally *posted* by the spotters. They may have traversed the networks and arrived at your PC sequence-out-of.

Copyright © 2026 Robert C. Furzer K4CY

14.3 BandMaps



BandMaps are an extremely useful graphical alternative to the DX Spots pane's table, arranging DX spots[232] in frequency order ▶



◀ To see the BandMaps, use the **View** menu or click the setsquare icon #15.

The frequency scale down the left side visually depicts the band currently in use on the radio, specifically the band's frequency range defined in the Bands & Modes table.

As with the DX spots table, clicking a spot (callsign) immediately QSY's the active CAT-connected radio to that frequency and mode. It pops the spotted callsign into the Call field of the log entry pane and does online and logbook callsign lookups, if so configured: a wealth of pertinent DX information is presented with just a single click.

Similarly, clicking a given frequency on the scale QSY's the active CAT-connected radio's VFO A to that frequency, sending it to the appropriate mode as well[233].

Spotted callsigns are connected to the frequency scale by lines, showing where on the band they are in relation to other spotted stations. It is not unusual to see most spots clustered around the usual FT# watering-hole frequencies, reflecting the level of interest in the FT modes.

The background and text colors are configurable, naturally. Callsigns are highlighted with colored backgrounds if they are 'new ones' (for various definitions of 'newness'. For consistency, the highlighting colors are the same as those defined in the DX cluster and DX Spots panes, and the worked/confirmed table. To change colors, right-click the DX spot window and select **Setup ⇨ Appearance ⇨ DX Spot highlight colors.**

On this BandMap with my settings, the `PU0FDN` spot stands out in red ▶ because he would be my first QSO with Fernando de Noronha (the first this year anyway[234]). My PC speakers jangled at me with an audio alert as well when the red spot appeared, ensuring I wouldn't miss out on the excitement.

EA3NG, spotted at the top of the example BandMap, has a bright green[235] highlight ▲ indicating that I have not worked an EA station on 15m SSB this year. I have worked EAs on other modes on 15m, otherwise he would have had a blue highlight; and I have worked EAs on SSB on other bands, or the highlight would have been yellow. The colors express a lot of valuable DX information, without me clicking or even mousing-over the BandMap. Your colors and interests may vary.

---

[232] As well as DX spots received from DX Cluster, BandMaps can also display pseudo DX spots (bookmarks).

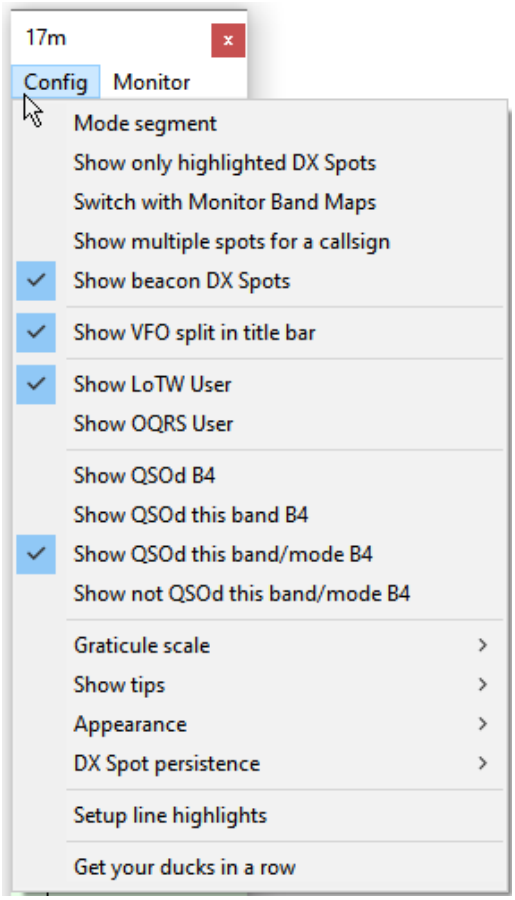
[233] If you are using the AFSK frequency offset option, the VFO will be offset accordingly.

[234] I re-start my DXCC quest every January 1st for the next annual DX Marathon.

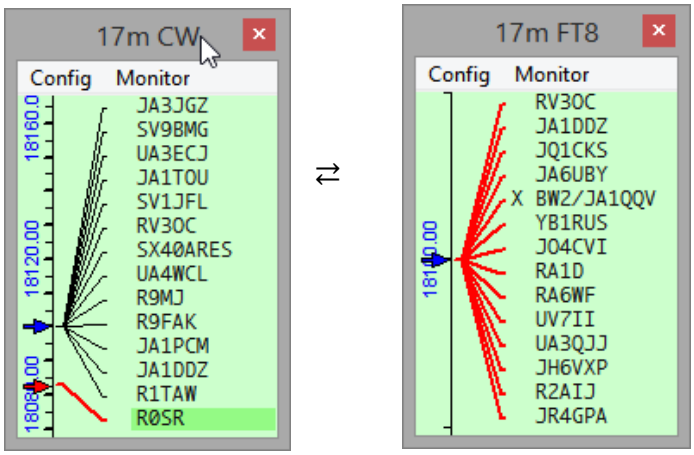
[235] These are *my* highlighting color choices: yours will no doubt differ because I am *special*.

Copyright © 2026 Robert C. Furzer K4CY

14.3.1 BandMap *Config* menu

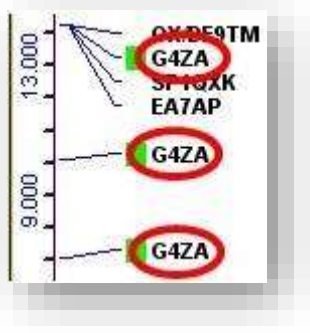


- **Mode segment**: the view zooms-in automatically to focus on the part of the band used for each mode. Tuning the VFO across the band zooms on the region between the upper and lower frequencies for the particular mode segment as defined in the Bands & Modes table). The BandMap's caption shows which mode is used in the segment shown ▼



- When you tune into an adjacent mode segment, the BandMap re-zooms and the scale changes again. If the band segment is very narrow (*e.g.* FT#), the frequency scale may show no numbers.

- **Show only highlighted DX spots**: if you prefer to focus exclusively on working 'new ones' and DXpeditions, this option hides the humdrum boring ones from the BandMap.
- **Switch with Monitor BandMaps**: see below.
- **Show multiple spots for a callsign**: display all spots for a given callsign provided the spots are on different frequencies ▶ This is useful if you have a BandMap with a very wide frequency range (say 2 to 30 MHz) to track the operation of a DXpedition. Otherwise, a given DX callsign only appears once at the most recent spotted frequency.



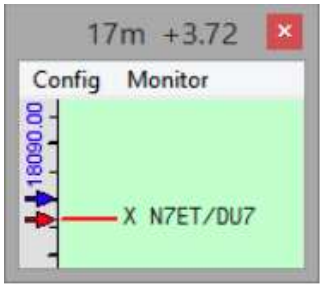
- **Show beacon DX spots**: you may or may not care whether someone is hearing and spotting beacons, such as the NCDXF beacons or the *hundreds* of QRP beacons on 10m. If not, Logger32 can ignore beacon spots. Otherwise they are shown just like other DX stations … but since you can't contact them, they are not alerted and highlighted as new ones.

Hinson tip: sometimes it is useful to know that there is an open path between the spotters and the beacon/s, especially when the band is otherwise devoid of activity (aside from FT# anyway): the beacons may be the only signals they can hear, despite the band being open for business. If you are not a DXer, beacon spots may be an unwelcome and annoying distraction from the stuff you want to see, so go ahead, turn 'em off.

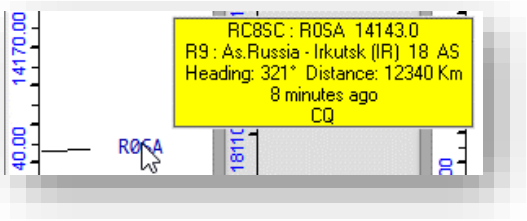
Copyright © 2026 Robert C. Furzer K4CY

- **Show VFO split in title bar** - when the radio is operating split (TX on one frequency, RX on another), this option displays the split differential in the caption in kHz, *provided* your radio reports its VFO B frequency during polling (some don't) ▶

  Notice also that the red arrow (for VFO A) and blue arrow (for VFO B) show the split visually on the BandMap's frequency scale.



- **Show LoTW|OQRS User**: Logger32 identifies spotted stations known to use LoTW or Club Log's **O**nline **Q**SL **R**equest **S**ervice with colored blobs to the left of the callsigns.
- **Show QSOd B4 …**: Logger32 identifies stations we have already logged with an **X** ▲ between the line to the frequency scale and the callsign. There are further options to show the **X** only if we have worked them on this band and/or mode[236].
- **Show not QSOd this band/mode before**: conversely, this option shows the **X** against stations we have *not* yet worked on the spotted band and mode.
- **Graticule Scale**: select the most appropriate scaling for the BandMap frequency display. Different scaling[237] may be used for each BandMap. Logger32 prevents the frequency text from overwriting other frequencies - for example, on the 10m band, you cannot show frequencies every 10 kHz as there is not enough room on the screen.
- **Show Tips**: show spot mouseover tooltips in full or reduced detail, with or without the elapsed time since the spot was sent ▶



  Mouseover the LoTW blobs to show special tooltips, depending on your settings ▼



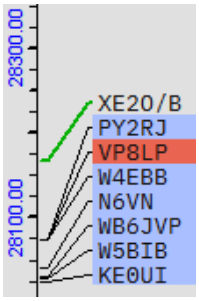
- **Appearance**: make your choice of fonts and colors through a submenu.
  - There are color configuration options for the text, background, lines, graticule, zoom indicators, even the worked-before **X**s.
  - **Show frequency marker B - Split**: shows a VFO B arrow on the frequency scale when the radio is in split mode. This only works if your radio reports the VFO B frequency when polled via CAT. Some early ICOMs didn't, apparently.
  - **Show frequency marker B - Always**: shows the VFO A *and* VFO B frequencies on the BandMap all the time. This only works if your radio reports the VFO B frequency when polled via CAT.

[236] Also, if the top left corner of the worked/confirmed table has been toggled to show the current year *e.g.* **2025**, the worked before **X**s and spot highlighting only take account of stations worked this UTC calendar year. If the corner says **ALL**, it checks the entire currently-open log for a match.

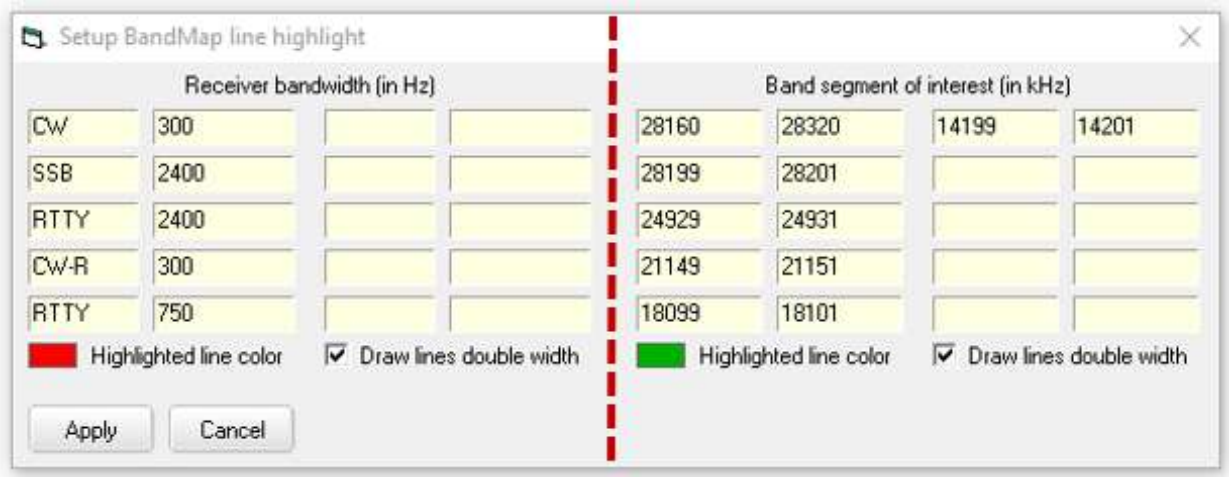
[237] In fact, many BandMap options can be configured differently per-band: for example, you might want a rainbow of BandMap background colors across the spectrum from 160 to 2m. Or something.

Copyright © 2026 Robert C. Furzer K4CY

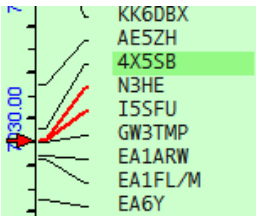
- **DX spot Persistence**: how long should a spot remain visible, after being spotted? The timer starts counting down from the time reported in the spot itself, not when it was placed on your screen. A DX spot reported at 12:01 and using a 5 min persistence, will remain visible until 12:06 and then disappear at 12:07, although the spot data is not wiped. To see a spot that might only just have been removed from the BandMap, simply increase the persistence time and it will magically re-appear, or check the DX spot History. In contrast, if you send SH/DX to the cluster, only those spots that fall within the persistence timing window will be displayed.



- **Setup Line highlights**[238]**:** adjust the width and color of the highlighting ▶ of lines linking the frequency graticule to the callsigns nearby your VFO frequency ▼



The bandwidth defined as 'nearby' typically reflects the mode. For instance, with the CW receiver bandwidth set to 300 Hz, the lines to any callsigns within 150 Hz either side of the VFO A frequency marker are highlighted in the color chosen on the left.



◀ Here, two CW stations have been spotted near my VFO frequency, hence both their lines are bright red. I may hear both, either or neither of them, plus others further HF or LF of the frequency if my receiver filters are actually wider than the defined bandwidth.

On the righthand side of the table (the two columns to the right of the red dashes on the screenshot above), you can specify up to 10 segments of particular interest on the present band (*e.g.* areas for beacons, RTTY, QRP, SOTA …) and highlight the lines that connect the graticule to DX spots in that area in a distinctive color. Simply enter the upper and lower limits of whichever piece of the band you want to highlight, and click then configure the color.

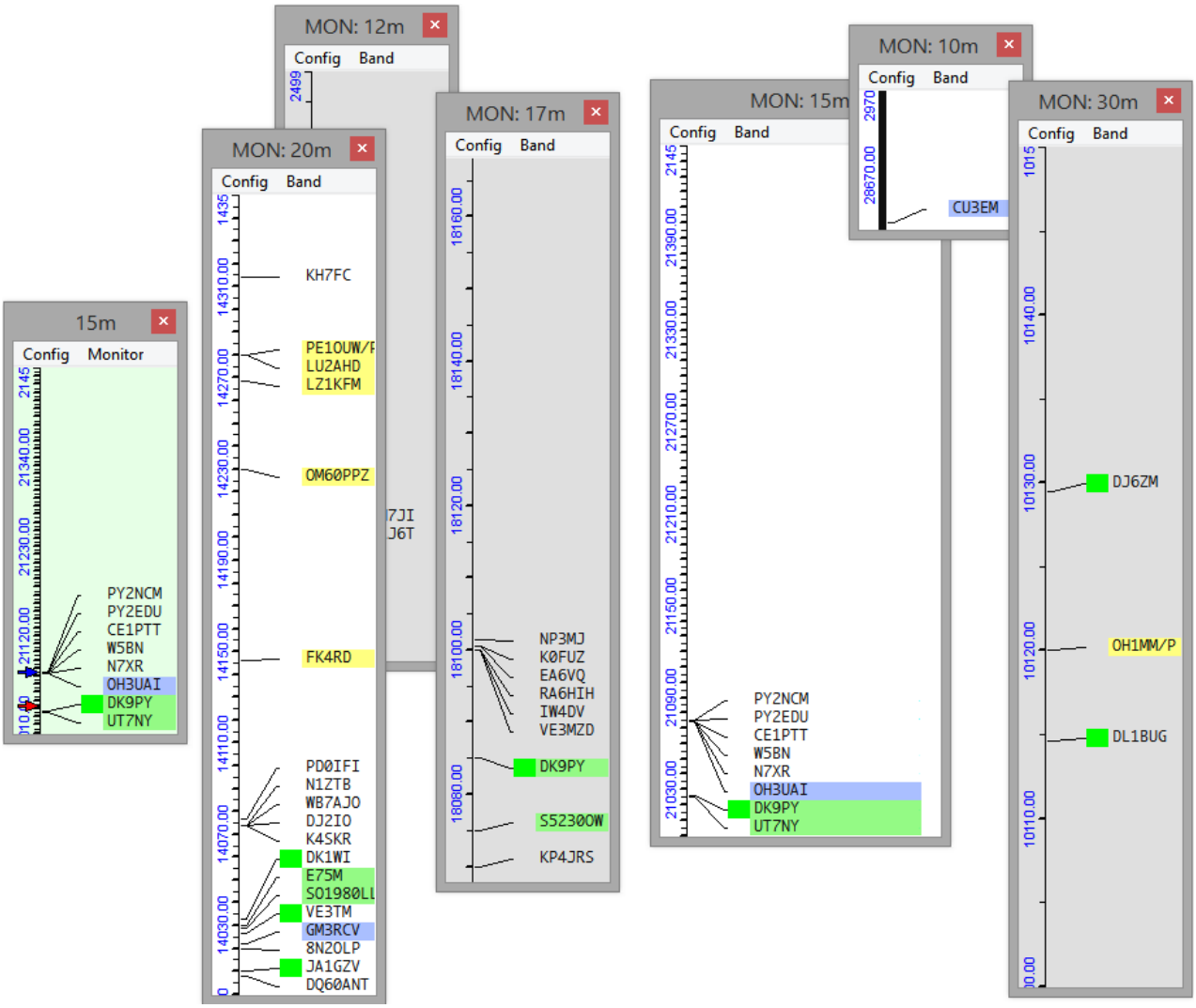
Hinson tip: the 'stations nearby' highlighting takes precedence over the 'band segment' highlighting. With my setup, lines to 10m beacons spotted between 28160 and 28320 kHz are green on the 10m BandMap but go red around the VFO frequency as I tune through them.

Click to double the width of the highlighted lines for an even more dramatic effect, if you like.

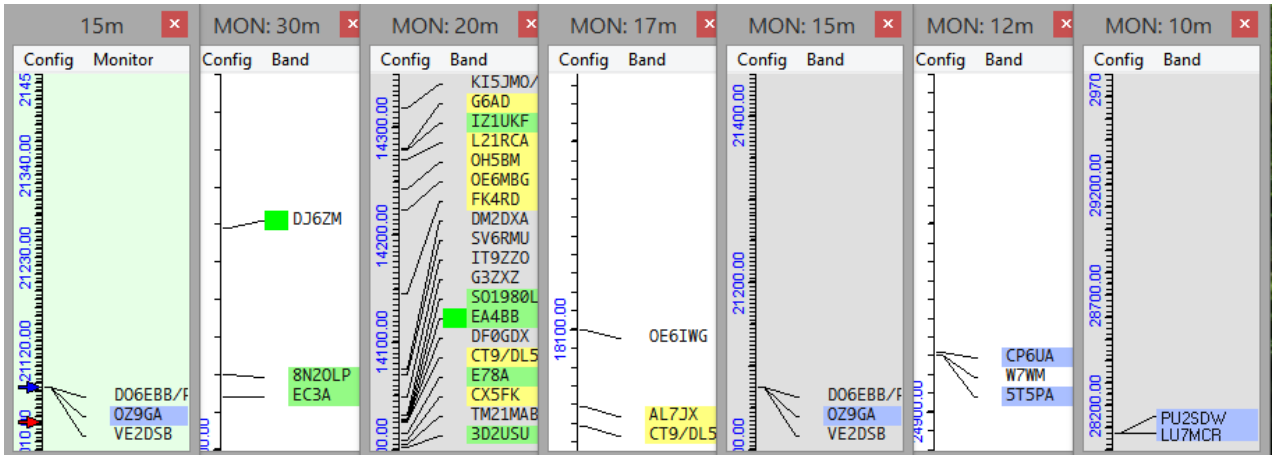
[238] <**Setup Line highlights**> is on the <**Config**> menu for the *main* BandMap, not the *Monitor* BandMaps.

Copyright © 2026 Robert C. Furzer K4CY

- **Get your ducks in a row**: having opened several monitor BandMaps to keep an eye out for interesting DX spot on various bands at once, you may end up with a jumble like this ▼



Rather than mess around with the mouse, carefully resizing and repositioning them, simply click **Config ⇨ Get your ducks in a row** on the main BandMap menu to resize them the same size as the main BandMap, all lined-up in ascending monitor-band-number[239] to the right like this ▼

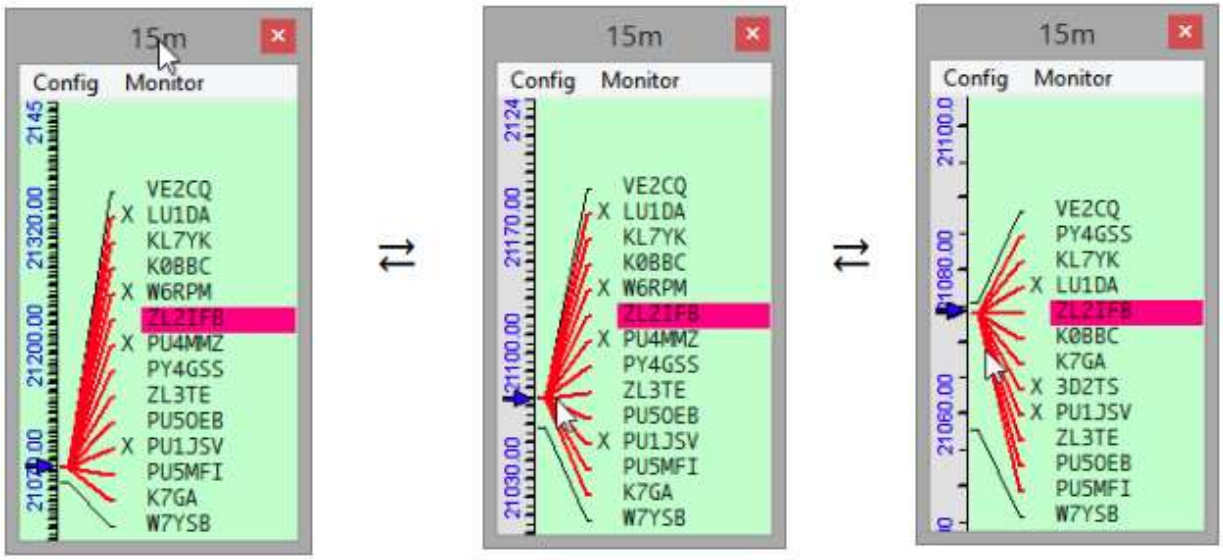


[239] To change the sequence, click <**Band**> on any *Monitor* BandMap, pick another unused number and line up your ducks once more.

Copyright © 2026 Robert C. Furzer K4CY

14.3.2 BandMap zoom feature

Mouseover a particular section of the BandMap that is of interest, in the area of the lines connecting callsigns to the frequency scale, and roll the mouse wheel forward to expand ('zoom in on') the frequency scale in that region in 10 kHz increments per wheel click[240].



Un-zoomed. Notice the frequency scale is green. W7SYB is spotted a little below (LF of) the busy 15m FT8 frequency.

Part-zoomed, while pointing at the red lines. The frequency scale is gray. W7SYB is ~20 kHz LF of the FT8ers.

Fully-zoomed-in. W7YSB is on 21058 kHz and VE2CQ is a little HF of the FT8ers.

Still pointing at the lines, roll the mouse wheel backwards to zoom back out, seeing more of the band until the whole band is shown.

To speed things up, press <**Shift**> while mouse-wheeling to zoom in or out ten times faster.

While zoomed-in, the background of the graticule (frequency scale) changes to whatever color you have chosen in **Config ⇨ Appearance ⇨ Zoom background color** - a visual cue that you are looking at a close-up of *part* of the band. When you zoom all the way out, it reverts to the normal background color for the BandMaps.

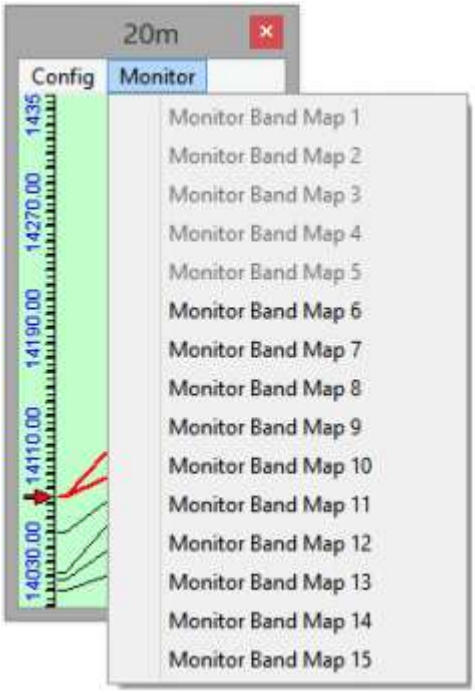
Hinson tip: having chosen a different background color for the scale, you *may* need to zoom in or out, triggering Logger32 to re-draw the BandMap with the new scale color.

[240] Mouse wheel zooming is disabled if the <**Mode segment**> option is active since the BandMaps *automatically* zoom-in as the VFO enters each mode segment. The UDP BandMap cannot be zoomed.

Copyright © 2026 Robert C. Furzer K4CY

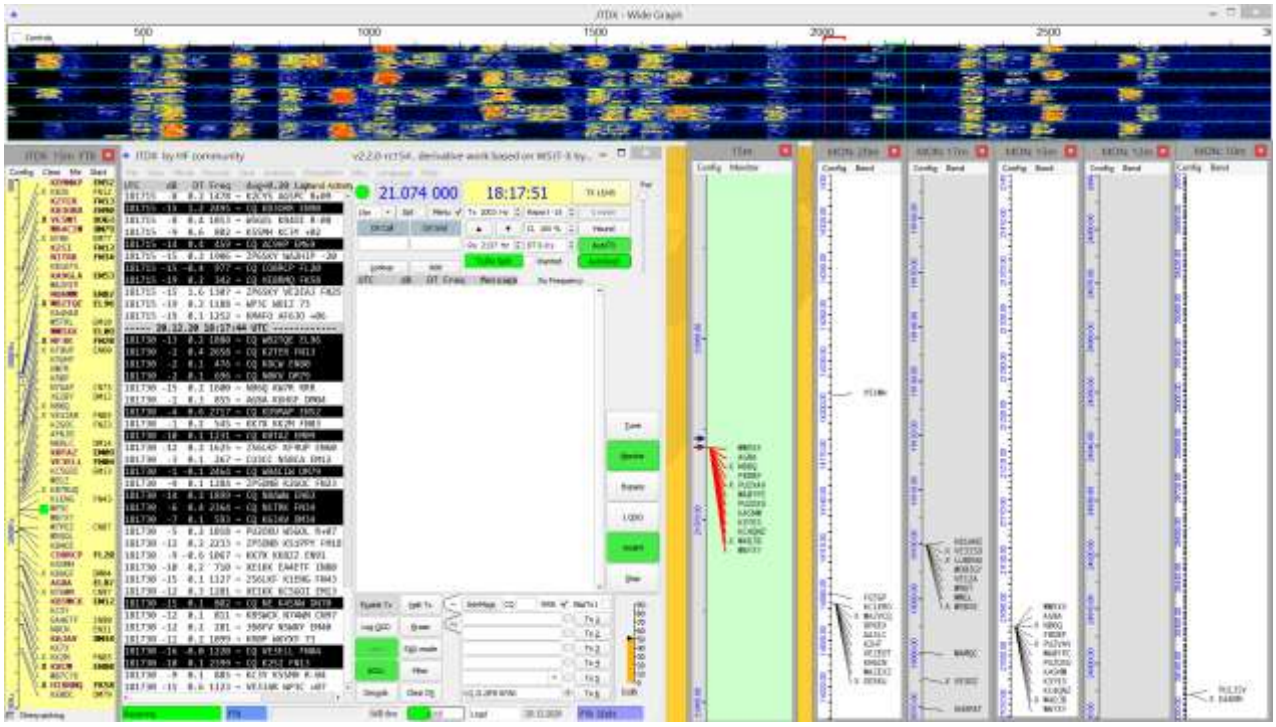
14.3.3 Monitor BandMaps

Aside from the main BandMap for the band to which your radio is currently tuned, up to 15[241] further BandMaps are available to keep an eye on DX spots for other bands.



Click <**Monitor**> on the main BandMap's menu to choose which additional Band Map/s to display ▶

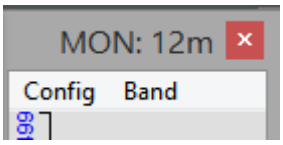
The Monitor BandMaps are simply numbered, so it's not immediately obvious which bands they each cover. I find it helps to set them up in frequency order. In the band selection menu here, the first five BandMaps are grayed out because they are already open: on my system, those are the 10, 12, 15, 17 and 20m BandMaps. I have barely enough room on a second monitor to display those 5 BandMaps side-by-side plus the main BandMap (with a green background) and a UDP BandMap (yellow background) when I'm using FT8, plus the JTDX main screen and the waterfall across the top ▼



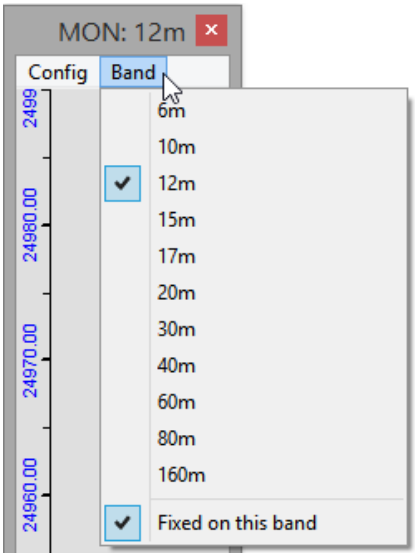
The monitor BandMaps are curiously similar to the main BandMap, with much the same configuration menus and options.

[241] Yes, fifteen! You may need a super wide screen or supplementary display, to fit them all on, along with all the other windows you need to see at once. It is unlikely that all 15 bands will be open at once though.

Copyright © 2026 Robert C. Furzer K4CY



◀ One obvious difference is that monitor BandMaps have a <**Band**> menu to select which band they cover.



The bands offered in the menu are taken from your Bands & Modes table ▶

<**Fixed on this band**> 'locks' that BandMap to the selected band. Simple enough.

Hinson tip: if leave this option un-ticked, with **Config ⇨ Switch with Monitor BandMaps** selected on the main BandMap, the monitor BandMap shows the band you were *previously* on *after* you change bands, rather than the one you are *currently* on (which is always shown on the main BandMap anyway). After changing bands a few times, this results in my neat, orderly array of HF BandMaps becoming disordered, with the 15m one duplicated and the 20m one missing ▼ … it's duck alignment time.



14.3.4 'In focus' color

Curiously, the BandMap window captions *always* have the MS Windows 'in focus' color, *even when they are not actually in focus*. The Band, Mode and Split Alert remain clearly visible in the BandMap captions. In the following example, although the vertical bar cursor and hence focus is actually on the CW Machine window, the 40m BandMap on the left also *appears* to be 'in focus' as well ▼



Both captions have the dark-to-light blue 'in focus' shading (on my system – yours will probably differ).

Copyright © 2026 Robert C. Furzer K4CY

14.3.5 "I'm lost" message on the BandMap caption

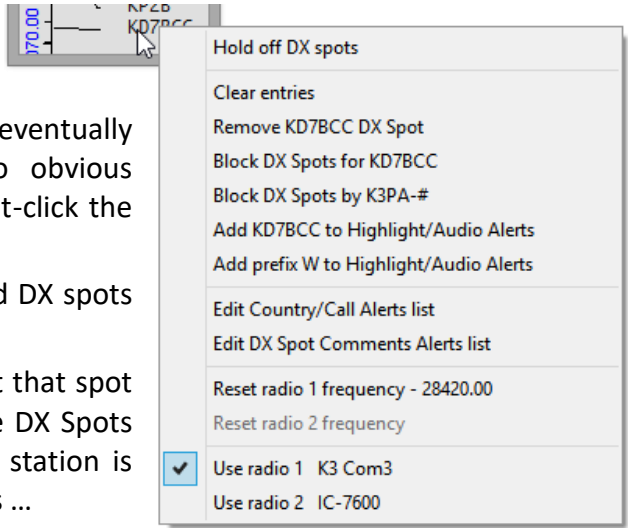
This caption message appears if you ask Logger32 to go to a band/mode that is not configured in your bandplan (such as 60m SSTV) ▶



Confused, it reverts to 20m SSB … and sulks.

Please try not to confuse the poor thing or taunt it.

14.3.6 BandMap DX spot right-click menu



- **Hold off DX spots**: this toggle stops Logger32 adding new spots to the BandMap while enabled. It is not simply a total freeze: the DX spots reach their timeout point and disappear, eventually leaving the BandMap empty. There is no obvious indication that the hold is on, unless you right-click the BandMap to see if this option is ticked.
- **Clear entries**: wipes all the currently-displayed DX spots off this BandMap.
- **Remove [**the right-clicked**] DX spot**: takes just that spot off the BandMap. It is also removed from the DX Spots pane and the DX spot map, but if the same station is spotted again, a new DX spot appears … unless …
- **Block DX spots for [**the right-clicked callsign**]**: in the same way as is possible in the DX Spots pane, it is possible to *block* DX spots for a callsign – specifically the one you right-clicked on the BandMap. The block is total *i.e.* it applies on all bands and modes.
- **Block DX spots by [**whoever posted the right-clicked spot**]**: if you are annoyed by spots from a prolific spotter (such as ZL2IFB!), this option takes care of that annoyance.
- **Add [**the right-clicked callsign**] to Highlight/Audio Alerts**: this is a handy way to establish an audio/visual alert for future appearances of a rare DX station on the BandMaps.
- **Add prefix [**of the right-clicked callsign**] to Highlight/Audio Alerts**: in the same way, you can configure an alert for the rare DX station's 'prefix' (*i.e.* his country).
- **Reset Radio 1|2 frequency – [**freq**]**: after you have clicked a spot to QSY your radio, this option returns it to the previous frequency.
- **Use Radio 1|2**: when you click a DX spot on this BandMap, which radio do you want to QSY? You may for instance be using separate HF and VHF/UHF radios: if you click a spot on the 2m or 70cm BandMaps, you presumably want to send it to the VHF/UHF radio.

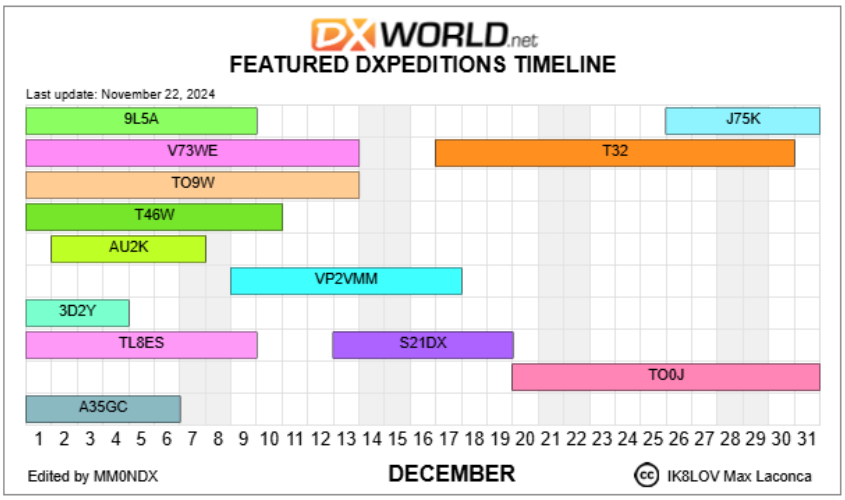
Copyright © 2026 Robert C. Furzer K4CY

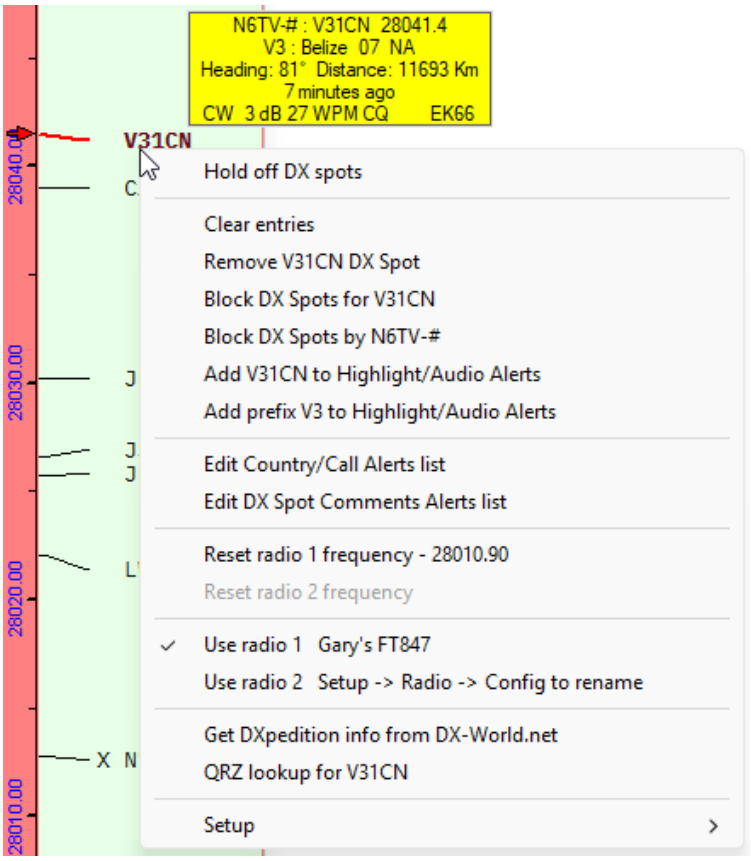
14.3.7 BandMap DX spot browsing

After enabling this setting and clicking to focus on the log entry pane, you can step sequentially up (HF) or down (LF) through spots on the current BandMap by tapping the plus or minus keys on the numeric keypad. With each tap, the radio tunes to the next spot frequency, the DX callsign is put into the log entry pane Call field and looked up, ready for you to work him. Or not.

14.4 DX/DXpedition station information

Max IK8LOV and Col MM0NDX kindly maintain this handy online timeline (a.k.a. GANTT chart, wall planner) showing, for the named month, the callsigns and durations of DXpeditions of which they have been informed ▶





▲ Each of those colorful blocks is clickable, taking us to further information and news about the DXpeditions, published on the DXWorld.net website.

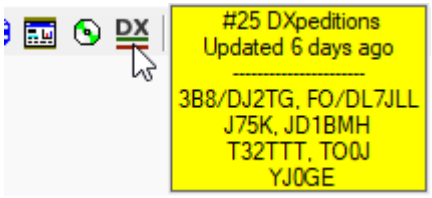
Logger32 can use this information to pick out the callsigns of known DXpeditions in **bold** (and in your choice of text color) in the DX Spots pane and BandMaps whenever they are spotted.

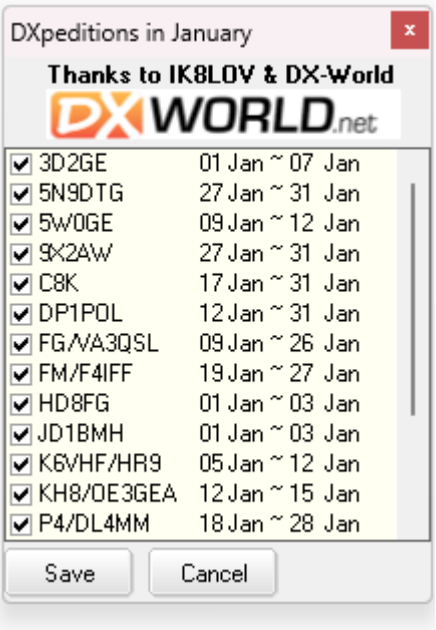
◀ Right-click a **bold** DXpedition spot, then click <**Get DXpedition info from DX-World.net**> to visit their DX-World information page, or <**QRZ lookup for** [station]> to visit their QRZ.com page, in your browser.

Hinson tip: the QRZ lookup works for plain vanilla non-DXpedition stations too. Nice!

Copyright © 2026 Robert C. Furzer K4CY

Hovering over DX icon #25 pops up a message telling us when the list of DXpeditions was last updated, and lists *current* DXpedition calls ▶



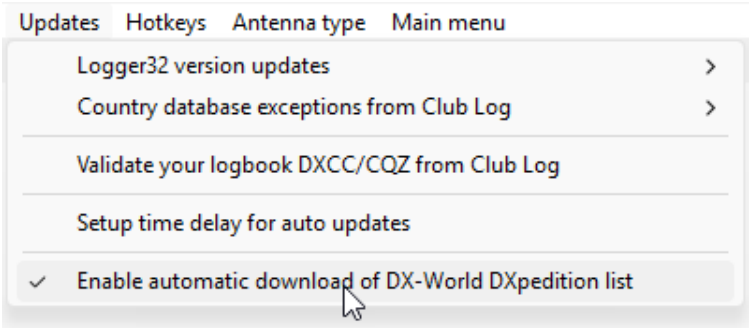


◀ Clicking icon #25 triggers Logger32 to connect to the DX-World website, download and install the latest available information on DXpeditions. We may un-tick boring DXpeditions that we don't particularly care about, then click <**Save**> to have Logger32 store the info as *C:\Logger32\DXpeditions NN.txt* where NN is the month's number.

Hinson tip: spotted callsigns are filtered, highlighted and alerted as normal so we shouldn't miss-out on any new ones, regardless of whether they are identified as DXpeditions or not. This function simply enables the right-click option to <**Get DXpedition info**> from the DX-World website for known DXpeditions *i.e.* DX spots picked out in **bold**.

Rather than remembering to update manually, Logger32 can update the DXpedition info automatically.

Simply click the option at the bottom of **Setup** ⇨ **Updates** ▶



If <**Enable automatic download of DX-World DXpedition list**> is ticked, whenever Logger32 starts up, it checks its default directory for the presence of a data file listing the current month's DXpeditions *e.g.* during December (month 12), it looks for *C:\Logger32\DXpeditions 12.txt*.

- If the current month's DXpedition data file is *not* found, Logger32 downloads the information from the DX-World website to *C:\Logger32\DXpeditions nn.txt*.
- If the current month's data file *is* found (having been downloaded previously), Logger32 checks the file date. If the file is more than a week old, it downloads the information from DX-World and recreates the current month's data file, thus keeping its list of DXpeditions updated weekly during the month. Previous months' files are automatically deleted.
- If a surprise DXpedition has just been announced and added to the timeline for the *current* month, we can either wait patiently up to a week for Logger32 to update itself automatically, or click icon #25 to trigger an immediate update, now, right away.
- If DX-World is prematurely offering data not for the *current* month but for the *following* month (which happens), Logger32 downloads and saves the data file with a number for the *following* month *e.g.* during December, it creates *C:\Logger32\DXpeditions 1.txt* for January. In this case, the *current* month's data file will no longer be updated by Logger32 until next year. Too bad.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: the DXpedition data file is plain text with lines of the form:

3D2GE^**45658**^45664^https://www.dx-world.net/pacific-travels-by-oe3gea/

The two numeric parameters represent the first and last days … so potentially we can hack (re-purpose) the <**Get DXpedition info**> function to embolden spots and jump to an arbitrary URL for *any* callsign, by editing the data file or replacing it entirely. In case you're wondering, **45658** is the Excel day number for January 1st 2025. Day 1 would have been January 1st 1900.

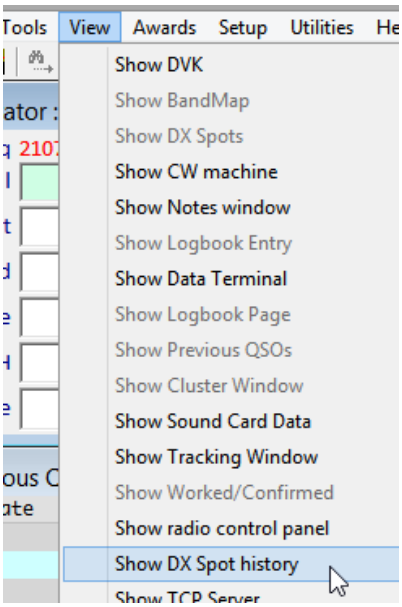
Hinson tip: a start date of the 1st of the month is shown even if the DXpedition was already active at the end of the prior month. An end date of the last day of the month is shown even if the DXpedition will continue into the following month. Visit the DXpedition info page for the true dates – or rather the published dates, which may not be entirely accurate anyway.

Bob tip: to disable the feature, un-tick <**Enable automatic download …**>, delete all the *C:\Logger32\DXpedition xx.txt* files … and don't click icon #25.

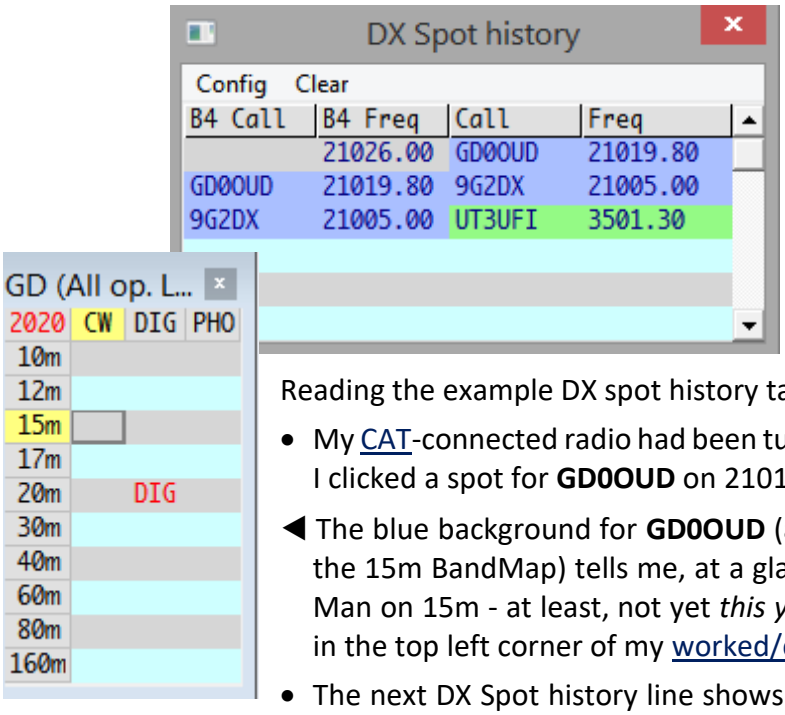
DXpedition spots on the Tracking window's DX Spots, IOTA Spots and JTDX maps are subtly emphasized relative to ordinary DX stations with thicker borders to their markers.

14.5 DX spot history

Logger32 maintains a cache of up to 20 *DX spots that you have clicked recently*. They can be displayed using **View ⇨ Show DX spot history ▶**



The example below shows that I had only clicked three DX spots since starting Logger32 ▼



Reading the example DX spot history table from the top down:

- My CAT-connected radio had been tuned to 21026 kHz (the 'B4 Freq') when I clicked a spot for **GD0OUD** on 21019 kHz.

◀ The blue background for **GD0OUD** (also shown on the DX Spots pane and the 15m BandMap) tells me, at a glance, that I had not worked the Isle of Man on 15m - at least, not yet *this year* since the current year was shown in the top left corner of my worked/confirmed table.

- The next DX Spot history line shows I then clicked a spot for another new one this year on 15m - **9G2DX**, also on 15m CW. My VFO instantly retuned from 21019 to 21005 kHz. The 'B4 Call' shows that GD0OUD had been in the Call field of the log entry pane when I clicked the 9G2 DX spot.

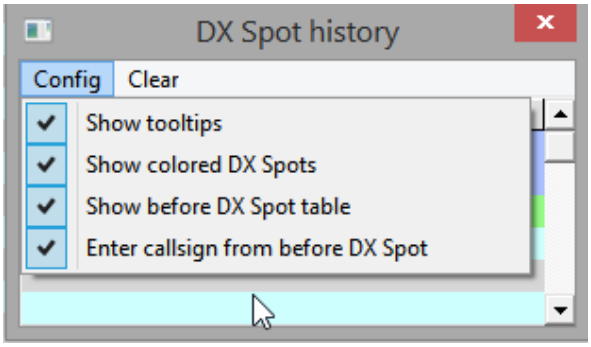
Copyright © 2026 Robert C. Furzer K4CY

- Later on, I reluctantly gave up calling 9G2DX and moved to 80m CW to chase **UT3UFI** who would have been a new one for me on 80m CW this year, as indicated (in this example) by the green highlighting in the table.

Entries in the DX spot history table are clickable, just like DX spots elsewhere in Logger32 … so thanks to the DX spot history table, I could easily return to 15m for another go at working GD0OUD or 9G2DX later, even if those spots had long since scrolled away from the DX cluster and DX Spots panes, and had timed-out and disappeared from my 15m BandMap.

Hinson tip: even big guns sometimes fail to work spotted stations, and move on. For little pistols like me, it's a common occurrence. Either way, the DX spot history table makes it easier for us to go back later for another try, hoping that the path is stronger and the pileup smaller.

14.5.1 DX spot history configuration



The DX spot history window's <**Config**> menu is, to be honest, enigmatic ▶

- Pop-up **tooltips** containing additional information about the spots can be shown when you mouseover any of the callsigns shown in the history.
- The spot text and background **colors** can be configured to highlight 'new ones', using the same color scheme as the spots on your DX Spots pane and BandMaps.
- <**Show before DX spot table**> actually displays two *columns* showing the 'B4 Call' callsign and 'B4 Freq' VFO frequency you were monitoring or working when you clicked the DX spot.
- I can barely guess what <**Enter callsign from before DX spot**> does - something to do with remembering and recalling the current callsign when you click a DX spot maybe?[242]

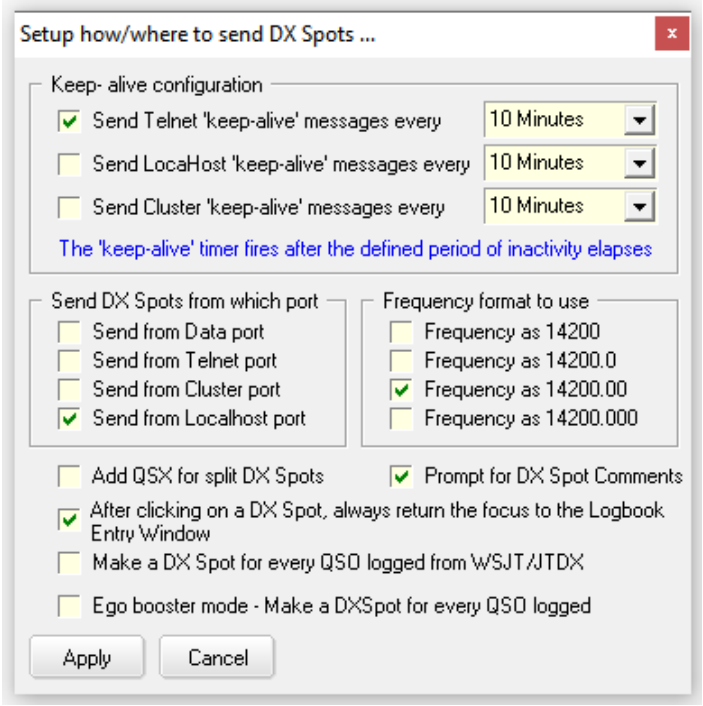
[242] If you figure it out, please let me know or tell us on the Logger32 reflector.

Copyright © 2026 Robert C. Furzer K4CY

14.6 Posting (sending) DX spots to DX cluster

14.6.1 Configure spot sending

Logger32 makes it easy to post spots to the DX cluster network, if correctly configured through **Setup ⇨ DX Spots ▶**



- **Keep-alive configuration**: Logger32 can send null messages to the DX clusters periodically in order to prevent them disconnecting us due to our perceived inactivity. The keep-alive function can be enabled and timed separately for the connections opened on the Telnet, LocalHost and Cluster tabs of the DX Cluster pane.
- **Send DX Spots from which port**: when you compose a DX spot, which cluster connection should it be sent out on? Pick one.
- **Frequency format to use**: most DX cluster users prefer spots with frequency resolutions of 100 Hz. Some of us prefer greater resolution down to 10 or even 1 Hz, although most DX clusters truncate kHz frequencies after the first decimal digit.
- **Add QSX for split DX Spots**: when working or logging DX stations working split, Logger32 can optionally add the DX station's listening frequency (our TX frequency) to the spot comment as well as spotting the DX station's TX frequency as normal.

> **Hinson tip**: although, at first blush, this may seem useful, consider carefully whether spotting specific QSX frequencies is really such a good idea. A key reason why DX stations work split is to get their callers to spread out a little, reducing mutual QRM to improve copy and boost QSO rates. If you post a specific QSX frequency, you are (in effect) encouraging lazy, naïve or inept DXers to click the spot and transmit on that specific QSX frequency. Commenting vaguely "UP ~5" or "UP 5-10" *hints* that callers should listen with the pileup to select a suitable TX frequency, preferably one that is not piled high with other callers! On the other hand, perhaps you are selfish enough to *want* to lead DXers astray, boosting your own chances of making a QSO by sending those lazy, naïve or inept callers to a spurious split frequency. 😒

- **Prompt for DX Spot Comments**: select this if you want the *option* to add short, helpful, pertinent comments to your DX spots before sending them – for instance "LP" (long path) or "UP" (he's working split, listening higher in frequency). You don't *have* to add a comment to every spot you send: simply hit return when prompted and the spot is sent without your comment. If you select <**Add QSX for split DX Spots**>, Logger32 pre-fills the comment field with the QSX frequency (see tip above) but <**Prompt for DX Spot Comments**> gives you the chance to modify or delete it before sending, perhaps deliberately making it less precise or changing it to a QSX frequency *range* rather than a specific QSX frequency. You can of course do this even *without* selecting <**Add QSX for split DX Spots**>: simply compose the comment as you wish before you send it.

Copyright © 2026 Robert C. Furzer K4CY

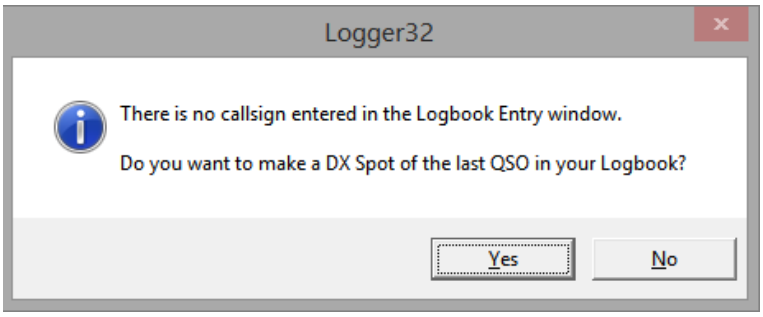
- **After clicking on a DX Spot, always return the focus to the Logbook Entry Window:** select this to have Logger32 focus (put the cursor) on the log entry pane whenever you click a DX spot, ready to log a QSO. It automatically pre-fills the log entry pane's Call field with the spotted DX station's callsign, and looks up any previous QSOs, updates the worked/confirmed table *etc*.

> **Hinson tip**: we can always edit the pre-filled callsign (*e.g.* if the DX station spotted as "P5DX" turns out in fact to be, say, P5DX/MM or SP5DX) or click somewhere else if we aren't ready to log the QSO just yet.

- **Make a DX Spot for every QSO logged from WSJT/JTDX**: I'm not sure why anyone would want to do this, but the option is offered for those who feel the need to brag about *all* their digimode QSOs. That might conceivably be appropriate if they are not very active and *only* ever work "DX" stations, or to promote awareness that a normally dead band is experiencing a rare opening, but such promiscuous spotting is generally frowned upon on the regular DX cluster network. Leave it to the Skimmers on the RBN. The robots don't care about being called 'promiscuous'.
- **Ego booster mode – make a DXSpot for every QSO logged**: see my previous comment! This option is potentially even worse, spotting everyone you log regardless of mode. Use with extreme caution.

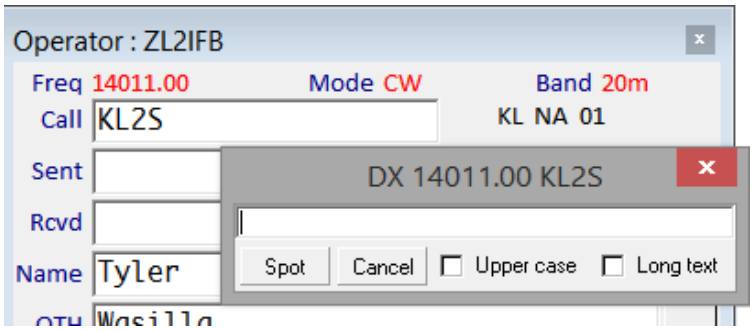
14.6.2 Sending (posting) DX spots

While you are logging a QSO, <**Ctrl**+**D**> prepares to sen**D** a spot to the global DX cluster network using the frequency and callsign currently in the log entry pane (if any).



◀ If the log entry pane is blank when you press <**Ctrl**+**D**>, Logger32 assumes that you *probably* meant to spot your most recently logged QSO instead.

If **Setup ⇨ DX Spots ⇨ Prompt for Comments** is set, you are prompted to add a comment to the spot, if you wish ▶



If you're not a competent typist and can't use the <**Shift**> key properly, tick <**Upper case**> … to *shout* the entire spot in CAPITALS 🕪

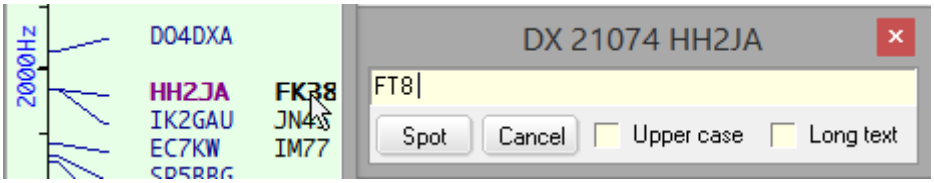
Tick <**Long text**> to be able to type and send longer comments to DX clusters that accept them. However, most cluster users will only ever see the truncated versions, so it's probably not worth the effort.

When you're ready, click <**Spot**> to send the spot to the cluster through the DX cluster tab defined earlier. Of course you must already be connected to DX cluster for this to work.

Copyright © 2026 Robert C. Furzer K4CY

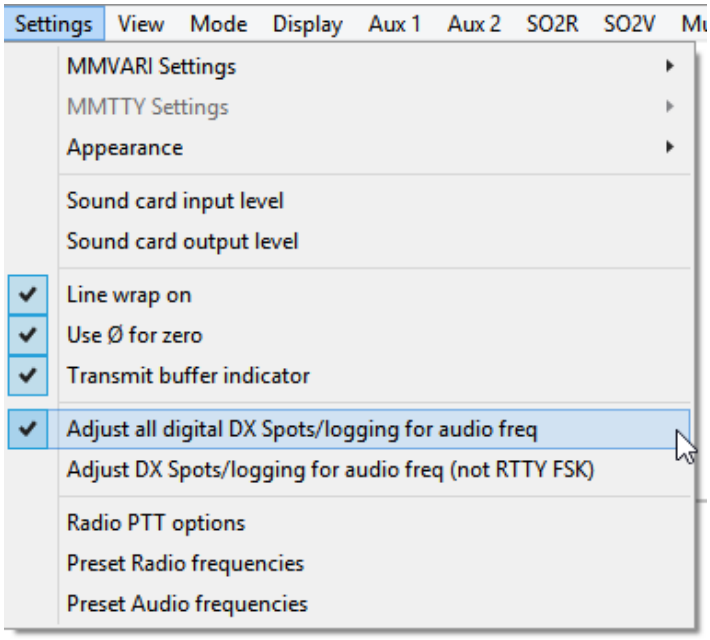
There are two further ways to spot stations:

- Type the DX command into the panel at the bottom of the DX cluster window, correctly formatted *e.g.* DX 7025 K4CY Hi there Bob!
- On the UDP BandMap, right-click the gridsquare (if shown – otherwise click in that area) for an interesting DX station that has been decoded in JTDX|WSJT-X. Logger32 can pre-fill the comment field with the digimode in use *e.g.* ▼



14.7 Adjust PSK/RTTY DX spots for audio frequency

When you click a DX spot for a PSK or RTTY station, Logger32 can automatically adjust the VFO frequency command sent to the radio to compensate for the audio frequency (offset) in MMVARI or MMTTY, such that the transmission ends up on the correct RF [mark] frequency. The option is on the Sound card data window <**Settings**> menu ►



Also, with the option selected, PSK and RTTY spots you post to DX cluster have the calculated RF frequencies in the same way.

The frequencies of FT# spots in JTDX|WSJT-X are *not* adjusted for the audio offset, however, largely because on those modes split operating is the norm. There is no real advantage (often disadvantages, in fact) in transmitting on precisely the same frequency as a spotted station, or indeed any other station you are working. Logger32 simply spots the VFO frequency, generally one of the standard FT 'watering hole' frequencies such as 14074 kHz.

14.7.1 DX spots received from DX cluster

When you click a DX spot, Logger32 looks up the spotted frequency in the Bands & Modes table to ascertain the likely mode. With an RTTY or PSK spot, Logger32 checks whether the Sound card data window is open. If so, before sending the QSY command to your CAT-connected radio, it adjusts the VFO frequency according to your preferred audio tone in MMVARI/MMTTY, such that the spotted RTTY/PSK signal should appear at your preferred audio frequency (if you can hear the DX!).

Copyright © 2026 Robert C. Furzer K4CY

The audio frequency correction occurs if:

- The option is selected.
- The Sound card data window is in use.
- The likely mode is RTTY or PSK (determined from the Bands & Modes table).
- The radio mode is SSB, USB, LSB, PKT-USB, PKT-LSB, DATA-LSB or DATA-USB or DIG (also determined from the Bands & Modes table) … but *not* FSK (see below).

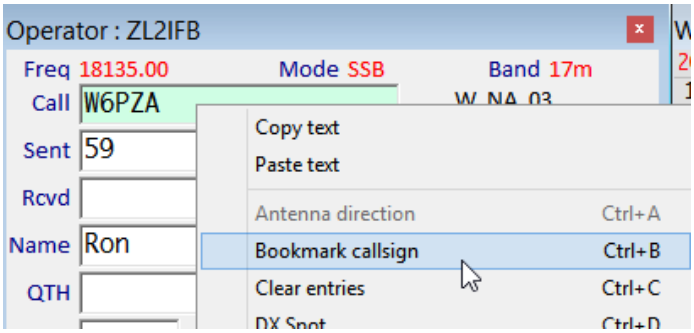
If you complete a contact, the audio offset is reversed such that the true RF frequency is logged.

14.7.2 Frequency-Shift Keying

If you are using FSK, no VFO adjustment for audio frequency is necessary: in effect, the radio does that automatically. There is one complication with RTTY spots: the MMTTY set up should match your FSK tones (*i.e*. standard tones or continental). It is also advisable to have your radio set the same way. If everything matches up, all will be well.

14.8 *Pseudo* DX spots (bookmarks)

You can bookmark interesting callsigns privately on your BandMaps, DX Spots pane and DX spot map *without* actually posting them to the public DX cluster network.



◀ Simply *precede* the callsign with an exclamation mark in the log entry pane then <**Enter**> (as if you were logging it); or press <**Ctrl**+**B**> to **B**ookmark the callsign; or use the right-click <**Bookmark callsign**>.

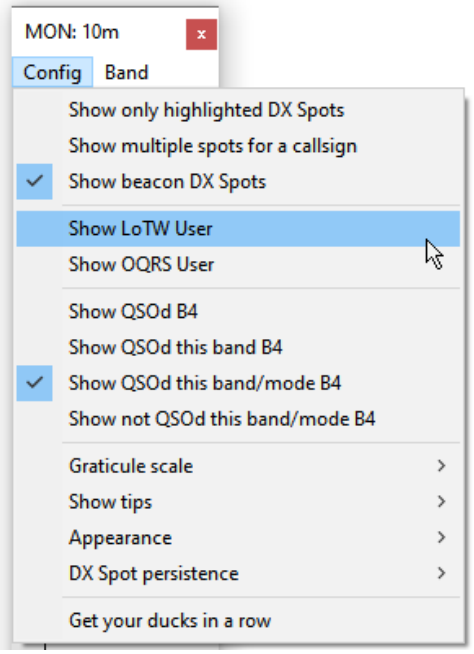
You can even use the $**Bookmark**$ macro in the CW Machine to pseudo-spot the callsign currently in the Call field of the log entry pane.

By default, bookmarked stations (pseudo-spots) are highlighted just like genuine DX spots, and can be used in exactly the same way (*e.g.* click to chase). However, if you prefer a distinctive forecolor (text/font color) for your bookmarks, right-click the DX Spots pane then click **Setup ⇨ Appearance ⇨ Pseudo spot forecolor**.

Hinson tip: if bookmarks look useful to you, check out the scratchpad too.

Copyright © 2026 Robert C. Furzer K4CY

14.9 DX cluster and spots FAQs



Q. How come the colored blobs denoting LoTW users aren't showing up on just one band's BandMap?

A. Carefully compare the <**Config**> settings on that band's BandMap against the other bands ▶

The LoTW and Club Log **O**nline **Q**SL **R**equest **S**ervice user blobs can be enabled or disabled independently on each BandMap.

Don't ask me *why* you would want to show the blobs on some bands but not on others. I have no idea.

While you are at it, it's worth checking the other settings for consistency across bands too.

Q. Can I edit the drop-down list of available Telnet-based DX clusters?

A. Yes. One-by-one, you can manually edit the details for each individual cluster, or delete them, or add more, as described in the previous chapter. Alternatively, if you have a bunch of changes planned and insufficient patience, you can edit the data files *i.e. C:\Logger32\TelnetAddresses.INI* (for the clusters listed in the Telnet tab) *or C:\Logger32\LocalhostsAddresses.ini* (for the Localhost tab) in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad. The syntax for each line is:

`Name here~URL|port`
e.g. `AE5E~dxspots.com|7300`

Having edited and saved the file, stop and restart Logger32 to read it in, updating the list of available clusters in the DX cluster pane's right-click menus.

Hinson tip: you really only need *one* reliable Telnet cluster with good connections to the cluster network feeding you plenty of fresh spots, perhaps with another as a backup. Trust me, you'd be wasting good operating time by studiously compiling and then maintaining long lists of *all* the DX clusters you can find. Logger32 'only' lists 20 anyway – that should be more than sufficient for anyone.

Q. My cluster connection has been working fine for ages but not today. Help!

A. That's almost certainly not your fault and nothing to do with Logger32. By *far* the most likely reason is simply that your normal DX cluster server happens to be offline (down, unavailable, broken, on fire, flooded out, QRT) at the moment. Cluster servers are pretty reliable on the whole providing sub-second response times to hundreds of users around the clock, but they are not really designed to offer availability levels that would be appropriate for, say, a bank's website or a missile control system. They also differ in frequency accuracy.

You can wait patiently for your usual DX cluster to come back online, periodically trying to reconnect through the DX cluster window. Meanwhile, enjoy yourself finding the DX without

Copyright © 2026 Robert C. Furzer K4CY

the benefit of DX spots, rediscovering old skills! If that's too hard, visit one of the Web-based DXcluster services in your browser: it won't send spots to Logger32 but still you'll have plenty of DX spots to chase manually.

If there's no sign of your usual cluster springing back to life within a few hours, or if you just can't wait a moment longer, try connecting to a different DX cluster in the hope of finding one that is more reliable. Here are a few popular DX clusters worth trying:

Cluster	Address	Port	Notes
AE5E	dxspots.com	7300	Ron's DXcluster node is rarely down
EA4URE	EA4URE.com	7300	DXcluster de la **U**nión de **R**adioaficionados **E**spañoles
LZ7A	Cluster.LZ7AA.com	7300	Sofia Radio Club
PA4JJ	77.174.195.163	7300	Jan PA4JJ's DXcluster node runs on a Pi
RBN	telnet.reversebeacon.net	7001	**R**everse **B**eacon **N**etwork – automated Skimmers and digimoders spew forth a veritable *torrent* of DX spots that will stress your system … and test your patience
W4MYA	dxc.W4MYA.us	7373	Central Virginia Contest Club DXcluster
These DXclusters all worked fine when I last checked them in November 2024			

Loads more clusters with Telnet capabilities are listed at DXcluster.info/telnet/ ▼



Despite being listed in some table somewhere, at any given time, some DXcluster nodes are likely to be unresponsive, moribund or dead, perhaps terminally. So, if one doesn't seem to be working when you try to connect, simply go for another.

Hinson tip: the DXcluster.info list lets us filter cluster nodes according to the cluster software they are running. Cluster commands and syntax can vary, so if you are familiar with, say, a CC Cluster or DX Spider node, you might like to un-tick the other types then click <**Filter**> to shrink your options to a more easily-digested set.

Also, the NOTES/SOFTWARE column shows which versions of the cluster software each server is using: clusters that are numerous versions behind others of the same type are probably not being actively maintained. Perhaps the sysops have lost the plot.

Finally, you *might* find that 'local' clusters are more responsive and reflect local radio conditions better than DX ones, although personally I don't notice any difference, so I prefer reliable clusters, regardless of proximity.

Through the global DXcluster network, many DXclusters share the same spot information so simply pick a cluster from the list, enter its address and port, and off you go. You may have to reconfigure any server-side filters and settings though … which is easier if you are using CC User.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: cluster-based spot filtering is more efficient than Logger32's filtering, using the computing resources of the cluster node and reducing the amount of unnecessary traffic sent to your shack PC. The cluster can be configured to send only spots for bands and modes you normally operate, ignoring and not wasting time and bandwidth on others. However, it is relatively crude whereas Logger32's filtering is more sophisticated, for example taking advantage of the information from your log to distinguish ordinary, boring spots for DX entities you have already worked, logged and maybe confirmed on the spotted band and mode, from exciting 'new ones'. The two sets of filters work well together. They get along and play nicely.

"There's no point in inviting the world into your home if you only allow a few of them in."

Bruno ISOGQX

Clusters differ: some receive spots from local CW or RTTY Skimmers, from the local area or country, with or without RBN connections, with or without FT# spots (possibly from PSKreporter), using various cluster server programs *etc*. Some are 'private', meaning they are only available to logged-in club members, circulating spots from club members either to other club members or publicly depending on the configuration and commands available.

By the way, DX cluster Telnet addresses may be written like `node.name.here:port` or `telnet://node.name.here:port` *e.g.* `telnet://dl8las.dyndns.org:7300`

Some DX cluster nodes are accessed by their numeric IP addresses but most have names, letting your system look up the corresponding IP addresses dynamically using DNS.

DX cluster system administrators can choose any valid Telnet port number. Different ports on the same node may offer different feeds (*e.g.* with or without RBN and FT# spots).

"Node", "server" and "system" are synonymous in this context.

Although web-based DXclusters such as DXsummit.fi using https can be viewed with a web browser, Logger32's DX cluster functions use Telnet – a simpler and more efficient Internet protocol, better suited to rapid dissemination of DX spots, especially when coupled with Logger32's ability to QSY your radio instantly to the spotted frequency with the spotted DX callsign all ready to log. Seconds count in the race to bag newly-spotted juicy morsels of DX *before* the raging ravenous horde descends.

Q. Why does the radio frequency disappear from the log entry pane after I click a DX spot and the radio QSYs?

A. Hmmmm. Odd. It seems the CAT connection between your PC and radio is being dropped *after* the QSY command is sent. Use the radio debug window to take a closer look at the CAT data flowing at that point and hopefully you'll figure out what's going on. Good luck.

Q. Where *is* that guy with a weird callsign in the DX Spots pane and BandMap?

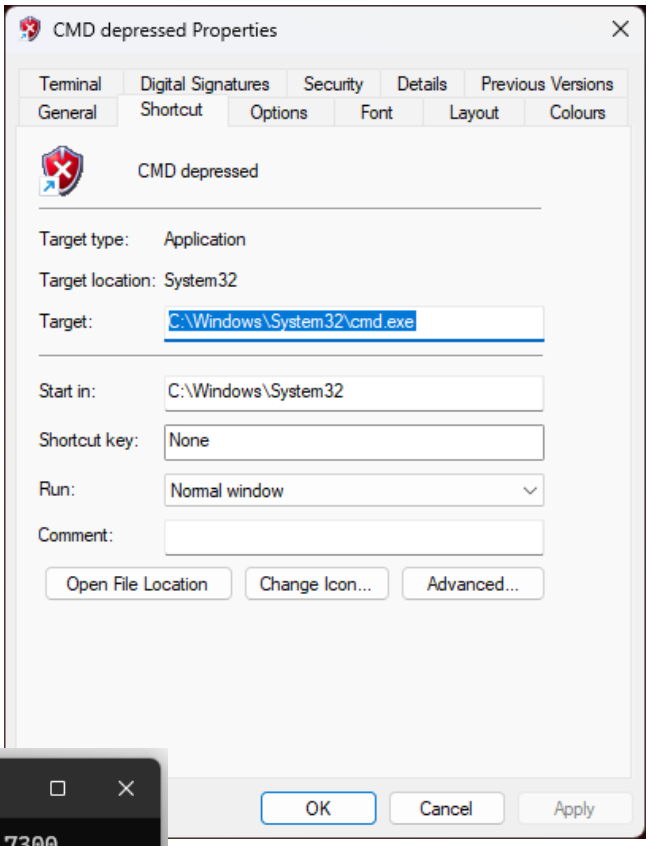
A. Mouseover the callsign to see a tooltip with the country name (provided Logger32 can identify it – and it seldom fails). It will also tell you if that country is *needed*. That is, provided you have configured your BandMap tooltips to show this useful information.

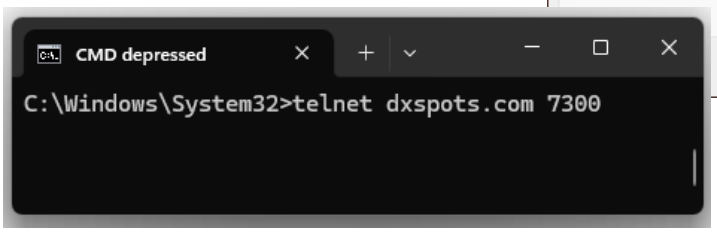
Copyright © 2026 Robert C. Furzer K4CY

Q. How do *you* check whether various Telnet clusters are usable, Gary?

A. I find it quicker and easier to do this from the command line in Windows:

1. First, Telnet needs to be enabled in Windows features under Control Panel. This is a one-time PC setting. If you need instructions, ask Google.
2. Now I open an ordinary command prompt. Having previously defined a desktop shortcut for this ▶ I simply double-click to open it.
3. At the command prompt, I type **telnet** followed by the DXcluster address and (optionally) the port number separated by spaces, then <**Return**> ▼

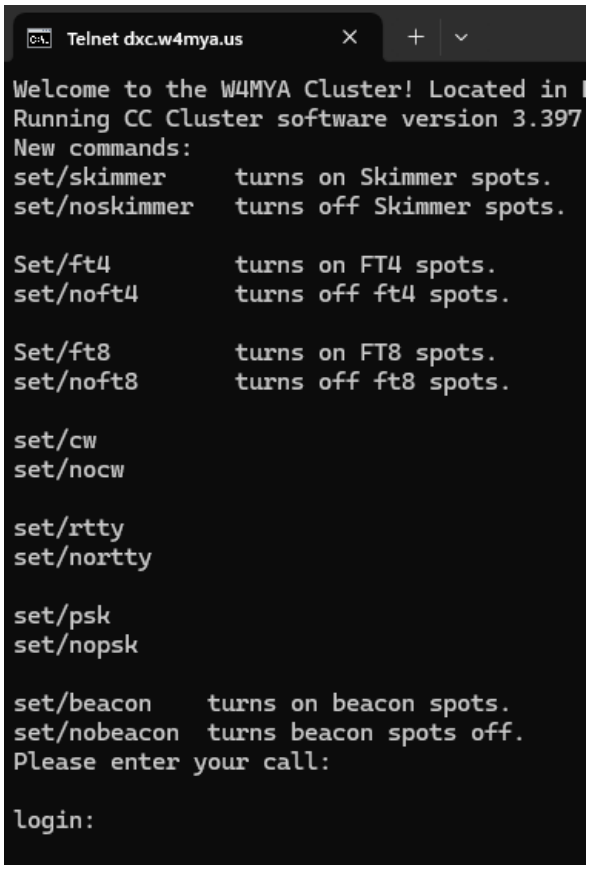




Hinson tip: some DX cluster nodes direct incoming Telnet connection requests to their default port numbers, even if none was specified. Some don't. Some nodes stream selected spots (*e.g.* RBN or FT# spots) only to specific ports. So, it is generally best to specify a particular port number if you know it.

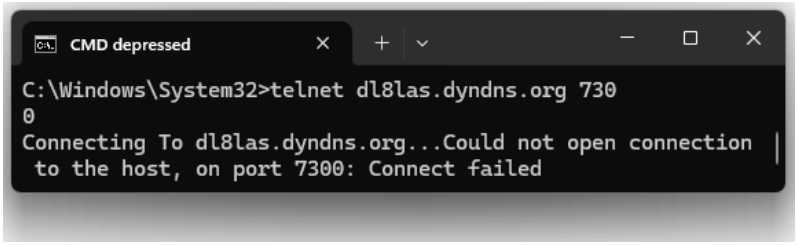
4. The Windows Telnet app then tries to connect to the DX cluster node.

If it connects, the node may display a welcome message, then prompt me for my login ID (*i.e.* my callsign) ▶



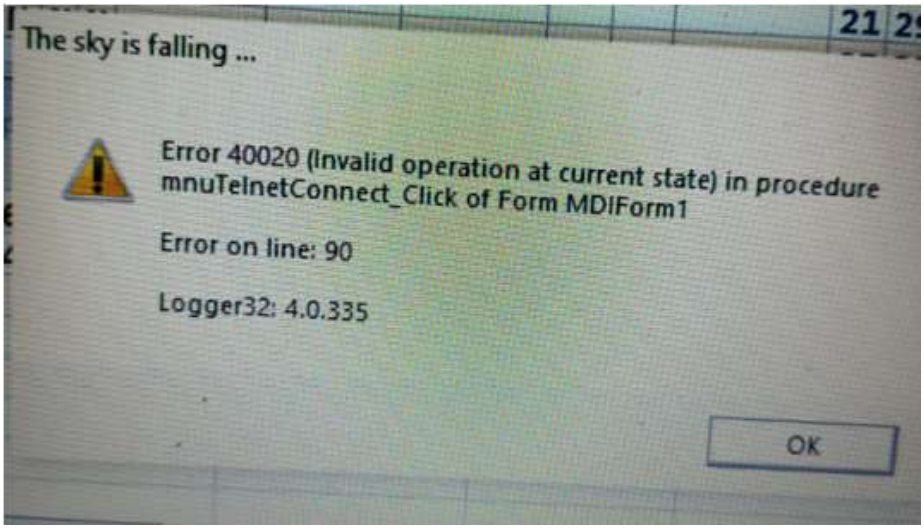
Copyright © 2026 Robert C. Furzer K4CY

Normally after entering my callsign, the DX spots start flowing thick and fast.[243] I type **quit** <**Return**> to logout. Seconds later, the Telnet connection is closed and the app terminates with "Connection to host lost." Entering a <**Return**> takes me back to the command line if I want to check another cluster node in the same way, otherwise I close the command window by clicking the corner ✕.



◀ If the node is *not* QRV, refuses the connection, or passes no spots on an active band, I have my answer and I close the command window using the corner ✕.

Q. Any idea how to get a Telnet connection? Logger32 displays this error message:



A. To trigger this error, do this:

1) Try to connect to a DXcluster that happens to be offline right now or no longer exists *e.g.* 21735.93.40 port 7373.

2) Nothing happens, so try to connect to a different DXcluster.

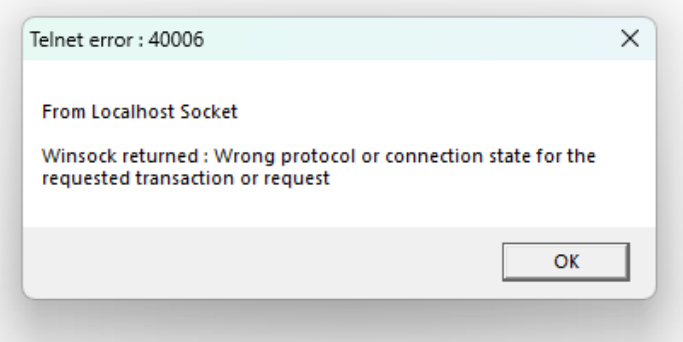
3) Logger32 displays error 40020 because it was still busy trying to connect to the original Telnet server when you asked it to make another Telnet connection. It threw its toys out of the pram.

To get around the error, click <**OK**> and **wait a couple of seconds** while Logger32 attempts to close or reset the failed Telnet connection. *Then* connect to a working DXcluster with a valid Telnet address.

[243] At this point I could check and adjust various settings for that DX cluster node – my band and mode filters, for instance – but since I struggle to remember the commands and syntax for the various DX cluster systems (including HELP!), I generally don't bother, preferring CC User's simple graphical interface instead.

Copyright © 2026 Robert C. Furzer K4CY

Q. I noticed my Localhost pane was empty, devoid of spots. When I tried to send the SH/DX command to the Localhost, I got the following incomprehensible error 40006. What's up doc?

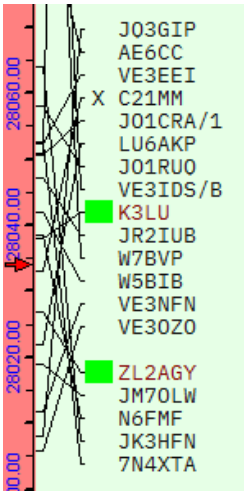


A. The Localhost pane is normally connected to a Telnet server running on the same PC, such as the one provided by CC User or a Skimmer. If CC User (or whatever) is not running *e.g.* if you closed it, if it crashed or locked-up, or if its Telnet port configuration does not match the Telnet port configuration in Logger32, it is disconnected from Logger32 … so your SH/DX command has nowhere to go … hence the comprehensible error message. QED.

Q. Why do my BandMaps sometimes resemble a cat's cradle?

A. The Windows function that is *supposed* to display the spots in frequency order sometimes gets itself terribly confused.

Here towards the bottom end of 10m, the VE3IDS/B label is shown in amongst the CW stations at 28046 kHz, linked by a near vertical line to its actual beacon frequency of 28153 kHz, way off the top of this image ▶



There's nothing we can do about this, except wait. The cat's cradle normally resolves itself when the BandMap is next updated, moments later.

It is rumored that Bill Gates resigned from Microsoft over this very issue.

Q. My BandMaps are empty. Spots are arriving normally but don't appear on the BandMaps. What's wrong with my setup?

A. Check your PC's clock: Logger32 compares the timestamp in each arriving DX spot against the current UTC time from the PC to determine whether it is a fresh spot worth displaying, or a stale one (older than the DX Spot persistence period). If your PC says it is 13:00 UTC, with a 10-minute spot persistence, any arriving spots timestamped *before* 12:50 UTC will not be added to the BandMaps as they appear to be stale. If for some reason (wrong time zone?) your PC clock is fast by, say, an hour, *all* the incoming spots will appear to be at least an hour old, as stale as a week-old loaf. Staler than a sitting politician's pre-election promises.

It is easy to check the accuracy of your PC clock against the current time: simply visit www.Time.is in your web browser. The website initially shows your local time as a big, bold digital display but click "UTC" further down the page to display the current UTC time instead. To correct your PC clock, go back to the Logger32 initial setup and/or section 2.4.

Copyright © 2026 Robert C. Furzer K4CY

Q. I hadn't noticed this before as there are very few 'new ones', but my alert sounds don't work. When I push the test button, nothing happens. Is it broke? Am I missing something? Am I the only one?

A. No you're not the only one. It's here because it qualifies as an FAQ. The most likely reason that audio alerts aren't sounding is that naughty Windows has spontaneously changed the audio card being used to sound them, perhaps following a Windows update. Aside from the lack of sound in the shack, you may notice the rig transmitting unexpectedly if the audio is being fed to the transmitter with VOX enabled, or you may hear audio alerts being transmitted over the air if you are monitoring your transmitted audio. It is not uncommon, unfortunately, to hear the Windows chime and spoken alerts being broadcast in SSB on the FT# sub-bands. So, listen to your rig's transmit audio monitor function at least once a day, and double-check your audio settings from time to time.

> **Hinson tip**: you might like to relax your spot filters to receive more alerts, either temporarily (for testing) or permanently. Adjust the settings until you reach *your* happy medium, somewhere between "*Far* too many annoying alerts" and "No alerts for ages – is it still working?". During a major contest, DXpedition or opening, expect to receive more alerts.

Q. What use is this if spots keep scrolling by, too fast to read?

A. Good question! The DX Spots pane can be a handy way to confirm that spots from various bands are arriving at the usual rapid rate from the DX cluster, but most of them are probably of little to no interest.

Logger32 can filter out spots for whichever band you are using and your selection of other bands, displaying them separately on the BandMaps. It can even highlight the few that are of particular interest such as your 'new ones' – typically new DXCCs, band/mode-slot fillers, DXpeditioners or pals from your club, stations that use LoTW, spots with IOTA or SOTA in the comments, or whatever.

If for some reason you feel the need to read all the spots as they flood-in, you can right-click the DX Spots pane and click to tick <**Hold off DX spots**> to divert incoming spots to a buffer rather than processing and writing them to the pane. After you are finished browsing through the spots shown, click to untick <**Hold off DX spots**> which cancels the hold, releasing a flood of buffered spots and continuing as normal.

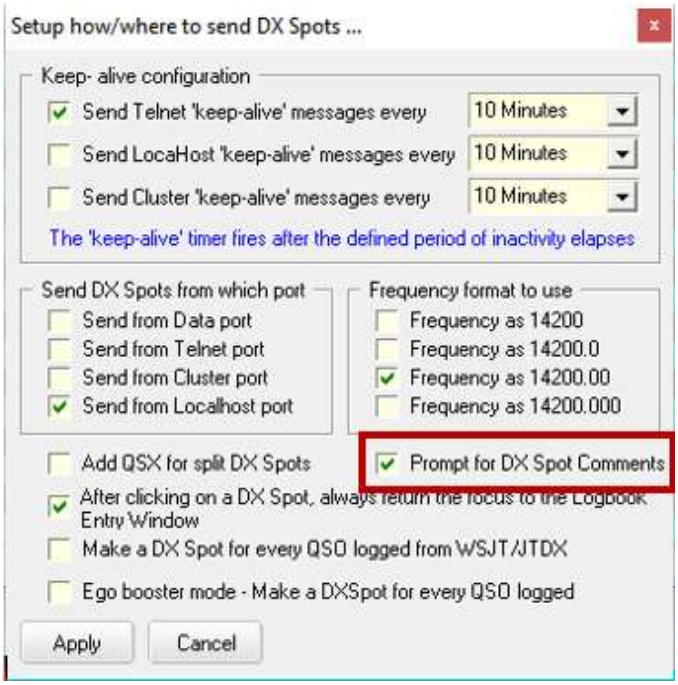
You won't miss any spots *unless* you leave the pane on hold so long that the Telnet buffer fills past the brim (and no, I don't know how many that is, except 'a lot').

While frozen, the caption's colored background flashes to remind you that the display is frozen solid, much like Captain Cook's ship Endurance in the remote Weddell Sea (quite some way South of the South Orkney Islands) back in 1915, before climate change was invented.

By the way, the same feature is available in the DX cluster pane and on the BandMaps.

Copyright © 2026 Robert C. Furzer K4CY

Q. How do I include comments in my DX spots?

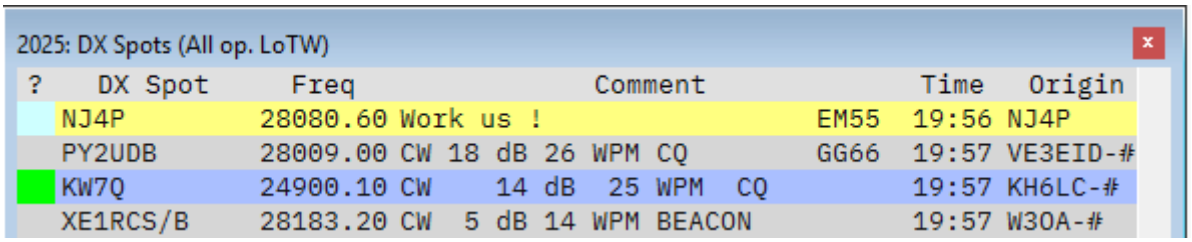


From Logger32's main menu, click **Setup ⇨ DX Spots** then tick <**Prompt for DX Spot Comments**> ◀ here on the right.

Having ticked that option, you will be invited to type in a short comment before sending each spot to DXcluster - or simply hit <**Return**> to send a plain vanilla uncommented spot.

Q. Why does my DX Spots pane caption start with the year?

A. It is a reminder that Logger32 is currently highlighting new ones you have not yet worked this year. Click the top left corner of your worked/confirmed table to toggle between all-time or current-year highlighting and gasp in amazement as the DX Spots pane caption, and any highlighting of spots, follow suit.

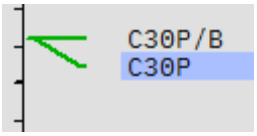


▲ Notice also that the DX Spots pane caption shows (in brackets) either just the current operator's callsign or "All op." (meaning all operators in the open log), followed by the types of confirmation for which new ones are being highlighted. We can change these options using **Config ⇨ Highlight**.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why doesn't Logger32 consistently highlight spotted DX stations from countries I *need*?

A. Several possible reasons. Likely culprits:



- They are beacons, generally spotted with callsign suffixes such as /B, /BCN, /B[digit] (*e.g.* /B1), /BEACON or /NCDXF – such as here[244] ▶
- They are flying or sailing *somewhere* in the world, using their home callsign with /AM or /MM respectively;
- They are (according to the spot comments or frequency range) using a mode that is not on your needs list on this band;
- They are on a band on which you have worked the country already – possibly in the distant past if you have selected <**ALL**> rather than the current year in the top left corner of your worked/confirmed table;
- Your DX spot filtering/alerting settings are not configured correctly for your needs.

Q. Why don't I see WWV/WCY updates?

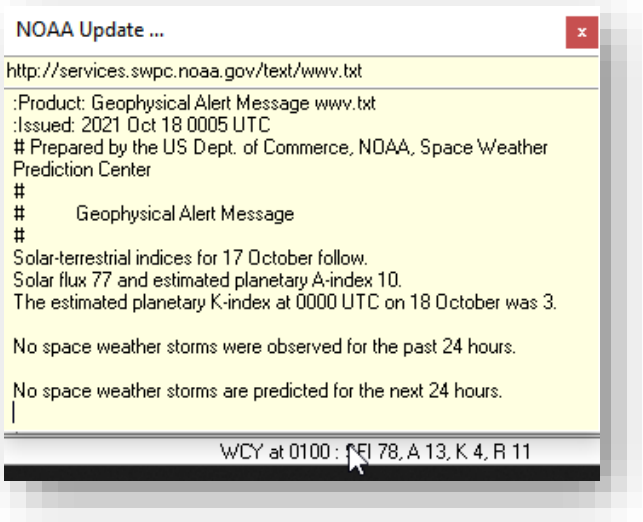
A. Solar and geomagnetic data are circulated every 3 hours on the DX cluster network. If you are not connected to the cluster, you won't see them. Doh!

When Logger32 receives a WWV or WCY report on its DX cluster connection, it writes the latest data and UTC timestamp to the status bar provided you have enabled **Setup** ➪ **Appearance** ➪ **Show WCY|WWV** in the DX Spots pane right-click menu – ideally showing *both* WCY *and* WWV ▼



Hinson tip: if you simply can't wait for the *next* report, send "SH/WWV 1" to DX cluster to grab the most recent one. Why not add this command to your cluster logon script?

By the way, right-clicking the WWV/WCY data panel gives you the option to view the current NOAA report ▶



It is a simple text viewer, not an HTML-aware mini-browser. If for some reason you change the URL, nothing appears to happen until you close and reopen the NOAA Update window, whereupon Logger32 grabs and displays the text from the new URL. To get back to the default URL, copy it from just above this paragraph, then close and reopen the window … and stop fiddling with the settings or you'll be sent to the naughty step. Again.

[244] Although I *needed* Andorra on that band (hence the plain C30P spot was highlighted), I could not have contacted the Andorra beacon (so the C30P/B spot was lowlighted).

Copyright © 2026 Robert C. Furzer K4CY

Q. The DX Cluster window is connected to various things, so the tabs show the names of the nodes, not the original tab names. Aside from disconnecting, how do I tell which tab is 'the Local Host tab', 'the Telnet tab', or whatever?

A. Look in the top left corner of the DX Cluster window. There you'll see the original generic name of the currently-selected tab. For instance:



Q. How can I tell which spots come from which DX cluster connection?

A. Logger32 merges and sends spots to the DX Spots pane and BandMaps from all the active, connected tabs on the DX cluster pane. By choosing different spot colors in each tab (*i.e.* right-click a tab label, then configure the colors for that tab), you will be able to determine a DX spot's source by its color.

Q. How can I setup a BandMap to show *all* HF spots?

1. Open your *C:\Logger32\ADIFBands.txt* file in a text editor and add a new line at the bottom for a band called GEN (as in GENeral coverage).
2. Add an entry to your Bands & Modes table covering the full range of your HF radio (*e.g.* from 0.003 to 30.000 MHz). Give the entry a band name of GEN.
3. Close and re-open Logger32 to pick up the change.
4. Monitor the band GEN in a BandMap to see all the spots across the whole HF spectrum.

Hinson tip: *loads* of spots circulate on the global DX cluster network every minute, far too many to display on one BandMap unless you have a display as tall as the Empire State Building, and excellent eyesight, or unless you are feeding your spots from a limited local DX cluster node or SDR. So, on the **Config** menu, you probably want <**Show only highlighted DX Spots**> to sort the wheat from the chaff. Another, complementary approach is to zoom the GEN BandMap to expand different parts of the range *e.g.* just the high bands during the day, and just the low bands at night.

Copyright © 2026 Robert C. Furzer K4CY

Q. *One* of my BandMaps has packed up. It remains empty, refusing to show any new spots for that band. What have I broken?

A. Work your way systematically through the DX spots processing flow diagram, from the end (step 11) to the beginning (step 1), checking things at each stage.

Do you see new spots arriving for *any* band? If not, maybe your DX cluster Telnet connection has failed, or the cluster server is down. First, check that you are still in fact receiving DX spots for that band by monitoring the DX Cluster pane. If not, maybe the band itself is simply dormant at the moment, or perhaps you have a band-blocking filter applied at the DX cluster end.

If you see spots arriving for the band, right-click the BandMap to see whether you have ticked the top option <**Hold off DX spots**>. That will definitely block the display of new spots.

If new DX spots are arriving for the band and <**Hold off DX spots**> is *not* ticked for that BandMap, move the mouse pointer away from the BandMap for a while and watch for updates. If the BandMap caption says "Freeze ….", wait until those dots disappear, then patiently await the arrival of a new spot.

If the new spot is not shown, check the BandMap's frequency scale. Is the spot frequency within the range shown? If not, adjust the BandMap's frequency range *e.g.* by pointing the mouse at the frequency scale and rolling the mouse wheel back to zoom out all the way. The new spot should then appear.

If nothing works, nothing you do seems to bring the BandMap back to life, something may have gone wrong in the code behind the screen … so try closing and re-opening the BandMap to reset it. If that doesn't help, close and restart Logger32. If it *still* doesn't work, cross your fingers, reboot your PC and try yet again.

If you've failed at every prior step and sadly ended up here, dejected, your last resort is to send a desperate plea for help to the Logger32 reflector.

Q. Having set <Hold off DX spots> on a BandMap a while ago, the BandMap has cleared … leaving no callsigns for me to right-click and un-set <Hold off DX spots>. What now, Einstein?

A. Don't worry, right-clicking the BandMap area where callsigns *normally* appear still opens the same menu. You don't need to right-click a real callsign. An imaginary one will do. How about, say, P5DX? EZ1AAA?

Q. Some of my BandMaps appear to be frozen or blocked. Can I reset them all?

A. Yes: close all the BandMaps by clicking the corner X on the current band's active BandMap (not just the **MON**itor BandMaps). Now reopen the set by clicking the setsquare icon #15 

◀ on the toolbar and they should all spring magically back to life, rejuvenated, keen to serve.

Copyright © 2026 Robert C. Furzer K4CY

Q. I'm a CW nut. When my BandMaps are using 'mode segments', why do I still see all the spots, not just my lovely lovely CW spots?

A. The 'mode segments' are based on your Bands & Modes table. CW is generally permitted and defined for the entirety of every amateur band, so you are 'zooming in' on the entire band! Instead, disable 'mode segment' and simply zoom in on the lower end of the frequency graticule using the mouse scroll wheel.

You could also try filtering out non-CW DX spots (ideally on the DX cluster/s you are using, otherwise in the DX Spots pane in Logger32) if you really are a CW nut.

Q. I am a 10m CW nut. Normally I zoom-in the 10m BandMap to cover the lower 300 kHz of the band including the CW beacons … but when I turn off my radio to save the planet, the BandMap un-zooms, so when I turn it back on, it takes a *lot* of mouse-wheeling to recover the original zoom level. Why is that? Is there anything I can do to avoid index-finger RSI, apart from leaving my radio on 24x7?

A. It's a CAT-n-mouse thing.

When you turn off your radio, the CAT becomes unresponsive (plays dead) so Logger32 presumes the radio's frequency has dropped to 0 Hz which is not within the 10m band, triggering the BandMap to un-zoom.

As a workaround, after you turn the radio back on and the CAT is revived, hold down the <**Shift**> key while you roll the mouse wheel to zoom in again on the BandMap.

Remember, **a shifty mouse zooms quicker**.

Q. No spots! *No spots!* I am bereft of spots! Help, I'm lost without them!

A. Try these, in sequence:

- Check that you still have a functional link to the DX cluster network - most likely an Internet connection, more specifically one that works for Telnet. If not, fix it. [That's out of scope of this Logger32 manual. Good luck.]
- Check whether the DX Cluster window is actually connected to a DX cluster. If not, connect.
  - If you can't connect, or if you connect but there are no spots arriving, the cluster may be broken. If it is a members-only cluster, you may be using the wrong password to login. Maybe try a different, less-broken, more open one or contact the sysop.
- Check/adjust the DX cluster filters at whichever cluster/s you are using: it's possible there are simply no spots at the moment on the particular bands and modes being passed by the cluster outbound filters and sent to you. Perhaps conditions are truly lousy.
- Likewise check Logger32's spot filtering. For example, if for some curious reason you are blocking incoming spots from ALL continents, you should anticipate a *long* wait until DXers on the moon start sending spots …
- Be patient. After all, topband SSTV DX spots from Antarctica don't come along *that* often.
- Catch measles, become a hormonal teen or paint a ceiling.

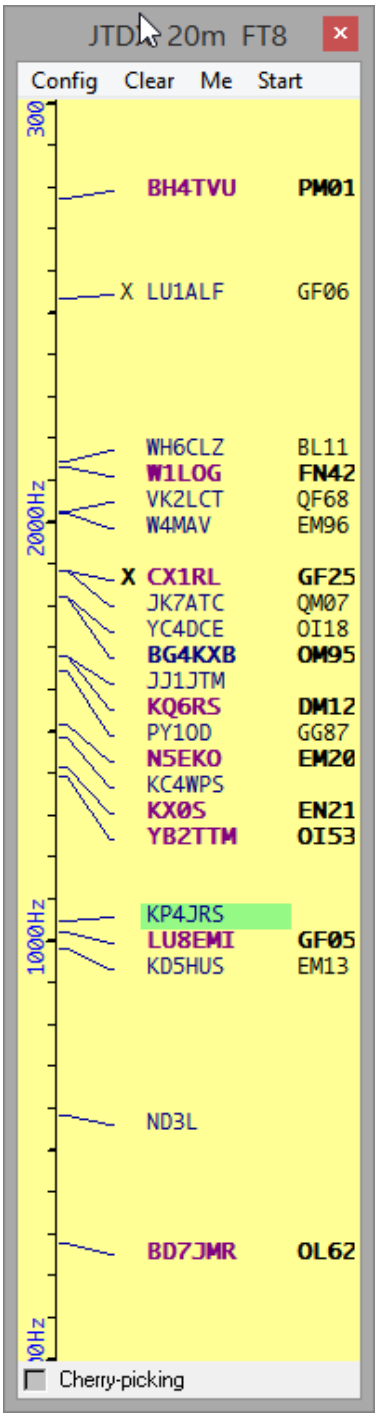
Copyright © 2026 Robert C. Furzer K4CY

15 The UDP BandMap and FT#



> "Networking is not about collecting contacts. Networking is about building relationships"
>
> *Heidi Roizen*

Logger32 can receive and display the callsigns and grid squares decoded by digimode software on its UDP BandMap ▶



As with Logger32's regular BandMaps, this is far more than just a pretty display. It's a graphical user interface with which we control both Logger32 and the digimode software – specifically **JTDX and WSJT-X used for digital data modes such as FT#**.

The 'band' being mapped here is in fact only a small part of the amateur band: specifically, that range of audio frequencies emerging from the receiver that gets passed through the PC sound card/audio device to the digimode software: hence the *audio* frequency scale on the left side.

Decoded digimode messages are periodically broadcast by the digimode software (if so configured) through your shack network via UDP (**U**ser **D**atagram **P**rotocol). Aside from decoded messages, when a digital QSO is completed and logged in the digimode software, the logged QSO details can also be broadcast via UDP. Provided Logger32's UDP port (socket) is open, it may receive those QSO details, look up stations worked, log the QSOs and optionally spot the QSOs on DX cluster (if so configured).

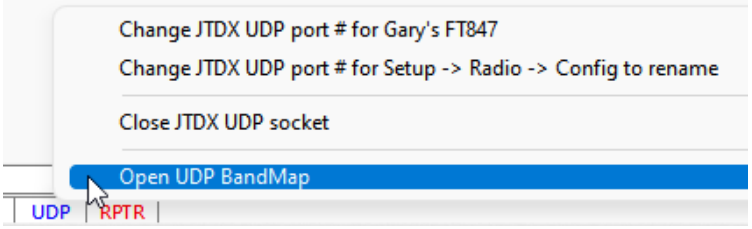
Up to five 'QSO logged' UDP messages can be queued for processing. QSOs are normally logged a few seconds after the messages are received by Logger32. If the messages are from software *other* than JTDX, no additional information from the log entry pane or previous QSO mask is added, and it is not even necessary to have the UDP BandMap open: so long as its UDP socket is open, Logger32 is patiently waiting to receive, interpret and process those 'QSO logged' messages.

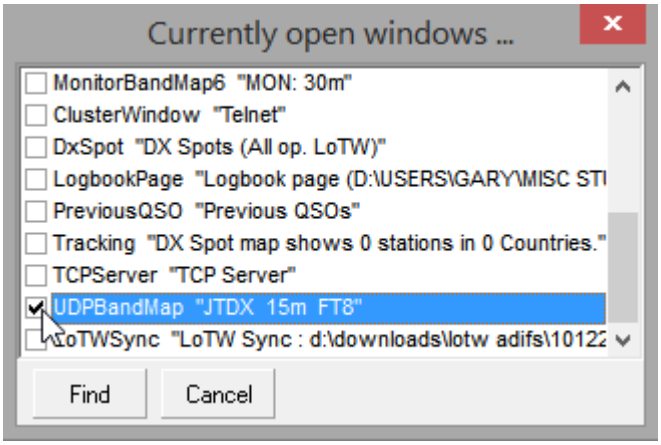
Logger32 can also send so-called 'reply' messages to JTDX, causing the digimode software to react as if you had double-clicked a callsign directly in its decoded messages panel by calling someone for you.

Copyright © 2026 Robert C. Furzer K4CY

15.1 Open UDP BandMap

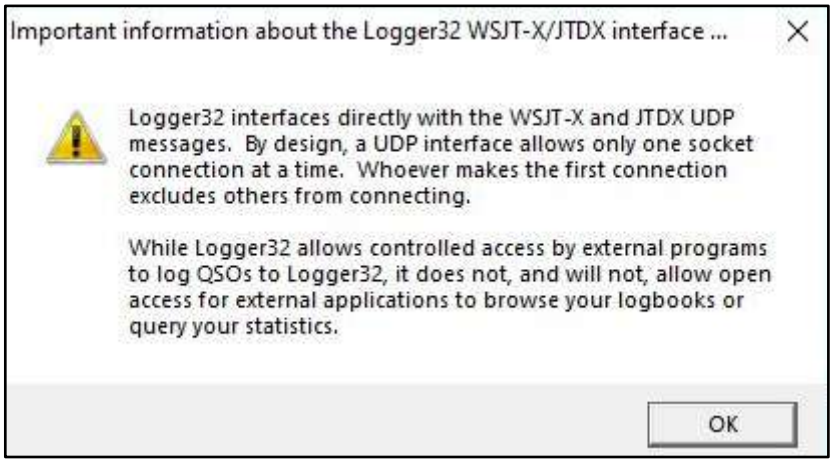
Open the UDP BandMap using <**Ctrl+J**> or by right-clicking the <**UDP**> panel on the status bar then click <**Open UDP BandMap**> ▶





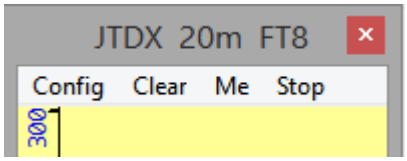
"UDP" turns blue when Logger32's UDP socket is open, waiting for incoming network messages. Notice that you can open the UDP socket with or without displaying the associated UDP BandMap. The UDP BandMap may also get lost – perhaps hidden behind another window or off the edge of your monitor: **View ⇨ Find lost windows** should reset the window to the top left of the main screen, if it is in fact open.

The first time it opens, a message points out a deliberate design constraint in how Logger32 handles UDP traffic and commands ▶



Hinson tip: the UDP socket is *solely* intended for use with digimode software such as JTDX, not as a general-purpose network interface into your Logger32 logbook for other kinds of program.

15.2 UDP BandMap menu



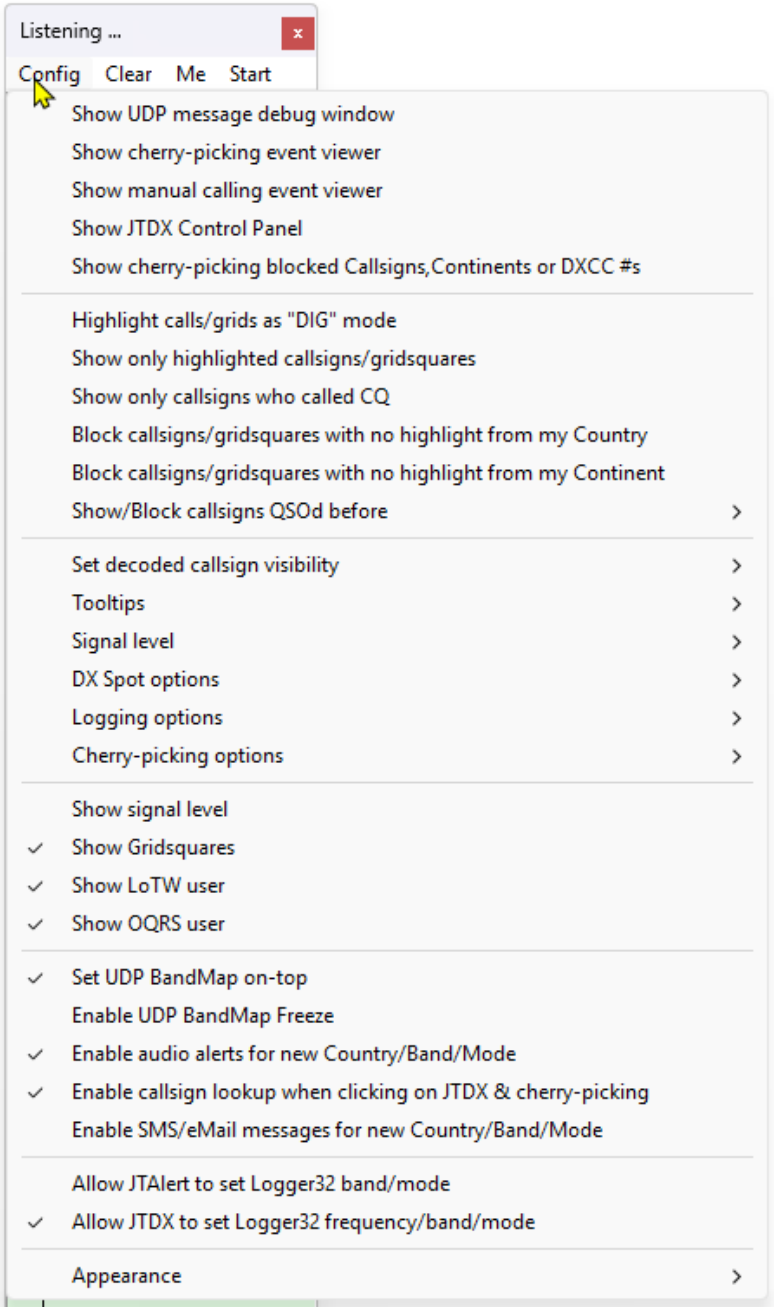
◀ The top line menu on the UDP BandMap has 4 items: Config, Clear, Me and Start|Stop.

… Read about each of those below …

Copyright © 2026 Robert C. Furzer K4CY

15.2.1 UDP BandMap <Config> menu

There are a ***lot*** of configuration options and submenus in the <**Config**> menu ▶



With *eight* submenus and yet more beneath, clearly there are *loads* of configurable items for the UDP BandMap because it is such a complex, flexible function in Logger32.

- **Show UDP message debug window:** helps identify, diagnose and fix problems in the UDP messaging between Logger32 and JTDX.
- **Show cherry-picking|manual calling event viewer:** these two monitor/debug functions are useful to diagnose problems in the way Logger32 attempts to initiate and complete QSOs with selected stations ('cherries') via JTDX, either automatically or manually.
- **Show JTDX Control Panel**: see below.
- **Show cherry-picking blocked Callsigns, Continents or DXCC #s**: see below.

Hinson tip: callsigns that magically appear on your UDP BandMap are *not* "DX spots" *i.e*. callsigns that have been copied by various other DXers around the world then shared through the DX Cluster network. ***All* the callsigns shown on the UDP BandMap have been decoded from transmissions received over the air on *your* antenna, receiver and computer.** Whereas inevitably some of the DX stations spotted on DX Cluster are inaudible at your QTH right now, *every single callsign currently on your UDP BandMap has just been copied by the digital mode software on your system within just a few kilohertz within the last few minutes*. Impressive stuff!



Copyright © 2026 Robert C. Furzer K4CY

- **Highlight calls/grids as "DIG" mode:** whether you have logged a DXCC entity or grid square on *any* digital mode, as opposed to the particular mode currently in use, determines the highlighting, tooltips and "QSOd B4" X mark *e.g.* if you have never worked Greece on FT4, but you *have* worked Greece on FT8 or RTTY, with this option enabled a Greek station using FT4 will *not* be highlighted as a new one … because you already have a digital QSO with Greece. Disable this option to treat each mode independently, and have fun chasing the Greeks on every mode and submode you possibly can.
- **Show only highlighted callsigns/gridsquares:** if the digimode sub-band is busy, humming with activity, the UDP BandMap can easily fill with decode callsigns hence spots for 'new ones' might not appear at all, or may be nudged out quickly by other – ordinary, uninteresting – decoded callsigns. Enabling this option hides the boring ones, leaving just the 'new ones' (whatever you have configured that to mean *e.g.* DXCC entities never worked, or not worked this year, or not on this band and/or this mode; also any callsigns or DXCC entities for which alerts are in place, and current DXpeditions). If any 'new ones' had been hidden from the BandMap due to the lack of space, they will *probably* now be shown *if* there is space and *if* they have not yet gone as stale as a long-forgotten half-loaf in the corner of a students' digs.
- **Show only callsigns who called CQ**: is another way to trim down the number of decoded callsigns on the UDP BandMap. Some feel there is little point even seeing the callsigns of stations who are only searching and pouncing on the stations that interest them, never CQing, since if you call them, they are unlikely to respond to you (assuming you are not P5DX![245]). In contrast, anyone who CQs[246] is clearly open to being called, ideally when they next call CQ or at least towards the end of their QSOs (when they send RR73 or 73 messages). They still might not respond, assuming they even copy your call, but only showing CQing stations is a rational space-saving distraction-reducing strategy, useful when the band is busy.
- **Block callsigns/gridsquares with no highlight from my Country|Continent:** yet another way to filter out humdrum spots on the UDP BandMap is to hide spots for your fellow countrymen or continental neighbors, *unless* they qualify as 'new ones' meaning they are highlighted. Chances are, an Irish HF DXer isn't particularly interested in working other Irish hams, perhaps not even other Europeans. That may not be true on, say, 6m or VHF/UHF though, or LF for that matter.

> **Hinson tip**: there are a bewildering number of menu options to explain in this detailed section. As well as following the sequence of the menu items throughout this manual, I have tried to use Logger32's exact wording consistently, so if you want to read about a particular menu item, you might **F**ind it easier using <**Ctrl+F**>, then type in the item as worded to search the manual for it.

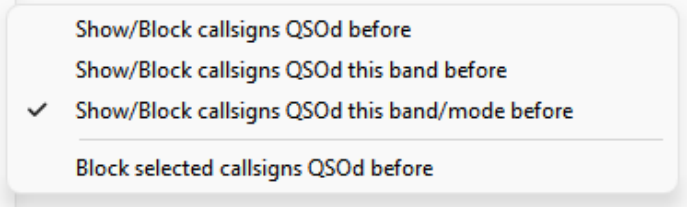
[245] If P5DX is, in fact, searching and pouncing on stations as a way to avoid being overwhelmed with responses to a CQ, enabling this option will prevent him from appearing on your UDP BandMap and triggering your 'ultra rare DX' audio alerts and SMS/email messages. If you are attentive and notice a flurry of activity, you *may* see his callsign appear in decodes flowing through the lefthand band activity pane in JTDX, and no doubt he will be spotted, excitedly and repeatedly, on the regular DX cluster network, appearing on your normal BandMap if you are connected … but by then you may have missed your chance to work him before all of hamdom descended on him!

[246] Callsigns of stations who have CQ'd can be picked out in bold on the UDP BandMap, so you can perform this filter by eye. Look for the option under **Config ➪ Appearance ➪ Highlight** …

Copyright © 2026 Robert C. Furzer K4CY

- **Show/Block callsigns QSOd before** submenu:

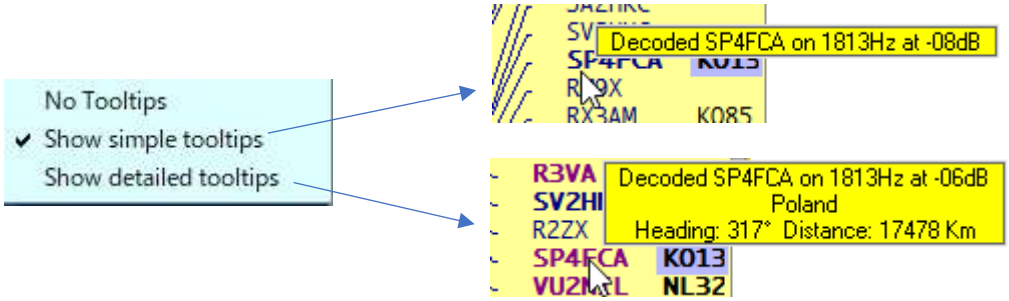
  The wording is confusing … but … the idea is *first* to select one of the 3 upper filtering criteria, then either *block* callsigns that meet the criterion by ticking <**Block selected callsigns QSOd before**> at the bottom, or *show* them by unticking it. As seen ▶ here, I am perfectly happy to work the same stations I have worked before on the same band and mode again, no worries. *My* QSOs aren't rationed. That said, being a DXer, I am less keen to work dupes than I am to chase new ones. It's a matter of individual priorities. Yours may well differ from mine.



- **Set decoded callsign visibility** submenu: determines the maximum[247] time spotted stations remain on the UDP BandMap since they were last decoded. The choices range from 30 seconds to 9 minutes, with an option to define even longer timeouts up to 8 hours. On busy, dynamic bands such as 20m, 30 seconds works nicely for me. If you are monitoring quiet, slow-moving bands such as 6m, you may prefer longer periods since stations decoded on brief openings or peaks in QSB may well hang around patiently waiting for a call. A given DX station's timer gets reset every time it is decoded hence it can remain on the UDP BandMap indefinitely so long as it is still being received well enough to be decoded (subject to there being sufficient space on the screen to display all the spots on the UDP BandMap).

> After each decode period, JTDX sends messages to Logger32 about stations it has decoded. These messages are placed into an internal array which is then sorted with newest messages at the top and any duplicate messages removed. Any changes to the array (such as new callsigns or gridsquares) triggers Logger32 to reload the UDP BandMap: the BandMap is cleared, and then, starting with the most recently decoded callsign, unexpired callsigns are transferred one-by-one from the array to the UDP BandMap unless it runs out of room.

- **Tooltips** submenu: when you mouseover a callsign on the UDP BandMap, Logger32 can display simple (callsign, audio offset, strength) or detailed (adding the country, heading and distance) tooltips about that station▼

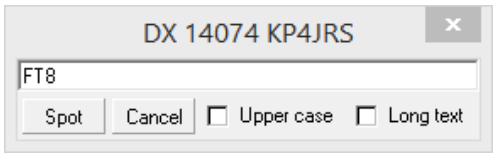


- **Signal level** submenu: choose whether to display signal strengths as **SNR** (signal-to-noise ratios expressed in dBs, as used with most digimodes) or **S-units** (calculated on a 6 dB-per unit scale where S9 is notionally equivalent to a 50µV signal on a 50Ω impedance receiver antenna input, as used with most legacy modes and radio S-meters).

[247] As the BandMap fills up, older stale entries may be displaced by fresher ones even before their time expires if there is no room to display them all.

Copyright © 2026 Robert C. Furzer K4CY

- **DX Spot options** submenu: you can right-click in the decoded grids square/report area of the UDP BandMap to generate a DX spot for the adjacent callsign. This configuration submenu gives you a choice of spot formats:
  - **Simple DX Spots** have the usual DX spot info (callsign and VFO frequency) and a comment starting with the mode ▶ You can edit the comment (including the mode, if you need to) before hitting <**Spot**> to submit it to DX cluster. Force the text to <**Upper case**> if your Shift or Caps Lock keys have mysteriously stopped working and you want to SHOUT your spot, and select <**Long text**> to `stretch it out`[248].

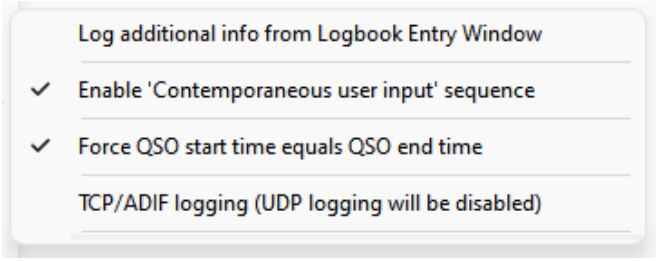


  - **Formatted DX Spots** are fully composed (pre-formatted) for you with the DX callsign and VFO frequency, the mode, signal-to-noise ratio, grid square (if decoded) and the audio offset *e.g.* ▼

```
EY8MM       14074.00 FT8 -08dB from MM48 481Hz       03:45  ZL2IFB
```

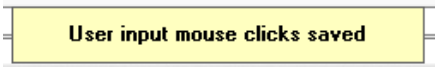
    Formatted DX spots are composed and sent immediately to DX cluster without the need for any further clicks … and no opportunity to add cheeky little notes of your own.

- **Logging options** submenu ▶
  - **Log additional info from Logbook Entry Window**: tick this to log TX_PWR, QTH, ADDRESS, COMMENT, NAME, STATE, and CNTY fields in the log entry pane or from online lookups as well as the basic QSO information.



> **Hinson tip**: if the gridsquare is known for the station you are logging (*e.g.* if it was transmitted as part of the usual FT# CQ call, or if you have logged the grid in a previous QSO with the same station), tick this option to calculate and log the distance from your home QTH.

  - **Enable 'Contemporaneous user input' sequence**: since QSOs must involve "contemporaneous direct initiation by the operator on both sides of the contact" to qualify for ARRL contests and awards such as DXCC, Logger32 can *temporarily* cache your mouse clicks to replay on demand. It starts doing so when Logger32 start up, displaying a pop-up message in the lower right corner of the screen

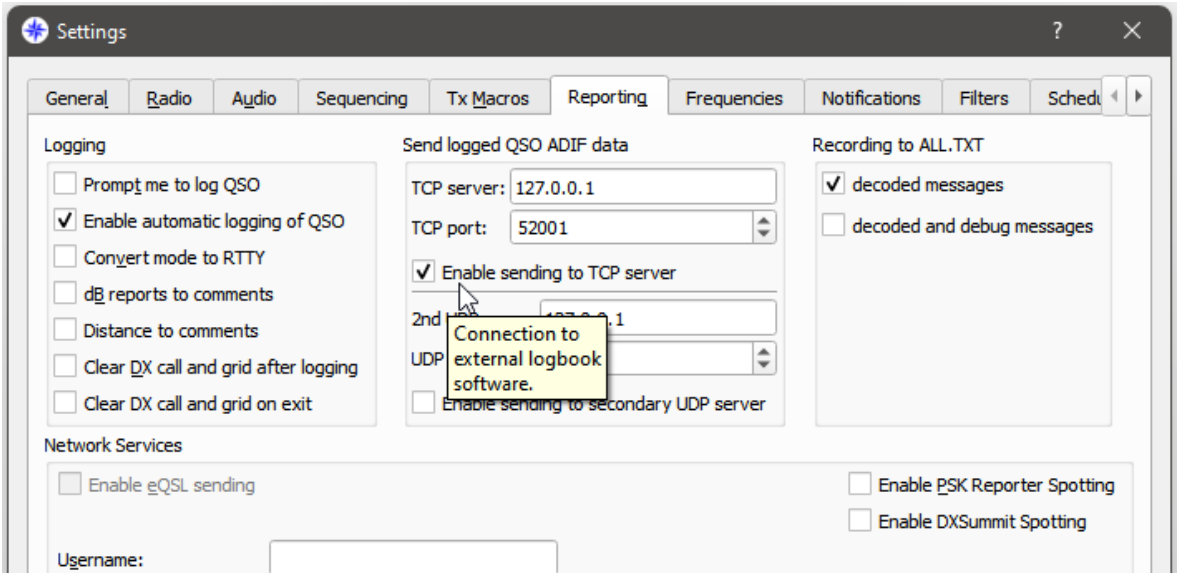


    ▲ With <**Enable 'Contemporaneous user input' sequence**> selected, one of your cached clicks is replayed to log each JTDX or WSJT-X QSO in Logger32's open log, as if by magic.
  - **Force QSO start time equals QSO end time**: QSOs sometimes take quite a while to complete, from the initial call to the final RRR, RR73 or 73 message. This option treats the end time (when a digimode QSO is logged) as reasonably definitive and probably very similar for both parties, then makes the QSO start time the same (rather than when you *actually* first started calling someone, which may have been some long time before).

[248] Not really. Select <**Long text**> if you want to send comments of more than 30 characters … despite knowing that some clusters won't accept them and crude column-aligned cluster clients may not display them properly. Otherwise, your PC dings a warning and steadfastly refuses to accept the 31st character.

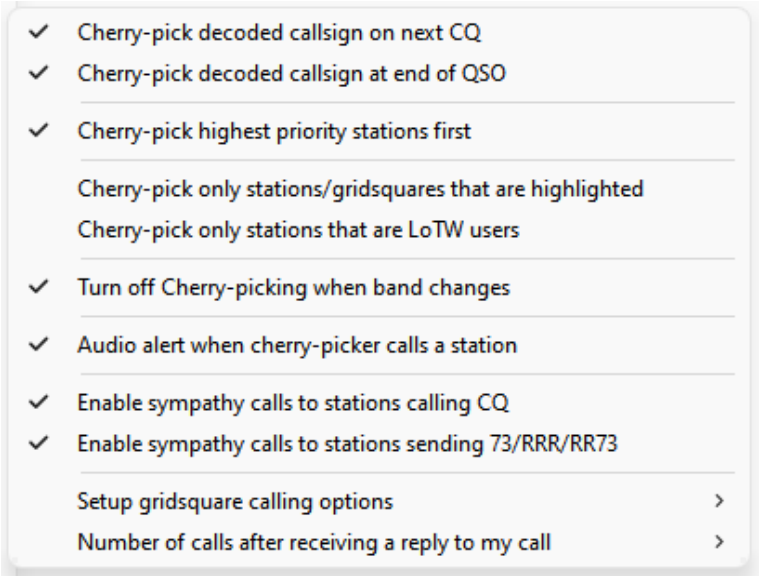
Copyright © 2026 Robert C. Furzer K4CY

- **TCP/ADIF logging (UDP logging will be disabled)**: select this option on the *UDP* BandMap (!) to enable Logger32 to receive and log QSOs in ADIF format via TCP *instead of via UDP*. You'll need to configure JTDX to send logged QSOs via its TCP server function as well ▼



> **Hinson tip**: Logger32's routine/periodic log backups (if so configured) are automatically and silently **disabled** while the cherry picker runs, in order to conserve CPU resources for processing digimode signals. You have been warned. Backups when Logger32 closes down should still work OK, and you can always click icon #1 to make a backup manually at any time.

- **Cherry-picking options** submenu with eight options plus two further sub-submenus ▶



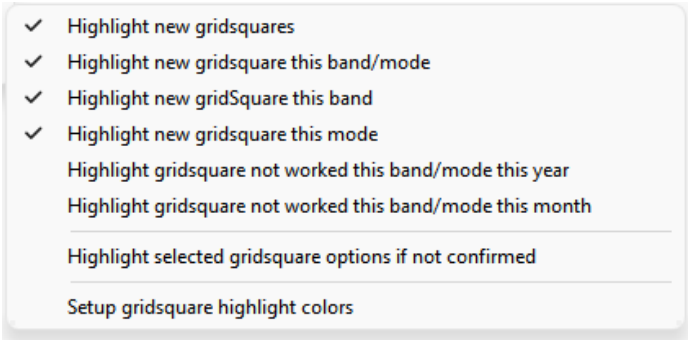
  - **Cherry-pick decoded callsign on next CQ**: a selected station will be called when he CQs.
  - **Cherry-pick decoded callsign at end of QSO**: a selected station will be called when he sends an RR73 or 73 message, without waiting for him to CQ.
  - **Cherry-pick highest priority stations first**: given several possible stations to call, Logger32 decides which one is the most attractive (the ripest cherry!), and calls him.
  - **Cherry-pick only stations/gridsquares that are highlighted**: instructs Logger32 to ignore anyone that isn't highlighted as a 'new one' – a genuine cherry, provided you have correctly configured Logger32 to highlight only the stations you want/need *e.g.* new DXCCs, new gridsquares.

Copyright © 2026 Robert C. Furzer K4CY

- **Cherry-pick only stations that are LoTW users**: preferentially working stations known to have used ARRL's **L**ogbook **o**f **T**he **W**orld increases your chances of having your digimode QSOs confirmed quickly, but reduces your chances of working new ones that are *not* known to be LoTW users (such as DXpeditions that have not yet signed and uploaded their first logs).
- **Turn off Cherry-picking when band changes**: this is *a jolly good idea* (and the default setting) because: (1) it gives us the chance to change everything that needs to be changed when changing bands, before attempting to call any cherries – things such as twiddling the knobs on our manual amplifiers, antenna switches and tuners, and finding a suitable transmit frequency on the waterfall; and (2) re-enabling the cherry-picker presumably qualifies as a 'contemporaneous input', so any valid new countries we bag and get appropriately confirmed should qualify for DXCC. Hopefully. Anyway, un-tick this option to let the cherry-picker robot continue hunting for cherries when you change bands, without taking a digital breath;
- **Audio alert when cherry-picker calls a station** is a feature requested by a dedicated 6m DXer, accustomed to monitoring the FT8 watering-hole on a deserted band for long tedious periods of nothingness until, magically, it springs to life for a few brief and exciting minutes. A beep when Logger32's cherry-picker starts finally finds someone to call is a handy reminder to pay attention to the hobby instead of gently snoozing in the corner of the shack or idly browsing social media.
- **Enable sympathy calls to stations calling CQ**: if there are presently no cherries (*i.e*. no 'new ones') to chase on the UDP BandMap, Logger32 can respond to other stations' CQ calls to fill-in the time and keep your station active.

> **Hinson tip**: being 'active' can be an effective technique. A stealthy DX station quietly monitoring the band might just notice your callsign and give you a call, but only if you transmit, or someone else calls you, or you are manually spotting stuff on DXcluster *etc*.

- **Enable sympathy calls to stations sending 73/RRR/RR73**: in the same fashion, Logger32 can tail-end other stations when they end their contacts in a conventional manner, without waiting for them to CQ[249].
- **Setup gridsquare calling options** sub-submenu: you can collect contacted grid squares around the globe in much the same way as for DXCC entities, IOTA islands or SOTA peaks. These options ▶ determine how the grid squares are highlighted for you on the UDP BandMap.

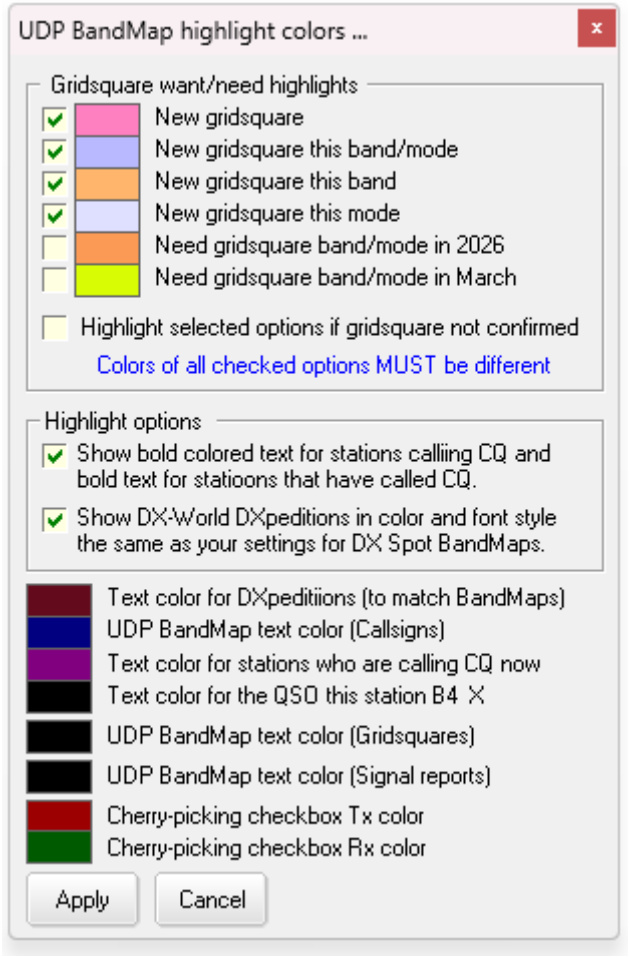


At the bottom of that submenu, <**Setup gridsquare highlight colors**> opens yet another submenu, the **UDP BandMap highligh colors …** form. The exact same form can also be opened using the UDP BandMap's **Config ⇨ Appearance ⇨ Highlight & text colors** menu option, saving 2 precious clicks.

[249] This option works well with JTDX. Due to a limitation deliberately built-in to the design of WSJT-X, it works on FT4 but not FT8 with WSJT-X. Don't shoot the messenger!

Copyright © 2026 Robert C. Furzer K4CY

The **upper section** of the form determines whether and what highlighting is applied to callsigns that are:



- **New gridsquares** *i.e*. we have *never* worked that grid before, never, not once[250];
- **New gridsquare this band/mode** – we may have logged the grid on a different band and/or mode, but *never* on the current band+mode combination;
- **New gridsquare this band** – we have *never* logged the grid on this band, not on any mode;
- **New gridsquare this mode** – we have *never* logged the grid on this mode, not on any band;
- **Gridsquare not worked this band/mode this year** – we may have logged it on a different band and/or mode, and possibly on the current band+mode combination in prior UTC years, but not yet this year;
- **Gridsquare not worked this band/mode this month** – we may have logged it on a different band and/or mode, and possibly on the current band+mode combination in prior UTC months, but not yet this month;
- **Highlight selected gridsquare options if not confirmed** does *not* highlight the grid (applying the criteria defined just above) if the grid has been worked, logged *and* confirmed, but only if it is presently *un*confirmed. Got that?

To change a highlighting (text) color, click a colored rectangle and pick a new color.

Hinson tip: I recommend using the same highlighting colors as for new DXCC entities, unless you enjoy being assaulted by a confusing kaleidoscope of colors for all the myriad possibilities. Your choice. Just sayin'.

The **Highlight options** mid-section of the form determines which stations' text is highlighted – those calling CQ, announced DXpeditions, both or neither. Make your choice and tick. Or not. The DXpedition text color can be set here or on the other BandMaps – they all use the same color. The CQer text color is set here ("Text color for stations who are calling CQ now", in the lower section).

The **lower section** lets us specify the text and highlight colors and other stuff.

Once configured to your liking, click <**Apply**> to save and use the settings, or <**Cancel**> to discard them and revert to whatever you had before.

[250] More accurately, the gridsquare has not been logged against any QSO under the present station callsign in the open logbook. You may have worked but not known about it (if the station never told you or confirmed his grid) or neglected to log it, or it could be in an archived logbook, or you may have used a different callsign. Or something.

Copyright © 2026 Robert C. Furzer K4CY

Number of calls after receiving a reply to my call is the final option on the **Config ⇨ Cherry picking options** submenu: if someone calls us, how many times should we respond to them before giving up in disgust if they don't respond? Choose 2, 3, 4 or 5 tries. 5 shows considerable patience.

> **Hinson tip**: if 2 is 1 too many tries for you, you'll have to assert your authority over the robot by stepping in to take charge of the technology, manually aborting the robot's second attempt. Go on, show it who's boss!

Returning now to complete this laborious description of the options on the **Config** menu:

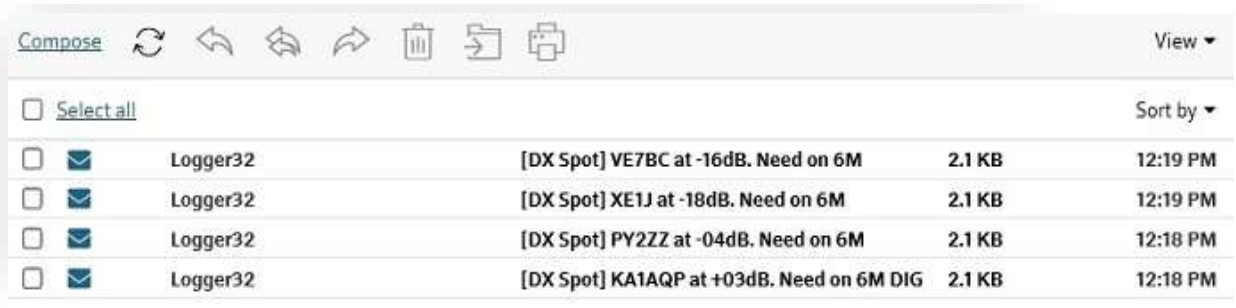
- **Show signal level**: display signal strengths, using the units defined above under the <**Signal level**> option *i.e*. signal-to-noise ratios (dBs) or S-units if you prefer.
- **Show Gridsquares**: on FT#, CQ calls typically[251] include the callsign and grid square of the station calling. If you take 'working the world' to mean making contact with every part of the Earth's surface, showing grid squares on the UDP BandMap, especially with highlighting for 'new grids', takes you a step closer to your goal.
- **Show LoTW|OQRS user**: colored blobs on the normal BandMaps beside stations who use LoTW and/or Club Log's **O**nline **Q**SL **R**equest **S**ervice can also be shown on the UDP BandMap, and can optionally be re-purposed to indicate membership of other clubs in the same way. Read more here.
- **Set UDP BandMap on-top**: stops the BandMap being obscured by other windows.
- **Enable UDP BandMap Freeze**: with this option ticked, when the mouse pointer goes into the UDP BandMap window, any window updates are delayed for a couple of seconds to stop things moving around, giving you a better chance of reading and maybe clicking your chosen callsign.
- **Enable audio alerts for new Country/Band/Mode**: audio alerts can be sounded for new ones on the UDP BandMap in the same way as for the regular BandMaps/DX spots.
- **Enable callsign lookup when clicking on JTDX & cherry-picking**: if you click a callsign in JTDX to call him, or if the cherry-picker calls him, Logger32 does its usual Internet callsign lookup and populates the log entry pane with his callsign, plus his gridsquare if that has been decoded (provided it is not then over-written by his previously-logged grid using the QSO mask feature, or from his callbook info from a callsign lookup). If un-ticked, the callsign is not looked up or shown in the log entry pane, but when the QSO completes, it drops directly into the log.

> **Hinson tip**: un-tick this option to let the cherry picker work cherries for you on a 6m radio in the background while – at the same time – you hunt for CW DX on your HF radio. Your HF DX hunting and callsign checks will not be rudely interrupted by any 6m digital excitement. You'll hardly know it is happening, in fact, except for the 6m QSOs appearing discreetly in your log … and maybe a puff of magic smoke as one of your receiver front-ends is blown by RF from the other radio …
>
> Oh and here's a top tip from N5KD Pete: it *is* possible to get callsign data into the logbook when calling JTDX stations which do not come through the log entry pane. The trick is to use TCP logging and tick <**Enable callsign lookup**> in the **TCP Event viewer ⇨ Config** menu.

[251] Except for some compound or special even callsigns which leave too few CQ message bits to encode the grid square as well. Unfortunately, this includes /MM maritime mobile callsigns, so unless sailors send their grids in another message or by another means, we don't know where in the world they are, nor which way to beam.

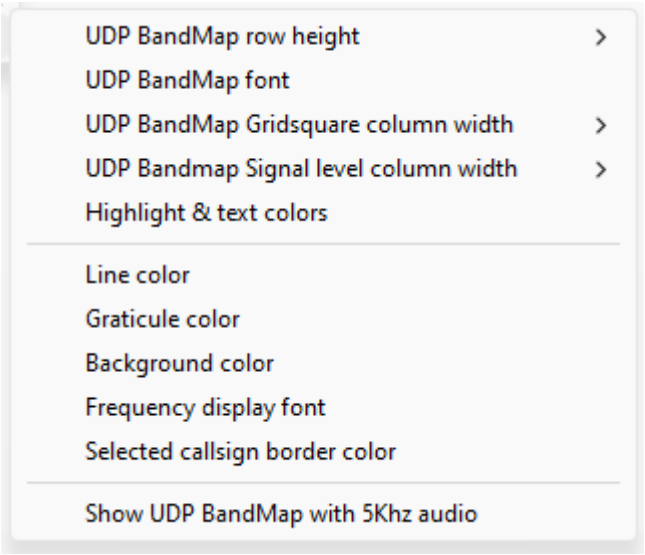
Copyright © 2026 Robert C. Furzer K4CY

- **Enable SMS/eMail messages for new Country/Band/Mode**: Logger32 can send you SMS/email messages when new ones are spotted if you first setup DX spot alerts (eMail/SMS/ScratchPad) in the DX Spots pane, and then enable this option. To avoid repeatedly annoying everyone around you, messages are sent just once per 15 minutes for each 'new one'. On the DX spot alert setup, all checkboxes are ignored except <**Check this if your email alerts are being blocked as spam**>. Examples ▼



- **Allow JTAlert to set Logger32 band/mode**: if you change band or mode in the digimode software, Logger32 follows suit, provided Logger32's CAT port is closed so that the digimode software can use *its* CAT connection to control the radio. See also the TCP client-server chapter.
- **Allow JTDX to set Logger32 frequency/band/mode**: *provided* you launch the digimode software using the UDP BandMap's <**Start**> button to set up the connection, Logger32 can take its current frequency/band and mode information from the digimode software rather than directly from the radio. While the digimode software has control of the radio through its CAT connection, Logger32 is out of control.
- **Appearance**: opens *yet another* submenu, with three *further* sub-sub-menus ▶

  The menu options are reasonably self-evident ways to adjust the visual appearance of the UDP BandMap … so, frankly, I can't be bothered to describe them here. Feel free to try them out for yourself *e.g.* extend the audio spectrum display to 5kHz if your radio has sufficient bandwidth. Tweak the UDP BandMap until just *before* the point you are making things worse, not better. Stop right there: you are wasting good digimoding time!



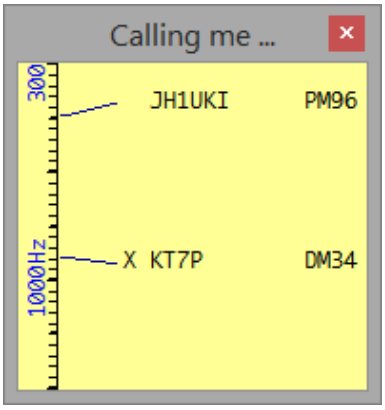
15.2.2 UDP BandMap *Clear* option

This empties the UDP BandMap, clearing the way for the *next* batch of decodes. It can be useful if you change bands/modes, and find the old UDP BandMap contents distracting. The UDP BandMap is not normally updated/redrawn until a new batch of decodes arrives, more specifically a batch with decodes from stations that are not blocked. If you are using **Config ⇨ Show only callsigns who have called CQ**, you won't see *any* new callsigns appear *until* a CQ message arrives.

Copyright © 2026 Robert C. Furzer K4CY

15.2.3 UDP BandMap *Me* option

Clicking <**Me**> on the UDP BandMap menu opens another UDP BandMap that only lists stations calling or working you[252] ▶

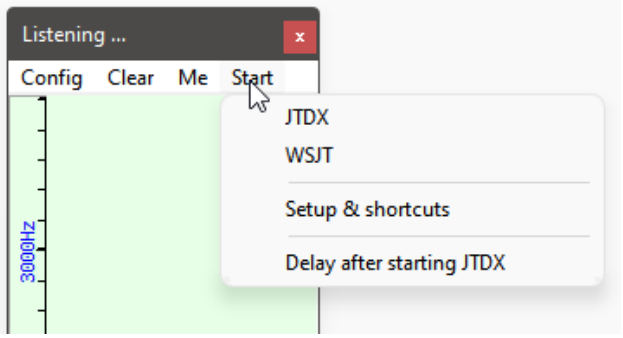


If you are using an attractive DX callsign on a wide-open band, filtering out the clutter of traffic between other stations makes it easier to focus on those calling and working you: who are you going to work next?

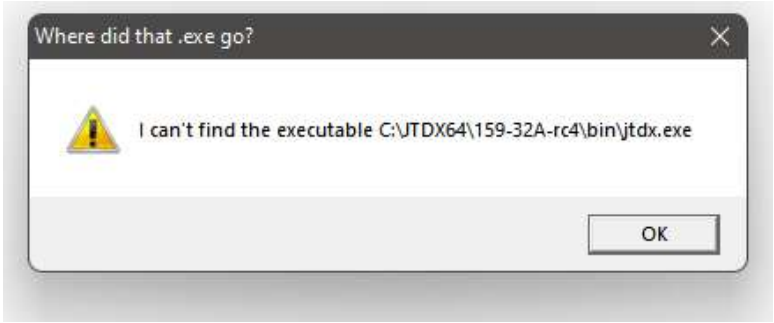
15.2.4 UDP BandMap *Start|Stop* menu

This is a cunning little dynamic flip-flop menu option. Initially it shows <**Start**>. Click <**Start**> and then the relevant option to have Logger32 launch, say, JTDX … whereupon the menu option flips to <**Stop**>. Click <**Stop**> for Logger32 to close JTDX for you, flopping the menu option back to <**Start**>.

The <**Start|Stop**> menu has several[253] options ▶



- **JTDX|WSJT *etc.***: starts|stops the respective program, applying the settings and macros from <**Setup & shortcuts**> then waiting for the <**Delay after starting JTDX**>. See below for the full nine yards on the start|stop sequence.
- If for some reason Logger32 cannot find the executable program to start, it says so ▼



[252] Unless you are *extraordinarily* popular, you can probably shrink the <**Me**> map to a fraction the size of the main UDP BandMap since it is unlikely to fill to the brim with your callers.

[253] JTDX is the preferred digimode software but you can define shortcuts to start and stop other programs instead or as well.

Copyright © 2026 Robert C. Furzer K4CY

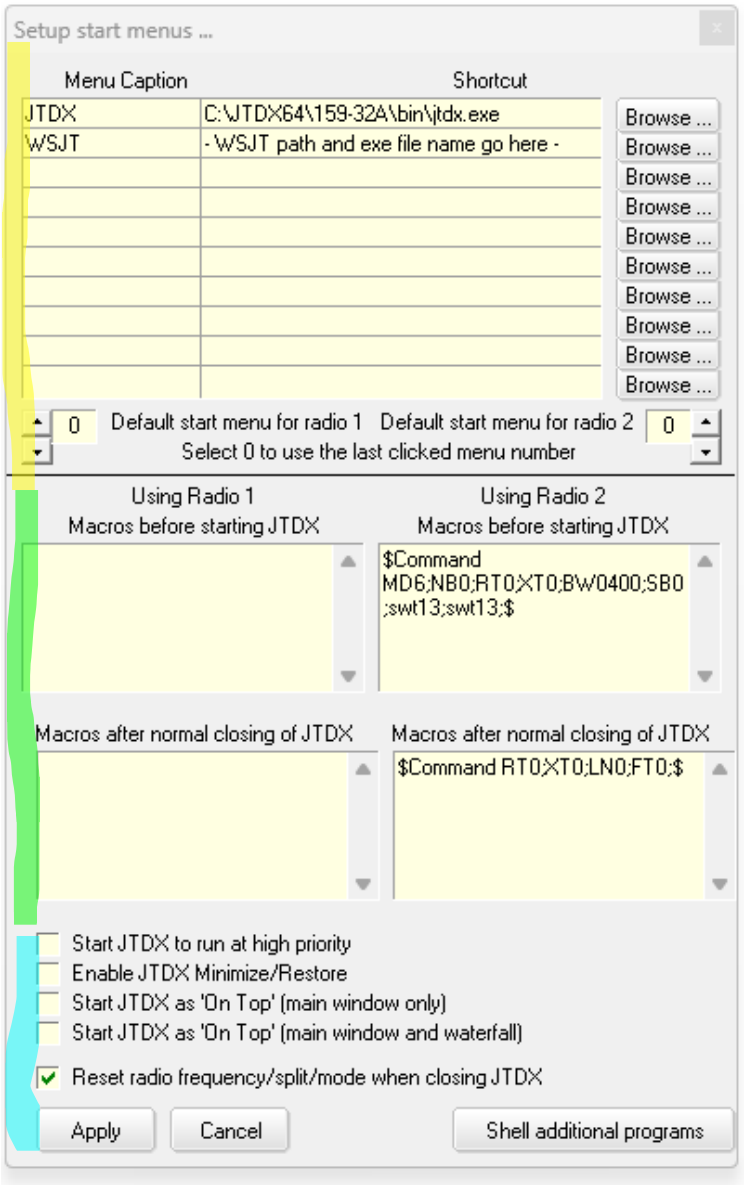
Did you make a mistake when typing the disk, folder or file name under <**Setup & shortcuts**> (see below)? Or have you updated, renamed, moved or removed the program since configuring the shortcut? Check that folder+filename string in the error message, carefully.

- If for some reason JTDX doesn't start up properly, or if Logger32 is unable to seize control of it once it is running, a message pops-up briefly near the system clock *e.g.* ▼



Hinson tip: try it again and maybe fiddle with the settings (*e.g.* increase the delay after starting JTDX) until it works more reliably. If nothing seems to work, rebooting may help. Or take a break. Brew a nice cup of tea. Ponder the meaning of life, the universe, DX and everything.

- **Setup & shortcuts**: opens a configuration form ▶



The **top section** of the form is where we tell Logger32 which digimode programs to start or stop through <**Start|Stop**>, and where to find the executables. Name up to 10 programs on the menu through the **Menu Caption** column, and either type the complete disk, path, file names and any command line parameters [254] into the **Shortcut** column, or click <**Browse**> to find the files using File **E**xplorer. The 'Default start menu' settings determine which line is executed for radio 1 and 2, and hence which program is launched, when Logger32 does a Quick Switch to FT#. Line '0' means use whichever program you last chose manually for that radio.

The **middle section** lets us define macros to set up our CAT-connected radios *before* Logger32 hands over control of the radio to JTDX, and *after* JTDX relinquishes control to Logger32 when it is closing. The syntax is the same as for the **R**adio **C**ontrol **P**anel – which is a convenient place to compose, try and refine the complete sequences. [The commands shown here are for my K3, turning off the sub-receiver and noise blanker, entering DATA mode,

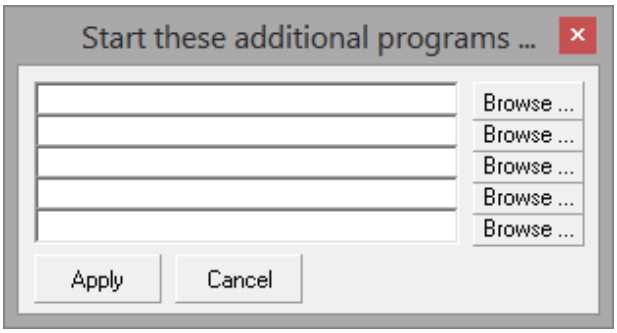
[254] You can open specific configurations of JTDX with the -rig command line switch: see the JTDX documentation for details (and good luck in your quest!).

Copyright © 2026 Robert C. Furzer K4CY

setting the IF bandwidth to 4 kHz and the output power to 11 watts when I start JTDX, and cancelling split when I close JTDX. Your setup and hence the commands required are probably different.]

In the **bottom section** of the form are five options and 3 buttons:

- **Start WSJT/JTDX to run at high priority:** decoding digimode messages within the few seconds available between receive and transmit periods requires some heavy-duty processing. Forcing the digimode software to a high priority *may* help ensure that processing DX spots, watching videos, reading and composing emails *etc*. takes second fiddle to digimode message decoding.
- **Enable WSJT/JTDX Minimize/Restore**: if selected (ticked), the digimode software follows Logger32's window state - in other words, if you minimize Logger32, the digimode software is also minimized, and both are restored if Logger32 is restored. Leave this option deselected if the digimode software windows should *always* be shown (the usual Minimize/Restore icons are not present on the JTDX|WSJT-X caption near the corner X if you launch it using <**Start**> on the UDP BandMap).
- **Start WSJT/JTDX as 'On Top' (main window only)**: prevents other windows obscuring the JTDX|WSJT-X main window showing decoded message, message buttons and controls. With this option ticked, the digimode software's main window is displayed on top of any other window … such as Logger32's windows, BandMaps, pop-up messages *etc*. To reveal content beneath, you can move the digimode windows aside, move whatever was hidden underneath to a different area of the screen, then move the digimode windows back – or (like me) do not tick the 'On Top' option in the first place!
- **Start WSJT/JTDX as 'On Top' (main window and waterfall)**: keeps *both* the main digimode software window *and* the waterfall on top at all times.
- **Reset radio frequency/split/mode when closing JTDX**: puts the radio back as it was before you opened JTDX for a bit of digimode action.

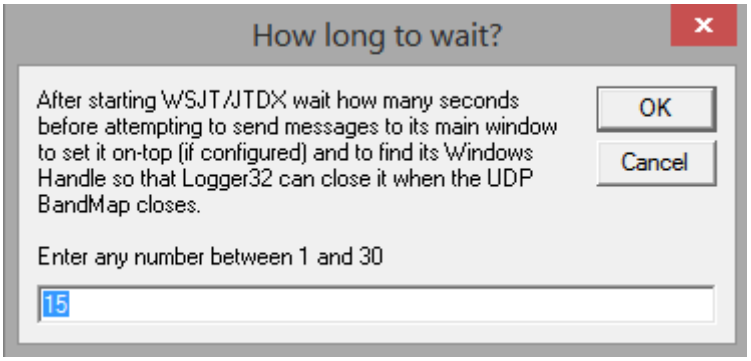


<**Shell additional programs**> lets you launch other programs (such as JTAlert) at the same time as launching JTDX.

◀ Browse to find and select a program, click <**Open**> to grab the file name, edit the line to add any launch parameters and <**Apply**> to save the new configuration.

- **Delay after starting JTDX**: the digimode software needs a moment to compose itself after it starts running.

This option gives it a breather of up to 30 seconds before Logger32 interacts with the software ▶

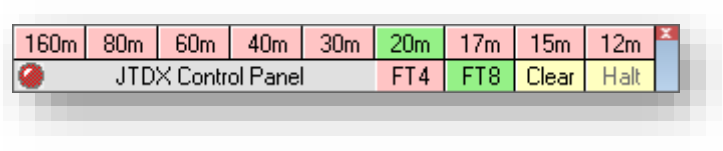


Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: adding additional program lines to the <**Start|Stop**> menu makes it easy to run different digimode programs or just different versions of the same digimode software. I normally install JTDX beta releases in separate folders, add them to the UDP BandMap start menu, then run and compare them with previous versions in the hunt for new features/bugs and to check out supposed bug fixes.

15.3 JTDX Control Panel

The JTDX Control Panel ▼ lets us control JTDX from within Logger32.



The red "LED" and green buttons on the JTDX Control Panel show that, at the instant I grabbed this image, Logger32 knew that I was transmitting FT8 on 20m.

Provided that: (a) The JTDX Control Panel was visible (use **Config ⇨ Show JTDX Control Panel** from the UDP BandMap menu); (b) We launched the digimode software using **Start ⇨ JTDX** from the UDP BandMap menu; and (c) Within the digimode software, the focus is on the band activity pane[255], we can command the digimode software to change bands and modes simply by clicking cells on Logger32's JTDX Control Panel, as opposed to using the band and mode selectors within the digimode software or pressing buttons on the radio. Shortly after being clicked, the JTDX Control Panel buttons change color when the digimode software dutifully confirms to Logger32 that the radio has indeed changed band or mode as commanded.

Which of up to 13 band buttons are shown, plus the mode and band button colors (selected and deselected[256]) and "font styles", are all configurable[257] by right-clicking the rectangle below the corner X [258]. We can also move the JTDX Control Panel by clicking the rectangle and (with the mouse button still pressed) dragging the panel somewhere else on the screen.

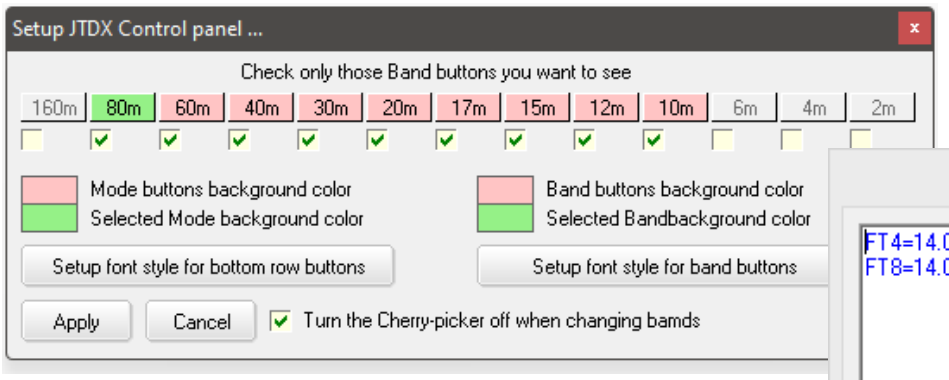
[255] If the control panel buttons don't work, try clicking anywhere in the left-hand band activity pane in JTDX to focus on it.

[256] Green for go, red for stop seems intuitive to me … but on your system, the choice is yours.

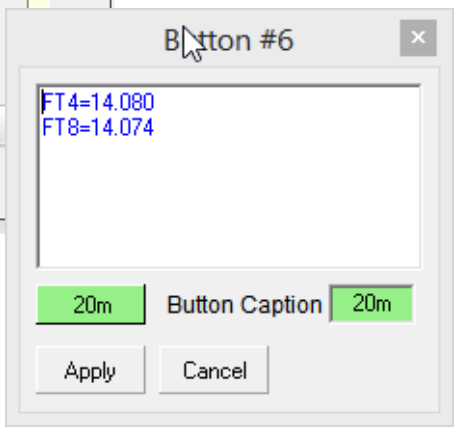
[257] Naturally – this is Logger32 after all. Don't like how something looks? OK, right-click and change it!

[258] Not the buttons, not the "LED", not the gray JTDX Control Panel. Right-click the little rectangle on the righthand side of the UDP BandMap, below the X.

Copyright © 2026 Robert C. Furzer K4CY



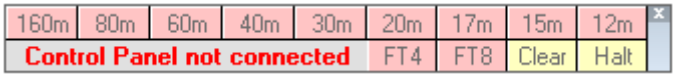
◀ *While the JTDX Control Panel is connected to JTDX,*



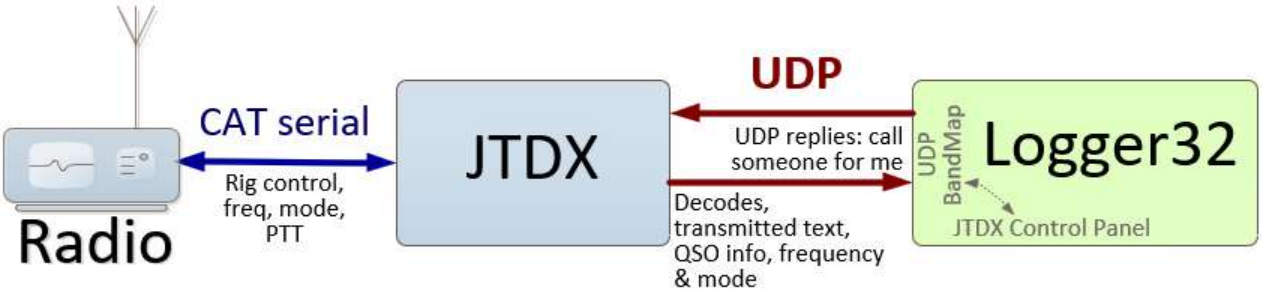
right-click any band button to redefine the preferred FT# VFO frequencies for that button, and/or to edit the Button Caption (its text label) ▶

Important note: license terms vary on 60m around the world. Before using 60m, check your license and comply with *your* permitted frequencies, modes, power limits *etc*.

If you close JTDX using **Stop ⇨ JTDX** on the UDP BandMap menu with the JTDX Control Panel still open, or if JTDX mysteriously shuts down, a bold red message explains why the JTDX Control Panel is no longer functional ▶



The block diagram shows how Logger32 can determine the radio's frequency, send it to a different band or swap between, say, FT8 and FT4, *without* a direct CAT connection to the radio[259] ▼



15.4 Opening & closing digimode software through Logger32

15.4.1 JTDX launch sequence

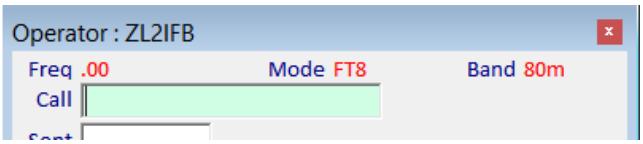
Clicking <**Ctrl+J**> with the focus on the log entry pane, or clicking the UDP BandMap's **Start ⇨ JTDX** triggers the following 15-step automated sequence:

1. Logger32 stores the current frequency and mode for the active radio.

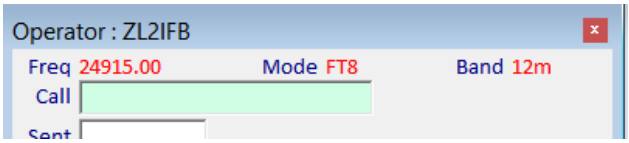
[259] Logger32's radio COM port closes when JTDX is opened using the UDP BandMap's **Start ⇨ JTDX**. *Either* Logger32 *or* JTDX can have direct CAT control of the radio at a time, not both.

Copyright © 2026 Robert C. Furzer K4CY

2. Logger32 changes the UDP BandMap menu <**Start**> option to the <**Close**> option.
3. Logger32 executes any macros defined in the 'before starting' section of the **Setup Shortcuts** configuration form, sending commands to the active radio via its CAT connection.
4. Logger32 closes the CAT connection and port, turning the **Radio** panel in the status bar to red text and temporarily showing the radio frequency ".00" in the log entry pane ▶



5. Logger32 launches the digimode program whose executable file is defined in the top section of the **Setup Shortcuts** configuration form, with high priority and 'on top' if so configured.
6. Logger32 also launches any additional programs defined on the form.
7. Logger32 waits patiently for the digimode software to launch and settle down – the number of seconds' delay being defined under **Start ⇨ Delay after starting JTDX**.
8. The digimode software opens *its* CAT connection to the radio, commanding it to the designated frequency for the present band and digimode, and the relevant radio mode for digital communications (*e.g.* USB, DATA or DIGI).
9. The digimode and additional software hook up, somehow.
10. The digimode software messages Logger32 with the radio's frequency and the digimode.
11. Logger32 updates the frequency, mode and band in the log entry pane ▶



12. The digimode software opens the designated audio device, then starts capturing and decoding audio from the radio.
13. After the delay period[260], Logger32 identifies the digimode software's main window from its caption (it's "Windows Handle" to geeks), then changes the caption to "JTDX & Logger32", indicating joint control. It also appends "on top" if that option is enabled.

> **Hinson tip**: if you have configured it to be on top but "on top" is not shown in the JTDX caption, Logger32 has probably given up trying to assert control before JTDX was ready … so lengthen the startup delay by a second or two and give it another go. Repeat as necessary.

14. Logger32 removes the Minimize/Restore icons from the top right corner of the digimode window and, if so configured, links that window with the Logger32 window status.
15. Logger32 sets itself up to log completed QSOs in WSJT-X automatically by identifying WSJT-X's "Log this QSO" pop-up window[261].

From there, Logger32 carries on as usual, receiving and acting appropriately on UDP messages from the digimode software *e.g.* adding callsigns and grids to the UDP BandMap, highlighting new ones and sounding audio alerts, updating the Tracking window maps, looking up the callsigns of stations you are working, picking cherries *etc*.

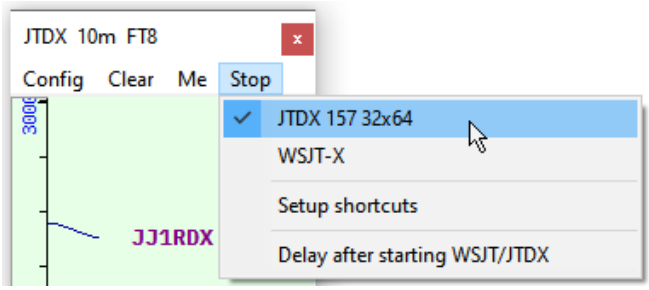
---

[260] By monitoring this sequence in action, you can tell whether you have set the launch delay appropriately. Too short a delay truncates the sequence, for example leaving the digimode software caption or log entry pane unchanged. Too long a delay simply wastes a little time … but it's better to wait a second or two longer than to terminate the sequence prematurely. 15 seconds works fine with *my* setup: *yours* may vary.

[261] Since JTDX can log completed QSOs automatically, this step is only necessary to overcome a limitation deliberately coded into WSJT-X – one of several reasons why JTDX is preferred.

Copyright © 2026 Robert C. Furzer K4CY

15.4.2 JTDX termination sequence



◀ Clicking <**Stop**> and then whichever digimode package is currently running (the one with the tick[262]), or (with Quick Switch enabled) clicking a DX spot for a station on another mode (such as SSB), triggers the following automated sequence of 7 steps:

1. Logger32 removes <**Stop**> from the UDP BandMap menu.
2. Logger32 commands the digimode software to close, plus the JTDX Control Panel if open.
3. Logger32 puts <**Start**> on the UDP BandMap menu.
4. The digimode software shuts down, closing its CAT and UDP connections, its windows and any open data files.
5. Logger32 detects that the digimode software has closed and re-opens its CAT connection to the radio, if it was open beforehand.
6. Logger32 executes any macros defined in the 'after normal closing' section of the Setup Shortcuts configuration form, sending commands to the active radio via its CAT connection.
7. Logger32 commands the active radio back to the frequency and mode it was on before JTDX was launched.

If, instead of using the <**Close**> menu, you simply close JTDX directly using its corner X or terminate the process in Task Manager, or if the digimode software crashes, Logger32 detects the unexpected change and pops-up a notification in the lower right corner of the screen ▶



Then Logger32 completes the termination sequence as best it can, re-opening its CAT connection to your radio *etc*.

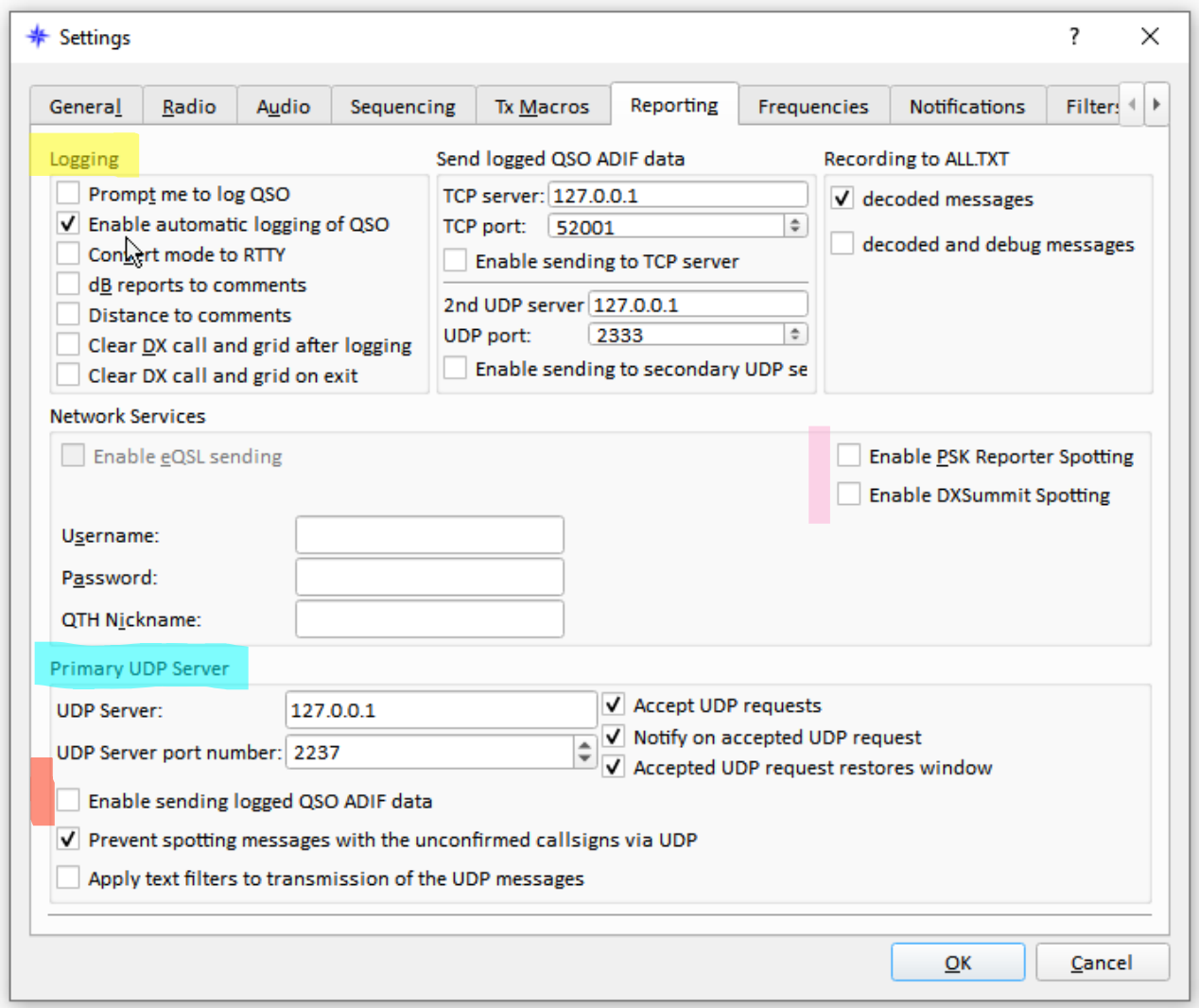
Hinson tip: if you choose *not* to use Quick Switch, a simpler form of automation is available just for the JT modes. Say you are using FT8 on 20m having opened JTDX from the UDP BandMap. Seeing an interesting FT8 spot on 15m, you click that … and Logger32 dutifully tells JTDX to QSY your radio to the FT8 watering-hole frequency on 15m (which may not be right, but it's a start).

[262] If you tick *another* one, without the tick, the ticked one stops and the other one starts up.

Copyright © 2026 Robert C. Furzer K4CY

15.5 JTDX configuration settings

Under the **F2/Settings** ⇨ **Reporting** tab, review the **Logging** and **Primary UDP Server** sections ▼



Only <**Enable PSK Reporter Spotting**> and <**Enable DXSummit Spotting**> if you actually *want* to send the callsigns of every station your system has decoded to these sites. Doing so is spammy.

Hinson tip: thanks to automation, all the DX spotting networks receive *stacks* of information already, particularly from more crowded parts of the world. They are inundated. If you are somewhere more remote, isolated and/or exotic, there is perhaps a *little* more interest in your propagation, and some value in the additional network traffic. Probably not though.

By the way, there's generally no need to tick <**Enable sending logged QSO ADIF data**> because Logger32 gets all the info it needs to log QSOs from the stream of decodes. Save a few CPU cycles and bytes.

Copyright © 2026 Robert C. Furzer K4CY

Under **Logging**, select <**Enable automatic logging of QSO**>[263]. When a QSO is completed and logged in JTDX, the QSO information is passed to Logger32 where it is recorded almost immediately in the open log. No clicks required! Focus on initiating your *next* QSO.

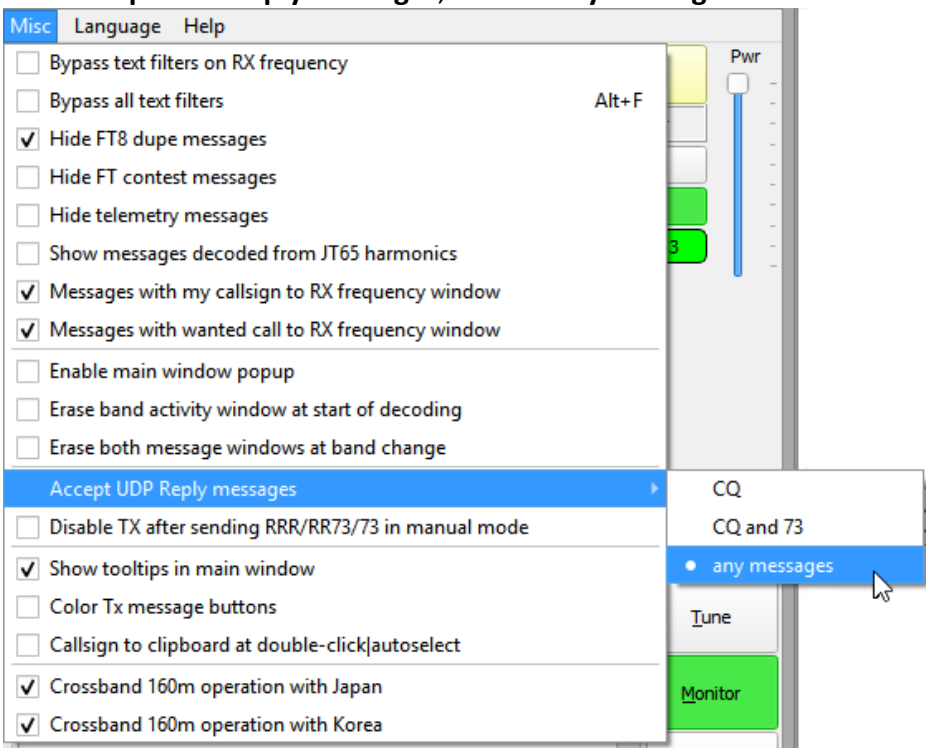
Under **Primary UDP Server**, the default values shown above work fine *i.e.* IP address 127.0.0.1 port 2237 with the first five options enabled. Feel free to experiment with other settings though.

<**Prevent spotting messages with the unconfirmed callsigns via UDP**> stops JTDX passing dubious (low assurance, potentially busted) callsigns to Logger32, preventing you and the cherry picker from wasting time and watts attempting to call them.

Do *not* enable the **TCP server** or **secondary UDP server** … unless you need them for some reason.

With JTDX, cherry-picking is a little complicated with further settings in the JTDX main screen menu[264] that are relevant to how it processes decodes and interacts with Logger32:

- Under **Misc ⇨ Accept UDP Reply messages**, select <**any messages**> ▼

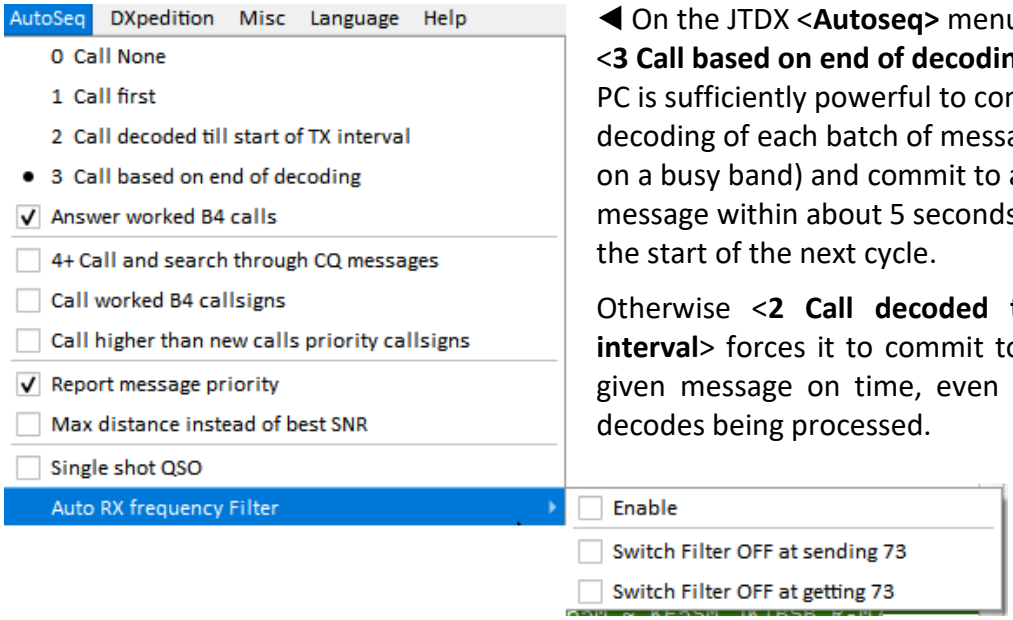


Otherwise, JTDX ignores some UDP reply messages from Logger32, leading to "JTDX did not switch to transmit" notes in the cherry-picker event viewer, missed opportunities to pick cherries just as they ripen, and hence a reduced yield in the cherry harvest.

[263] If you feel inclined or obliged to check the details before logging QSOs, select <**Prompt me to log QSO**> instead: just be sure to log all your completed QSOs, or there will be gaps in your log and maybe complaints from the people you worked.

[264] To display the JTDX main screen menu, click to tick <**Menu**> between the frequency and time panels.

Copyright © 2026 Robert C. Furzer K4CY



◀ On the JTDX <**Autoseq>** menu select <**3 Call based on end of decoding**> *if* your PC is sufficiently powerful to complete the decoding of each batch of messages (even on a busy band) and commit to a transmit message within about 5 seconds of the start of the next cycle.

Otherwise <**2 Call decoded till start of TX interval**> forces it to commit to and transmit a given message on time, even if there are still decodes being processed.

The advantage of <**AutoSeq3**> is that JTDX is more likely to react appropriately to messages sent to you and decoded quite late, sometimes even *after* it has started transmitting (JTDX changes the outbound message prior to the critical 5 second point, during the initial synchronization segment). With <**AutoSeq2**>, you will occasionally see late responses to your previous message being decoded and displayed *after* JTDX has already commenced transmitting a repeat of its previous message: it simply continues sending the repeated message, regardless.

Hinson tip: provided you are sufficiently on-the-ball to notice this happening and react promptly, you can *manually* click to select your next Tx message, overriding the autosequencer. If you are too slow on the change, however, the Tx message may be corrupted and undecodable.



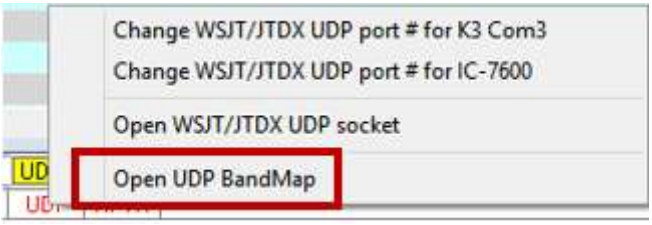
◀ Both the <**AutoTX**> and <**AutoSeq**> buttons on the main JTDX screen should be green (active): if they are pink, click them.

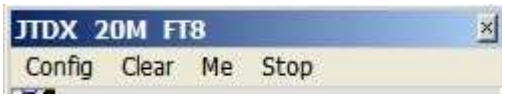
Hinson tip: I recommend using <**Tx/Rx Split**> all the time too. The entire audio segment is decoded so use the space. Simplex operating is unnecessary on FT#, and can make copy difficult with numerous simplex callers – just like an old-school CW or SSB pileup. Spread out a little! Right-click a quiet frequency on the waterfall and leave your TX there unless there is a good reason to move.

Copyright © 2026 Robert C. Furzer K4CY

15.6 *Using* the UDP BandMap

- Right-click the **UDP** panel on the status bar, then click <**Open UDP BandMap**> ▶
- Launch JTDX|WSJT-X, preferably using <**Start** > in the UDP BandMap menu.

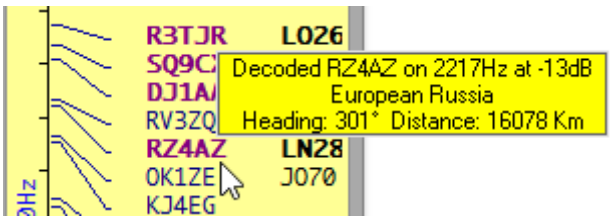




◀ JTDX|WSJT-X plus the current band and mode are shown in the UDP BandMap's caption.

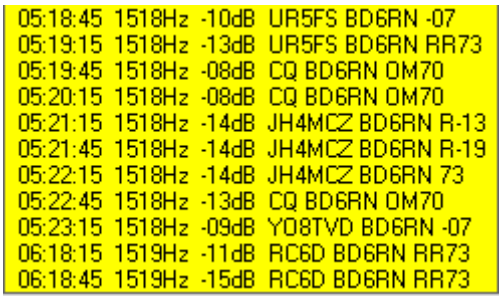
- Transmissions received and decoded by JTDX|WSJT-X are passed to Logger32 over UDP. Depending on its configuration:
- The decoded stations' callsigns are displayed on the UDP BandMap, in bold if they have CQ'd.
- Any 'new ones' are highlighted by Logger32 *i.e*. new DXCC entities and/or grid squares.
- Stations worked before have an X between the frequency line and the callsign, plus green blobs in the same area if they are LoTW or Club Log's **O**nline **Q**SL **R**equest **S**ervice users.

- Mouseover any callsign on the UDP BandMap for a popup tooltip showing additional information about the station, including the audio offset and strength of his most recently decoded message ▶



- Right-click any callsign for a popup listing up to 15 recent decodes from that station ▶

  Aside from the message content, this can be handy to figure out whether the station is listening on the even or odd periods (hence when would be best to call), plus his audio frequency and signal strength, varying naturally due mainly to QSB.

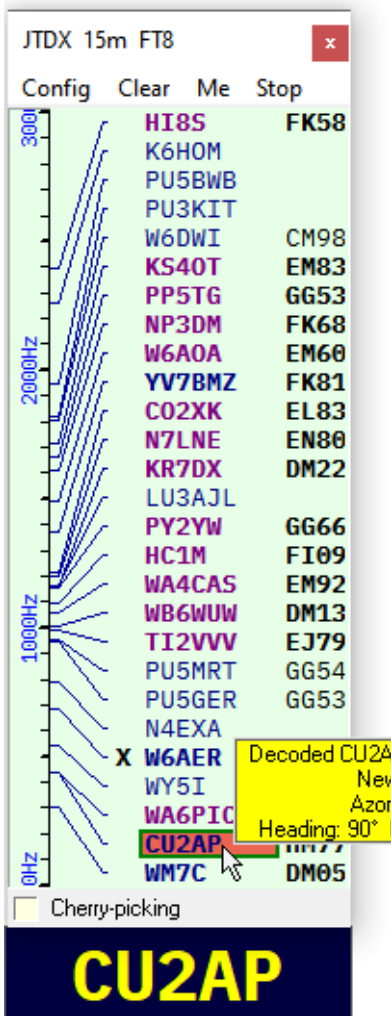


- Click any callsign you wish to call (see Manual calling below).
- Right-click the grid square if shown (if not, right-click in the space where it *would* have been) to spot that DX station on DX cluster.

Hinson tip: the UDP BandMap is one of Logger32's most valuable features, a real boon for digimode DXers, *far* better (in my opinion) than JTDX|WSJT-X alone and better even than add-ons such as JTAlert or GridTracker. Since Logger32 maintains my consolidated station log, it knows who I have worked (and hopefully had confirmed and verified) using *any* of my personal callsigns on *all* bands and modes. Thanks to regular updates from Club Log, it correctly identifies virtually *all* DXCC entities, even for obscure or ambiguous callsigns such as those strange ones often used by DXpeditions and special event stations. It knows which gridsquares I have or have not contacted, and alerts me if specific stations are decoded, using familiar highlighting colors and sounds. Unreservedly, I commend it to the house.

Copyright © 2026 Robert C. Furzer K4CY

15.6.1 Manual calling



Click the callsign of any station listed on the UDP BandMap that you want to call via JTDX|WSJT-X.

◀ The callsign is surrounded (after a short pause) by a colored box[265] and populates the Call field of the log entry pane, triggering the usual callsign lookup and logbook search for any previous QSOs with that station.

A 'UDP reply message'[266] is sent to JTDX|WSJT-X, telling it to call the DX station at the next opportunity (generally when he is CQing or has just finished a QSO).

If the DX responds to you, the box disappears. If not, Logger32 makes up to 5 calls before giving up.

If there are no decodes from the DX for 90 seconds after you have called him, or if you click the callsign again, the box disappears, the call is abandoned and the log entry pane is cleared.

◀ This dark blue box and yellow callsign was the floating callsign pane on my system.

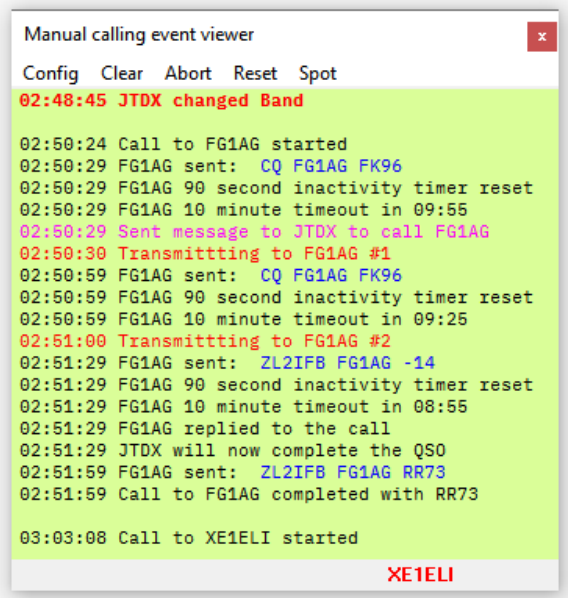
Hinson tip: having clicked a callsign on the UDP BandMap to start calling a specific station, for some reason on my system, Logger32 quite often highlights a *different* callsign and starts calling someone else. I find it helps to display the floating callsign window beside the UDP BandMap. With the selected callsign emblazoned big and bold in bright yellow against a dark blue background, I'm more likely to spot the discrepancy in time to click and so call the station *I* wanted to contact, not the one Logger32 unilaterally chose.

[265] Choose/change the box color using **Config ⇨ Appearance ⇨ Selected callsign border color**.

[266] Logger32 communicates with JTDX by returning a UDP message containing a decode from the selected station, that had previously been sent to Logger32 via UDP. That "reply message" is taken by JTDX to be an instruction to call the sender at the next appropriate opportunity. Logger32 doesn't have a free hand to tell JTDX precisely which message/s to send and when, unfortunately: the digimode software does *not* behave like a dumb modem, similar to MMTTY and MMVARI.

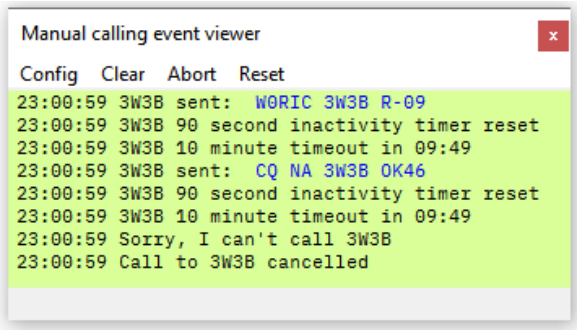
Copyright © 2026 Robert C. Furzer K4CY

Allowing time for the selected DX station to work other, luckier DXers, a timer counts down 10 minutes from the initial click, leaving time enough for us to have a fair go at contacting them.



To show the manual calling activities, click **Config ⇨ Show manual calling event viewer ▶**

The event viewer gives strong clues about what it is doing. Warnings and QSO logged messages are shown in **bold** to stand out from the rest.



◀ In this example, it dutifully refused to call 3W3B because I am not in North America, and Bruce was specifically calling CQ NA.

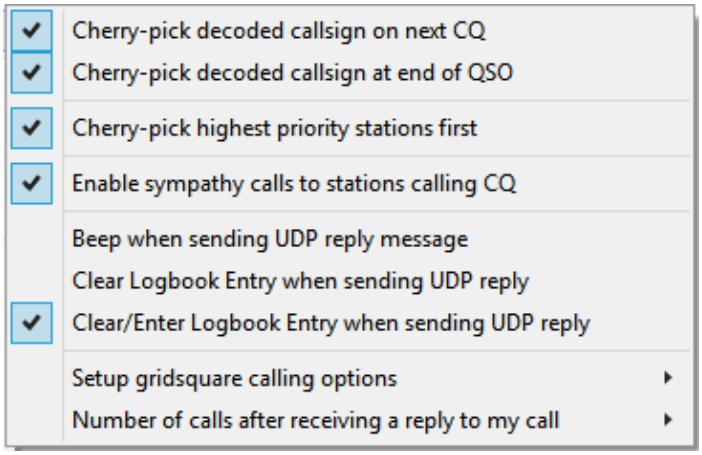
Hinson tip: manual calling takes a little more effort and concentration than automated cherry-picking, but it is often worth it when chasing specific rare or semi-rare DX stations. The cherry picker does have its priorities but it *appears* scatter-brained, seemingly flitting from station to station on a whim whereas with manual calling we can concentrate exclusively on working whichever DX callsign *we* have chosen. Manual calling is also more in line with the spirit of DXCC rule 6a: *"Each contact claimed for DXCC credit must include contemporaneous direct initiation by the operator on both sides of the contact. Initiation of a contact may be locally or by remote."* Aside from that, I find it more satisfying: I enjoy active DXing more than passively observing.

Copyright © 2026 Robert C. Furzer K4CY

15.6.2 Cherry-picking (automated calling)

Logger32 can automatically select a station it believes is worth working (a “cherry”), then instruct JTDX to “pick” (call and hopefully work) it.

Configure the cherry-picker through the UDP BandMap **Config ⇨ Cherry-picking options** menu and submenus ▶



Once configured, enable cherry-picking by clicking to tick <**Cherry-picking**> at the bottom of the UDP BandMap ▼



As well as CQ messages, Logger32 can be configured to send ‘UDP replies’ for 73 and RR73 messages, so JTDX will ‘tail-end’ (start calling the DX without waiting for him to CQ). However, in FT# mode, WSJT-X only accepts ‘UDP replies’ for CQ calls.

The cherry picker does not call stations that you have already worked in the same band-mode slots (either all-time or just this year, depending on the setting in the top left corner of the worked/confirmed table). Those cherries are already picked and in the bag.



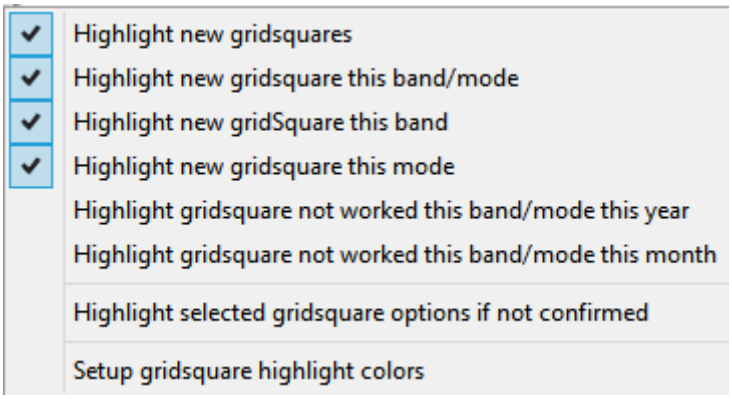
It recognizes CQ, CQ DX, and CQ [your continent]. Other directed CQ calls are ignored[267].

- With **Config ⇨ Cherry-picking options ⇨ Cherry-pick highest priority stations first**, Logger32 selects and attempt to pick the ripest cherries in each batch of decodes. It prefers highlighted stations (according to your configuration of highlighting for wanted callsigns and/or grids – see below), giving priority to new DXCCs over new grid squares, and to stronger stations if it has the choice. If this option is not selected, Logger32 simply picks the first cherry it sees.
- It remembers the callsigns of any stations that call you while you are in QSO with another. On completion of the current QSO (or when a failed QSO attempt is abandoned), Logger32 *may* respond to the callers … unless there are nice ripe cherries ready to pick instead.

[267] By design, Logger32’s cherry picker will not call DX stations with a slash in their callsigns if the current user’s callsign also has a slash. This limitation arises from a shortage of data bits to exchange both slashed callsigns in the same message. Try manual calling instead.

Copyright © 2026 Robert C. Furzer K4CY

If you get bored of chasing DXCC and IOTA, the UDP BandMap **Config ⇨ Cherry-picking options ⇨ Setup gridsquare calling options** can highlight various new grid squares for you ▶



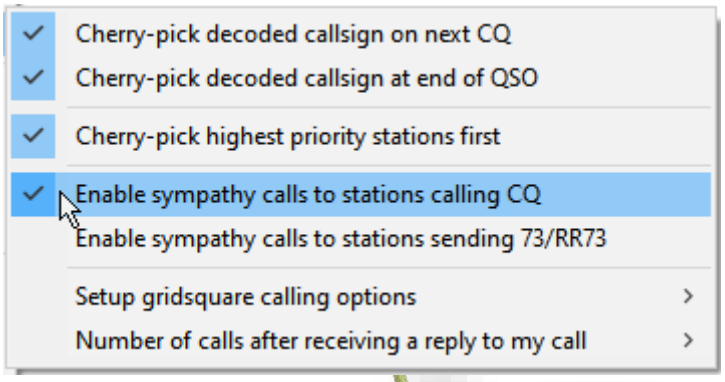
On FT#, CQing stations generally broadcast their grid squares as part of the CQ messages. JTDX decodes the messages and passes them to Logger32 via UDP. Logger32 displays the CQing stations' callsigns in bold on the UDP BandMap, plus their grid squares. It looks up the grids to find out if you have worked them (ever or on this band and/or mode), and colors the grid background to highlight any new ones.

If you click a callsign on the UDP BandMap (for manual calling), automated cherry-picking is disabled – and *vice versa*. Choose either robot, or neither, but not both.

When the cherry-picker picks a cherry:



- The DX callsign is clearly displayed (in cherry red!) in the status bar at the bottom of the UDP BandMap ▶
- The DX callsign is put into the Call field of the log entry pane, triggering the usual lookups that populate the worked/confirmed table, the Previous QSOs pane and details from the DX station's entry at QRZ, HamQTH, HamCall *etc*.
- If <**Cherry-pick decoded callsign on next CQ**> is enabled (ticked) under **Config ⇨ Cherry-picking options**, JTDX is sent a UDP reply message shortly after the next CQ call from the DX station has been decoded, triggering it to compose the standard messages and start calling him (using Tx1, or Tx2 if you have ticked <**SkipTx1**>).





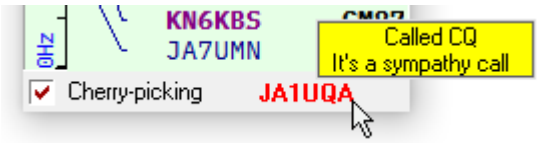
- If <**Cherry-pick decoded callsign at end of QSO**> is enabled, Logger32 can tell JTDX to 'tail-end' *i.e.* call the DX station after he sends an RRR, RR73 or 73 message to someone else, without waiting for him to CQ (however this option is unavailable when using WSJT-X on FT#[268]).
- If the DX station is not decoded for 90 seconds, Logger32 presumes he has faded away or expired and halts JTDX, stopping it from calling the DX. The DX callsign disappears from the UDP BandMap status bar. Logger32 is immediately keen to pick another cherry.

[268] Thanks to the way it handles UDP reply messages on FT#, WSJT-X only calls stations that have just called CQ. This is evidently another deliberate choice in the WSJT-X code, designed to hinder automation.

Copyright © 2026 Robert C. Furzer K4CY

- **Enable sympathy calls to stations calling CQ|sending 73/RR73** responds to CQ callers or tail ends others, even if they are not 'wanted' or 'needed'. The robot is hungry for digimode QSOs.

Mouse-over the cherry callsign to see what is going on ▶



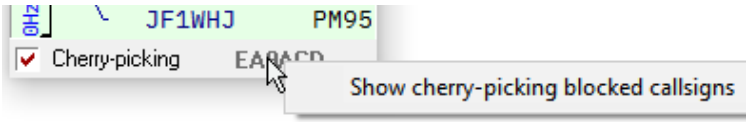


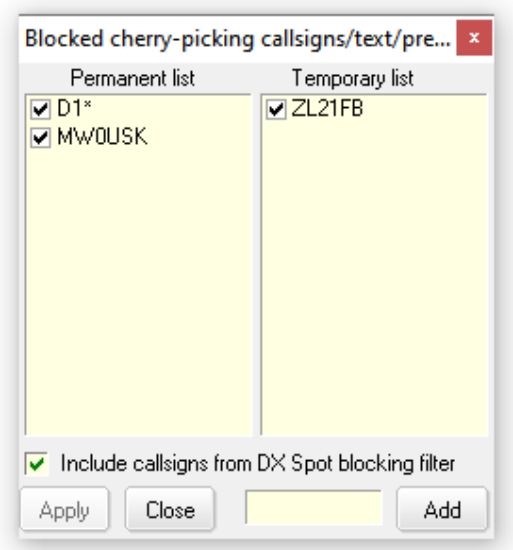
◀ To understand what the cunning robot doing in more detail, **Config ⇨ Show cherry-picking event viewer** gives a color-coded message-by-message account – an audit trail of its logic and the information flowing each way through the UDP connection between JTDX and Logger32. On the viewer menu, click **Config ⇨ Show all messages** for even more detail.

◀ The 3 numbers in the bottom right corner tell us (1) how many callsigns were decoded in the last batch, (2) the number of those that were new and hence were checked as to their wanted/needed status, and (3) the number of multi-stream decodes.

15.6.3 Cherry-picking callsign blocker

At the bottom of the UDP BandMap, right-click the area to the right of "Cherry-picking" then click <**Show cherry-picking blocked callsigns**> to show and configure the callsigns *etc*. that the cherry-picker will not call ▲





◀ The **Temporary list** shows any callsigns or strings from recent decodes that Logger32 has determined are not, in fact, genuine callsigns ▶ Corrupted messages sometimes sneak past the clever error trapping in FT8 and other digimodes, hence you may occasionally see decodes containing nonsensical gibberish "callsigns" such as 12AAC3B. Likewise, custom 73 messages containing strings vaguely resembling callsigns may be listed here. Logger32 adds them automatically to the temporary list to prevent the cherry picker trying to contact them. The temporary list is wiped when the UDP BandMap is closed.

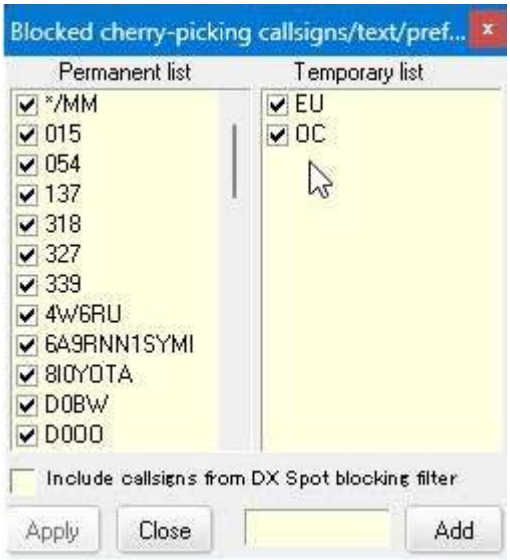


Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: occasionally I am called repeatedly by stations who either don't copy my replies, or have neglected to select Autoseq, so they keep on repeating the same message at me, parrot fashion. Most of them either stop sending or move on to the next message after a couple of cycles but a few persist *way* beyond that point … in which case I may add them to the temporary block list in order to blank them while I chase other stations that *can* actually copy me.

The **Permanent list** shows any callsigns or entities that are blocked indefinitely from being cherry-picked. These are saved to disk under the current Logger32 configuration and are reapplied every time the cherry-picker runs. To add items to the permanent list, right-click anything on the temporary list – in other words, first add stuff to the temporary list, then right-click it to make it permanent.

You can add a callsign to the UDP BandMap temporary block list by typing it into the yellow box at the bottom of the form then clicking <**Add**>. Wildcards work here, in the same way as when blocking regular DX spots. You can also block stations according to their continents or DXCC entity code numbers *e.g.* if for some reason you don't want to call or respond to any JA stations, use the code 339.



You can also click-and-drag unwelcome callsigns on the UDP BandMap in order to block them. If not already shown, the configuration form appears once you start the drag. Release the click to deposit the callsign in the temporary or permanent list ▶

To remove entries from either list, first click to un-tick them, then click <**Apply**> to action the removal.

Hinson tip: don't add your own callsign to either blocked list unless you *intend* to disable the cherry picker. It checks systematically through the received messages for people communicating with you. If you block those messages, you blindfold the poor robot.

<**Include callsigns from DX Spot blocking filter**> stops the cherry-picker calling any stations or entities that you have blocked on the DX Spots pane and the other BandMaps[269].

When you are happy with the setup, <**Close**> the form.

15.6.4 Cherry-picker goes fox-hunting



The FTn fox-n-hounds protocols were designed to cope with the me-me-me-me callers by splitting the pileup of baying hounds away from the DX station fox in both time-slots and frequency bands, and by condensing 'a QSO' to its bare essentials. The ability to complete one QSO *and* start the next QSO in the *same* compound TX message is an impressive tweak, and the multi-TX/multi-streaming option is another. Beneath all the magic technology, however, there are fundamental limits to the rate at which information can flow over a finite channel, ultimately limiting the number of hounds that will ever catch the fox. Study Claude Shannon and information theory, or simply listen to any unruly pileup for clues.

[269] They are neither shown nor managed on this form though.

Copyright © 2026 Robert C. Furzer K4CY

The cherry-picker can make a stab at calling and working DX stations using multi-transmit protocol variants such as MSHV, although it can get confused in a pileup … so it pays to keep an eye on the screen and be ready to take over (or call manually) if it appears to have lost the plot and ignored responses or given up too early after just 3 calls, perhaps even *during* a QSO.

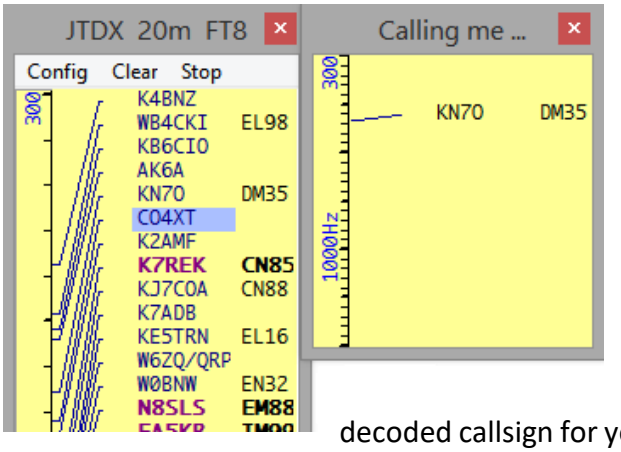
Hinson tip: DXing is a participative sport. Don't expect the PC to do *all* the hard work for you, especially if you value the fun and personal achievement of bagging those DX foxes yourself.

15.6.5 Other cherry-picker settings

Right-click <**Cherry-picking**> to show a further option <**Always open UDP BandMap with cherry-picking turned off**> ▶



This avoids you accidentally unleashing the cherry-picker simply by opening the UDP BandMap: you need to click <**Cherry picking**> to set the picker picking.



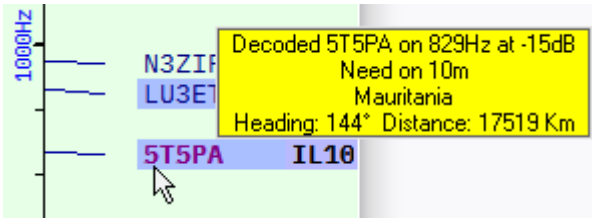
◀ Click <**Me**> in the UDP BandMap menu to open another BandMap showing *just* the stations calling you.

When you call CQ and receive replies, your callers' callsigns are displayed in the *Calling me …* BandMap.

JTDX will *automatically* respond to the first decoded callsign for you. When you finish the QSO, click another callsign on the *Calling me …* BandMap and JTDX will attempt to contact that caller next, if he/she is waiting patiently.

With **WSJT-X**, when you call CQ and receive replies, the callsigns are also displayed in the *Calling me …* BandMap but you must click a callsign to prod WSJT-X into attempting to contact that station.

Mouseover any callsign on the UDP or *Calling me…* BandMaps for additional information ▶



15.6.6 Logging QSOs

Thanks to the UDP messaging, when a QSO is completed and logged in JTDX, it is also logged in Logger32 a few seconds later. There is a *deliberate* delay before Logger32 logs FT8 and FT4 QSOs received via UDP – 5 seconds for FT8 and 3 seconds for FT4 – in order to defer Logger32's burst of CPU and disk activity until JTDX has completed its time-critical CPU-intensive message decoding.

However, there are minor differences between JTDX and WSJT-X:

- JTDX can log QSOs directly without user intervention as our 73 or RR73 message is sent, hence they are logged automatically in Logger32 just a moment later.

Copyright © 2026 Robert C. Furzer K4CY

- In WSJT-X, the log QSO window appears briefly when QSOs are completed, causing a slightly longer delay as Logger32 tells it to log the QSO, then logs the QSO itself.

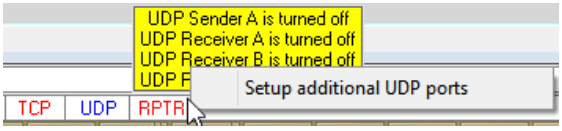
Your (VFO + audio offset) transmit and receive frequencies are written to the ADIF "FREQ" and "FREQ_RX" fields in the logbook[270]. *If* the DX station's grid square is known, Logger32 also calculates the short path distance from your home QTH for the ADIF "DISTANCE" field[271] ▼

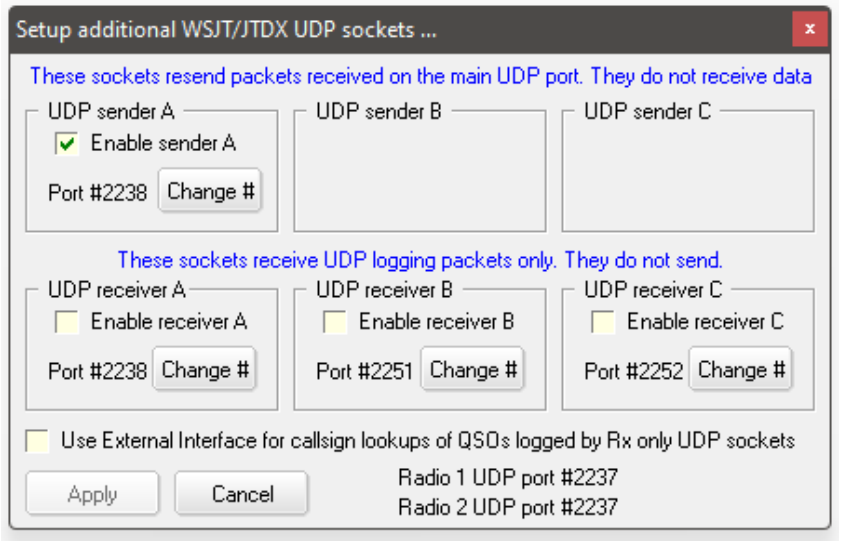


Hinson tip: details of the QSO information sent from JTDX|WSJT-X to Logger32 via **UDP** are recorded in *C:\Logger32\UDP Logging requests.txt*. If your system uses **TCP**, check out the file *C:\Logger32\TCPLog.txt* instead. Look through either file using a text editor for clues about what is really going on, especially if something is going wrong. Both files are deleted when Logger32 next starts so check or copy them before you next start up Logger32.

15.7 RPTR (JTDX|WSJT-X UDP message repeater)

If programs monitoring UDP traffic won't work because Logger32 is hogging port 2237, right-click the <**RPTR**> panel on the right of the status bar then click <**Setup additional UDP ports**> ▶





◀ Tick <**Enable sender A**>, set the port number to 2238 if not already set, and click <**Apply**>.

Logger32 now re-sends any UDP messages it receives through port 2237 through a 'repeater' UDP port 2238 for use by third-party apps.

[270] FREQ_RX is *only* logged if you launched the digimode software from the UDP BandMap. If you start WSJT-X, JTDX or whatever from anywhere *other* than the UDP BandMap, FREQ_RX is not logged even if the ADIF field (typically containing an audio offset) is sent via a UDP or TCP connection to Logger32. In order to log FREQ_RX (your receive VFO frequency), Logger32 uses the TX VFO frequency from the UDP message when JTDX is started and adds the RX audio offset. However, if Logger32 does not start JTDX, the initial TX VFO frequency is unknown and all bets are off.

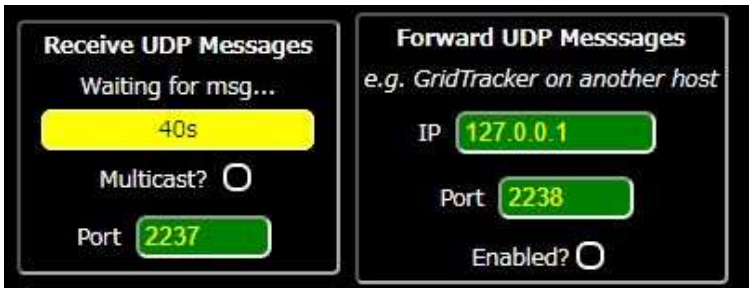
[271] ADIF specifies and hence Logger32 always calculates and records distances in kilometres, but just for you it can *display* miles, even nautical miles if you prefer. Not furlongs though, or chains. The conversion is trivial and happens so quick you won't even notice. Also, you may notice that I have modified my logbook column headers from the defaults so the ADIF field names are no longer all SHOUTY CAPITALS.

Copyright © 2026 Robert C. Furzer K4CY

15.7.1 Using GridTracker

GridTracker is a companion app for JTDX (and Logger32!) to help you identify and work 'new ones'.

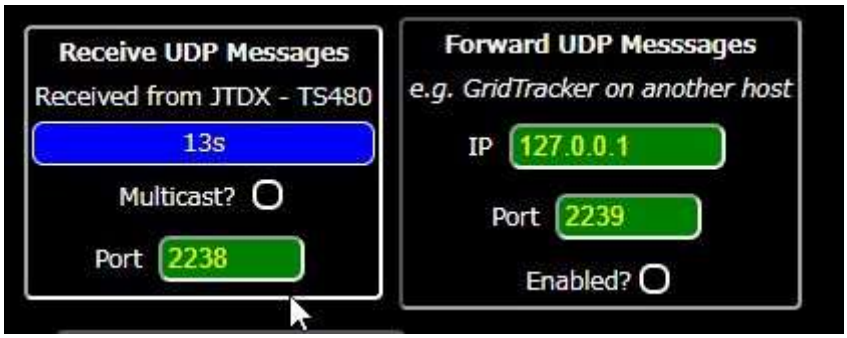
1. Download and install GridTracker. It is quite complicated so study the GridTracker documentation.
2. Run GridTracker and click the <**Setup**> icon. The default port to receive UDP messages is 2237 ▶
3. Change the Receive UDP Message port to 2238 and change the Forward UDP Message port to 2239. Alternatively, if for some reason you want to retain GridTracker's default UDP port 2237:
   - In Logger32: right-click the **RPTR** panel at the right of the status bar, then click <**Setup WSJT/JTDX UDP repeater**>. Click to <**Enable SEND A**>, set the port number to 2237, and click <**Apply**>.



Right-click the **UDP** panel in the status bar ▶ click <**Change WSJT/JTDX port #**>, type 2238 and click <**OK**>.



In WSJT-X/JTDX, change the UDP port to 2238 as well under **F2/Settings** ⇨ **Reporting** tab ▶

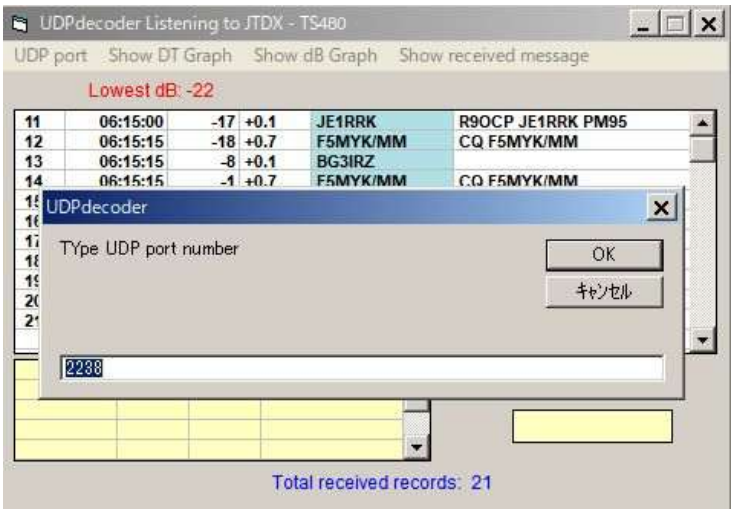


15.7.2 UDPdecoder

UDPdecoder is an app by JA1NLX that receives UDP messages from JTDX|WSJT-X via the RPTR function, displaying: the messages in a table; a dT distribution graph; a Signal-to-Noise ratio graph for specific callsigns; and the lowest Signal-to-Noise level received.

To use UDPdecoder:

1. Run Logger32.
2. Open the UDP BandMap.
3. Start JTDX|WSJT-X.
4. Open RPTR.
5. Run UDPdecoder.
6. Click <**UDP port**> to set the UDP port number ▶
   The default is UDP port 2238.



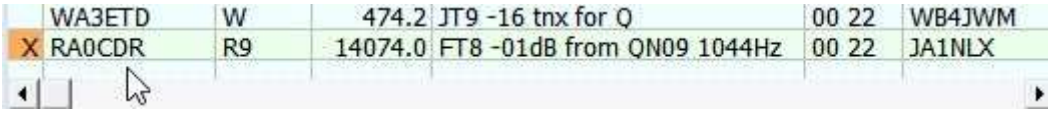
Copyright © 2026 Robert C. Furzer K4CY

15.8 Sending DX spots

Assuming you do not automatically spot *all* your FT4/FT8 QSOs[272], <**Spot**> appears on the Cherry-picking event viewer menu when a QSO is automatically logged ▶



Click <**Spot**> to send a pre-formatted DX spot to DX cluster ▼



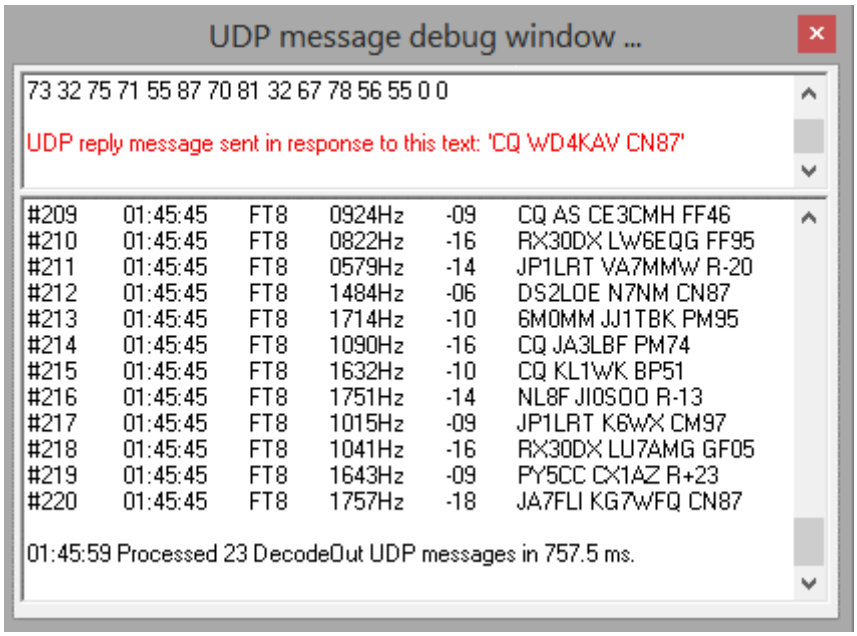
If you choose not to send the spot immediately, the spot text remains available until overwritten when the *next* QSO is logged.

15.9 UDP message debug window

The UDP message debug window can be very useful to diagnose communications problems between Logger32 and JTDX|WSJT-X.

Open it with **Config ⇨ Show UDP message debug window** ▶

UDP reply messages (sent by Logger32 to JTDX|WSJT-X) are shown in red.



The file *C:\Logger32\UDP Logging requests.txt* records every QSO logging request received from JTDX|WSJT-X via UDP during the current Logger32 session, for debugging purposes ▼



[272] Excessively prolific "ego-spotting" is generally frowned upon, leading to cluster comments along the lines of "Don't spot ur logbook", and may result in you being blocked from certain cluster nodes. Myself, I only ever spot DX. Oh and I get to define what "DX" means. Don't like that? OK, filter out my spots, no worries.

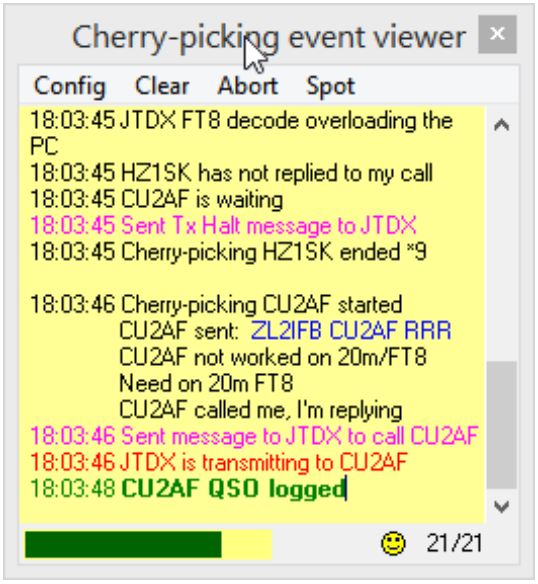
Copyright © 2026 Robert C. Furzer K4CY

However, the file is erased every time Logger32 starts, so if you need to check it, you might like to grab a copy *after* you have logged at least one digimode QSO but *before* you next start Logger32.

15.10 Event viewers

The event viewers show us what is going on 'behind the screen', displaying the sequence of messages exchanged with JTDX|WSJT-X along with clues about 'what the robot was thinking' – whether it was cherry-picking automatically or just calling someone we have selected.

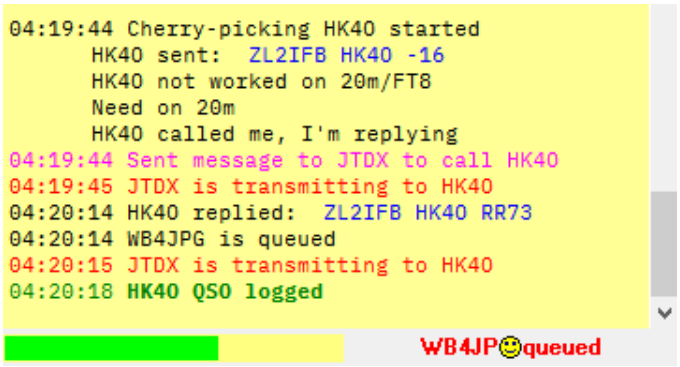
The sequence shows the reasoning, decisions and commands made by Logger32's cherry picker, plus JTDX|WSJT-X's reactions/responses.



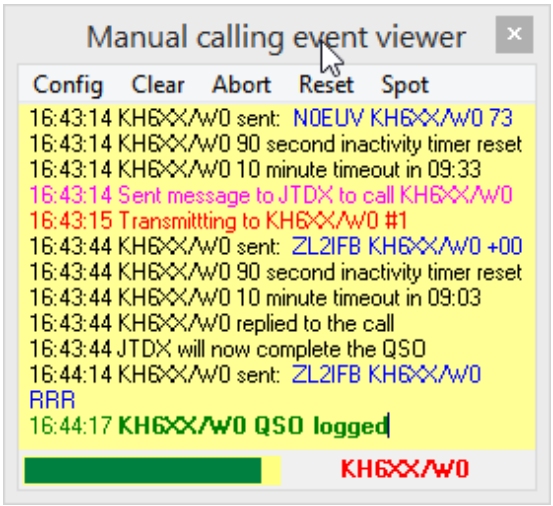
◀ The progress bar at bottom left marks the passage of time through FT# transmit/receive cycles.

The smiley face at the bottom right shows the robot is happy with no pain in the diodes down his left side. The numbers in the lower right corner show how many messages were decoded in the last cycle, and how many of those were from new callsigns – meaning that they had to be checked for their wanted/needed status, consuming a little time and several CPU cycles.

Other text may appear here too *e.g.* ▼



Messages in the main body of the viewer are color coded ▼



- Black text for the timestamp and sender's callsign, plus information about the sequencing, reasoning, decisions, actions and responses.
- Blue for FT8|FT4 messages received and decoded by JTDX, then passed to Logger32 via UDP.
- Pink for UDP reply messages sent to JTDX by Logger32, triggering JTDX to call the DX.
- Red confirming that JTDX has started calling him.
- Bold green for UDP 'log QSO' messages – QSO complete, it's in the bag!

Before you ask: *yes*, the colors *can* be changed. Read ahead for instructions or figure out the viewer's <**Config**> menu for yourself.

Copyright © 2026 Robert C. Furzer K4CY

Shortly after a sequence ends (*e.g*. the robot quits calling someone or a QSO is completed), the cherry picker event viewer usually shows a terminal "star-code" indicating *why* the sequence ended, in the form of an asterisk (star) plus a number. This is what the star-codes mean:

*1 The QSO has been logged, despite the DX station moving directly on to their next QSO or CQ call without sending us a 73 or RR73 – or at least, if they sent it, we didn't receive it.

*2 The DX station we were calling has responded to someone else, so the robot politely stops JTDX|WSJT-X from calling them and sits patiently on the side, observing and watching for other ripe cherries to pick.

*3 Code reserved for future use. It's a spare. One can never have too many star-codes.

*4 For some reason, having been commanded to call someone, JTDX|WSJT-X did not raise the relevant flag denoting that it is transmitting. The impatient robot waited just 2 seconds for the flag-raising ceremony, then gave up. It may try again on the next cycle.

*5 Lost contact: DX station not decoded for the past four cycles. The eco-friendly robot halts JTDX|WSJT-X to conserve a few watt-seconds for more esoteric processing.

*6 The robot called someone thrice without getting a response. It will now take a break, not calling the same station again for about 3 minutes to avoid nagging them, hoping for better luck next time. Instead it may call someone else meanwhile, keeping busy. Easily bored, she *hates* being idle.

*7 For some reason, <**Enable Tx**> was un-set in JTDX|WSJT-X, aborting our transmissions, so the call or QSO has been abandoned by the robot. Exactly why this happened, we don't know. Did we click the button? Did JTDX pull the plug because of one of its internal counters or timers expired? Did a spurious UDP message arrive from another dimension?

*8 Despite us plaintively calling someone, he rudely ignored us and started calling CQ DX. Mortally offended, the robot goes off in a huff, looking for a CAT to kick.

*9 The DX station we've been calling has not replied, and meanwhile someone else has called us so the affable robot will now attempt to work them instead. She's nice like that.

*10 Secret code, spooks alerted. Leave the shack quietly with your hands raised.

*11 The QSO was completed and logged. Happy days!

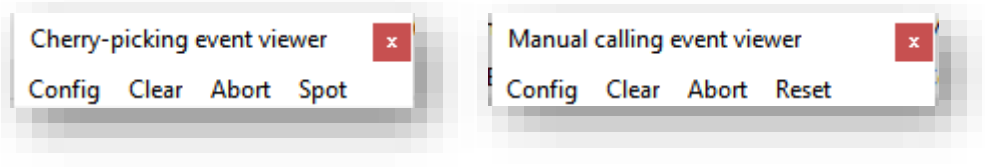
*12 JTDX/WSJT-X stopped calling someone. Perhaps it got distracted or lost interest?

*13 The robot didn't receive an anticipated status update from JTDX|WSJT-X saying that it had switched to receive after sending a message. Has JTDX|WSJT-X adjusted its timing to work someone with a lousy clock? Anyway, to avert a meltdown, the robot sends JTDX|WSJT-X one or two Tx Halt messages just in case the radio is stuck in transmit.

*14 The robot didn't receive an anticipated status update from JTDX|WSJT-X saying that it had completed decoding a batch of messages. Maybe it is looping, looping, or has completely lost the plot and fallen off its perch.

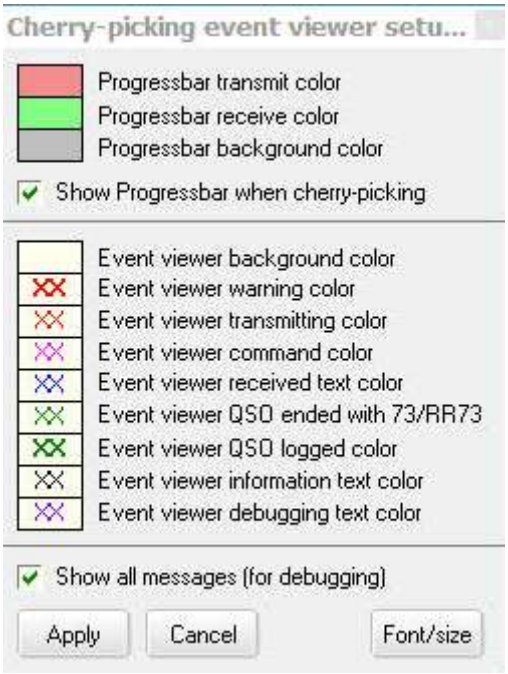
Hinson tip: it's worth studying the event viewers in action, for a while at least, to learn how the system works. There's a lot going on, especially when FT8 messages are streaming fast on a busy band. The event viewers show the robot concentrating on one callsign at a time (albeit sometimes queueing others), seizing appropriate opportunities to call wanted stations or respond to callers, pursue QSOs, and either log them when complete or reluctantly give up trying if they seem to be going nowhere … much like a proactive digimode DXer in fact. Watch and learn!

Copyright © 2026 Robert C. Furzer K4CY



▲ Both event viewer menus offer these functions:

- **Config**: the text colors and fonts are configurable ▶



> **Hinson tip:** different background colors can help distinguish the two event viewers – although, to be honest, I'm not sure why there are two separate panes for such similar functions, rather than a combined event viewer showing events for whichever robot is currently running?

  Also, on the cherry-picking event viewer config menu, <**Show all messages (for debugging)**> enables verbose mode with more details about what's going on: sit back and watch in awe as Logger32's robot plays JTDX|WSJT-X like an accomplished organist plays Bach at the cathedral.

- **Clear**: empties/wipes the event viewer, the same as closing and re-opening it, only without closing and re-opening it.
- **Abort**: tells JTDX|WSJT-X to Halt Tx. Stop sending! ***Now!*** It is almost as if you clicked the red <**Enable Tx**> panic button in JTDX|WSJT-X, draining the color out of it … except that it requires Logger32 to communicate effectively with JTDX|WSJT-X.
- **Spot**: sends a formatted DX cluster spot for the cherry currently being picked.
- **Reset**: zeroes the timers and counters that constrain the number of manual calls made to someone who doesn't respond promptly, enabling us to continue pestering the DX[273].

If either or both event viewers were open when the UDP BandMap was last closed, they re-open automatically when the UDP BandMap is next started. Also the UDP BandMap itself opens automatically if it was open when Logger32 was last shut down.

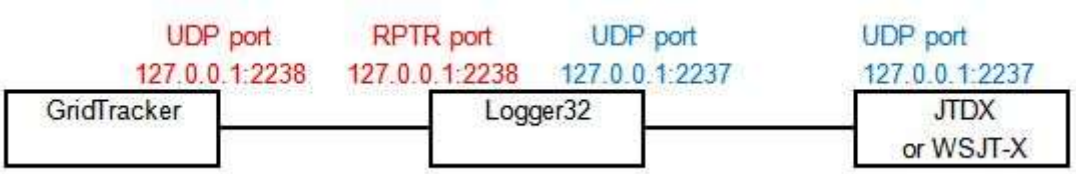
> **Hinson tip**: note that the cherry-picker robot also starts up automatically if it was running when the UDP BandMap was last closed. If there are any ripe cherries in the first batch of decodes, your transceiver will quite likely start transmitting a call to one of them straight away – even if your VFO is still drifting, your amplifiers aren't warmed up, your antenna switch may not be powered up, your ATU is un-tuned and you are maybe just a bit hungover from the previous night's digimode DXing, bleary-eyed and barely awake enough to double-click the Logger32 icon. If your station isn't fully automated and idiot-proof, be sure to **disable the cherry-picker** before chaos and catastrophe ensues, preferably remembering to do so *before* you shut down for the day.

[273] The cherry-picker's timers and counters are hard-coded and immutable, for fear of starting the FT# robot wars leading to the cherry-picker apocalypse.

Copyright © 2026 Robert C. Furzer K4CY

15.11 UDP port settings for GridTracker + Logger32 + JTDX

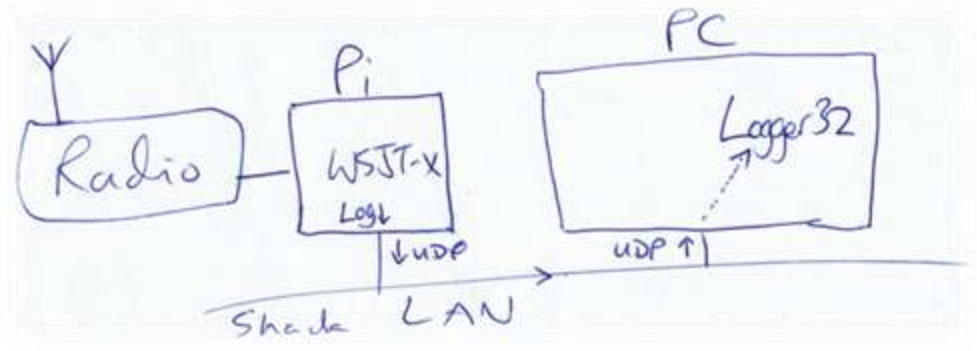
These are the recommended settings to link Logger32 with both JTDX and GridTracker, using Logger32's RPTR port function to copy UDP messages through ▼



The IP address and port numbers can be changed if needed, however the port numbers, at least, must correspond at each end of the connections.

15.12 UDP logging from a Raspberry Pi

Bill WC3B has figured out a way to connect a Raspberry Pi running WSJT-X to Logger32 on his shack PC, for logging purposes ▼




```
Command Prompt - powershell
Microsoft Windows [Version 10.0.19042.928]
(c) Microsoft Corporation. All rights reserved.

C:\Users\wbarnes>powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\wbarnes> $peerIP = "127.0.0.1"
>> $port = 2237
>> $endpoint = New-Object System.Net.IPEndPoint ([IPAddress]::Any, $port)
>>
>> $socket = New-Object System.Net.Sockets.UdpClient $port
>> $client = new-object net.sockets.udpclient(0)
>>
>> Try {
>>     while($true) {
>>
>>         $content = $socket.Receive([ref]$endpoint)
>>
>>         [Text.Encoding]::ASCII.GetString($content)
>>
>>         [void] $client.send($content, $content.length, $peerIP, $port)
>>
>>     }
>> } Catch {
>>     "$($Error[0])"
>> }
>>
>> $socket.Close()
>> $client.close()
>>
```

WSJT-X on the Pi sends its logged digimode QSO information to the shack LAN as ADI-formatted UDP messages.

◀ As shown by the dotted arrow on the highly sophisticated block diagram above, Bill runs a neat little PowerShell program on the PC to grab the UDP messages from the LAN and present them to Logger32's UDP socket.

Logger32 receives digimode QSO details via UDP messages and updates the shack log *as if WSJT-X was running on the same PC*.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: the same hack should work (I guess!) for WSJT-X or JTDX running on any platform on the same LAN – for example you might dedicate a spare computer and radio solely for FT#, leaving the main shack PC to run Logger32, propagation predictions, email, Web browsing and so forth.

This setup works for Bill. You may need to experiment a little to get it running sweetly in your shack. Thanks for the inspiration, Bill!

15.13 UDP BandMap FAQs

Q. I'm not interested in automated QSOs, so what's the advantage of linking Logger32 with JTDX?

A. If you prefer *not* to use Logger32's cherry-picker robot or its FT# manual calling features (which is absolutely fine – your choice), you can simply carry on CQing and clicking decodes in JTTDX to make FT# QSOs as normal … but if so configured, Logger32 *can*:

- Put the callsign of the station you are working in the log entry pane.
- Calculate his (or her!) location and turn your beam towards her (or him!).
- Display your previous QSOs with him, optionally carrying forward selected details such as her name and QSL information to be logged with the QSO this time.
- Check whether she would be a 'new one' (all-time, this year, this band, this mode *etc*.).
- Look him up in an online database such as HamQTH or QRZ, transferring further details to the log entry pane if you wish.
- Log the QSO in your main station log when completed, or clear the log entry pane if JTDX gives up and abandons an incomplete QSO with them.

These are the settings: on the UDP BandMap, under **Config ⇨ Cherry-Picking Options** enable <**Clear/enter logbook entry when sending UDP reply messages**>, telling Logger32 to put the callsign in the log entry pane. Also, click **Config ⇨ Logging options** and select <**Log additional info from log entry pane**>, telling Logger32 to add any additional information (such as the person's name from any previously-logged QSOs) when the QSO is logged.

Q. If I set the cherry-picker running, can I go to work or hit the sack?

A. No – that's a really bad idea for at least six good reasons.

For starters, it *may* be illegal to operate an unattended station under the terms of your license and/or the law. The authorities generally expect to be able to contact the station operator and close things down at short notice in case of issues. IANAL so read your license and ask a competent lawyer for advice.

Secondly, think about what will happen if (when!) you experience 'technical issues' such as rig, antenna, PC or sound card problems (*e.g.* random level changes leading to you transmitting nasty, overmodulated signals). Amateur equipment is generally designed, built and maintained for 'amateur service', as opposed to operating continuously 24x7: the stress may lead to overheating and premature failure, releasing the smoke, especially if you are not there to notice and react to things in time. In risk terms, the probability of such incidents is low but the potential impact is high, perhaps even deadly.

Copyright © 2026 Robert C. Furzer K4CY

Thirdly, consider the effects on your fellow radio hams, plus other radio services. Aside from you potentially transmitting around the clock and perhaps blocking locals from using the same band due to desensitizing their receivers or QRMing them, the system does not check that your transmit frequency remains clear – and things change constantly as propagation shifts and other users of the band move in.

Fourthly, bear in mind that some amateur radio awards forbid automated QSOs. DXCC, for instance, has the rule: "*Each contact claimed for DXCC credit must include contemporaneous direct initiation by the operator on both sides of the contact.*" You may well argue about the phrasing but despite the ambiguous wording, the intent is clear. Notice the final clause: if you are rare DX, stations that have contacted you may feel cheated out of their DXCC credits if they discover that *your* station was fully automated and unattended. They didn't contact *you*, they contacted your QSO robot, and those QSOs don't count for DXCC.

Fifthly, there is an ethical dimension to this. Joe Taylor K1JT and the team that invented the FT# digimode protocols object to full automation. It is their intellectual property, so they have the right to set terms and conditions on its exploitation.

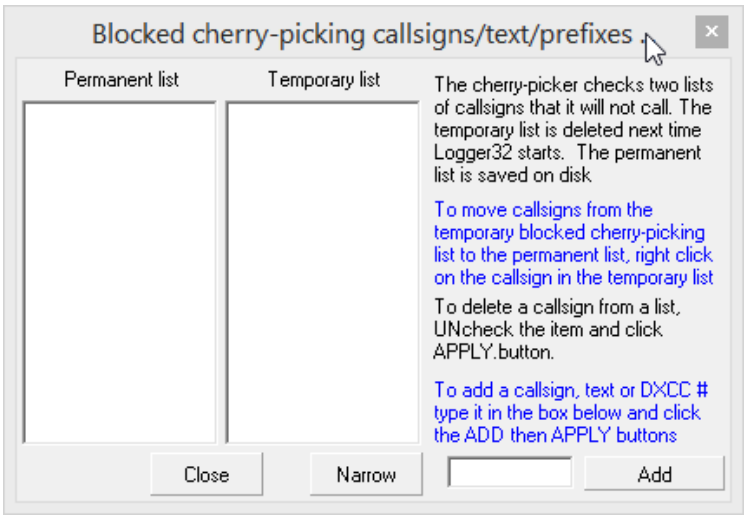
Finally, as if that's not enough already, there's the not insignificant issue of achievement and fun. Watching the system rack up FT# QSOs robotically may be interesting at first but, trust me, it soon gets very boring and pointless. Wandering off and coming back to see who we've logged takes things down another level. It's like letting a friend borrow your station for a while: you're happy they enjoyed themselves, and glad the technology held together, but clearly *they* made the QSOs, not *you*. Congratulations, you've taken yourself out of the loop. You are now redundant, a near useless appendage, relegated to the role of station-minder rather than operator. More radio sham than radio ham.

In short, be careful what you wish for.

Q. Why does the cherry picker blocker menu appear sometimes when I am selecting cherries to be picked?

A. You probably moved the mouse cursor a little as you clicked a cherry. The cherry-blocker function lets you stop both the manual and automated cherry-picker from attempting to contact certain stations - for example, known pirates.

Dragging the mouse cursor on the UDP BandMap opens the block form ▶



Hinson tip: no matter how anxious you are to start calling a DX station, *try* not to click while the mouse cursor is still moving. Hesitate briefly over a spot before clicking the mouse button. Pause a moment. Give it a rest. Let the dust settle.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why does the cherry picker pack up when I change bands? Have I broken it?

A. No, it's not broken. This behavior is intentional. See the middle option on the cherry picking options submenu.

Q. Why aren't the callsigns being called by the cherry-picker looked up for me?

A. On the UDP BandMap menu, open **Config ⇨ Cherry-picking options** and click to select <**Clear/Enter Logbook Entry when sending UDP reply**>. Now, as JTDX is calling a cherry for you, the callsign is shown on the log entry pane and floating callsign, looked up on the callsign database (*e.g.* HamQTH) and any previous QSOs with him/her are shown.

Q. Can I look up spotted DX stations while JTDX is busy working?

A. *Provided* you are using JTDX with Quick Switch disabled, you can click interesting DX spots to look up any previously-logged QSOs with those stations and check your worked/confirmed table, *without* changing mode and QSYing the radio.

To clear out the DX spot information and revert to whatever was in the log entry pane before you clicked the spot, simply click to focus on the log entry pane. You can also clear the log entry pane in the normal way by pressing <**F11**>, <**Ctrl**+**C**> or <**Alt**+**W**>, or clicking the left side of the pane in the area of the field labels.

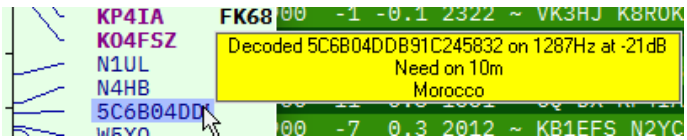
Q. Can I do anything to cut the amount of junk on the UDP BandMap?

A. With <**Show only callsigns calling CQ**> and <**Show only highlighted callsigns**> options selected, humdrum everyday stations and QSOs do not clutter up the UDP BandMap, leaving screen space for the more interesting ones. This can be a big help on busy bands at peak times … talking of which, QSYing to a quieter band is another obvious way to cut the junk. Bands near the MUF have fewer audible stations but (often) a higher proportion of DX.

Callsigns shown on the UDP BandMap are *not* DX spots from DX cluster, and are *not* filtered in the same way as DX spots on the regular BandMaps. The <**Block continent**> and <**Block spotter**> options are irrelevant and have no effect on what appears on the UDP BandMap.

Q. What's with those *weird* callsigns?

A. Logger32 merely receives and displays UDP messages sent by the digimode software. The digimode software is responsible for their content. Weird callsigns such as this ▼ are most likely the result of weak digital signals being decoded erroneously, the fancy data integrity/error-prevention controls in the digital protocols having clearly failed on this occasion. Alternatively, it *may* conceivably be the result of corruption in the UDP communications between the programs. Logger32, in turn, does its level best to determine which DXCC entity is indicated by all callsigns, including the weird ones, and highlights 'new ones' as applicable. In short, don't blame Logger32, or even the digimode software: the root cause is most likely a known integrity limitation of the digital protocol itself, coupled with weak signals and a noisy radio channel – yes, a *known* flaw, the result of a deliberate design compromise to make digital QSOs quicker and more straightforward, one that *almost always* works out just fine. Oh, and *some* weird callsigns are legitimate.



Copyright © 2026 Robert C. Furzer K4CY

Q. How do I log QO-100 satellite QSOs correctly?

A. Piggybacked on the Es'hail-2 commercial TV satellite, Qatar OSCAR 100 ("QO-100") geosynchronous amateur satellite is an orbiting cross-band multimode repeater (transponder) using a 13cm uplink and 3cm downlink. Its footprint covers most of the hemisphere centered on the Congo. As well as QRP QSOs using digital modes such as FT8, QO-100 can handle CW, SSB and even amateur video and contests.

Follow the instructions below to have Logger32 log cross-band QO-100 QSOs with the correct uplink (TX) *and* downlink (RX) bands, tagging them with propagation mode (ADIF field PROP) "SAT" and satellite name (ADIF field SAT_NAME) "QO-100" …

1. First, add the 3cm and 13cm bands to your bandplan by following these instructions ▼

Edit Bands & Modes

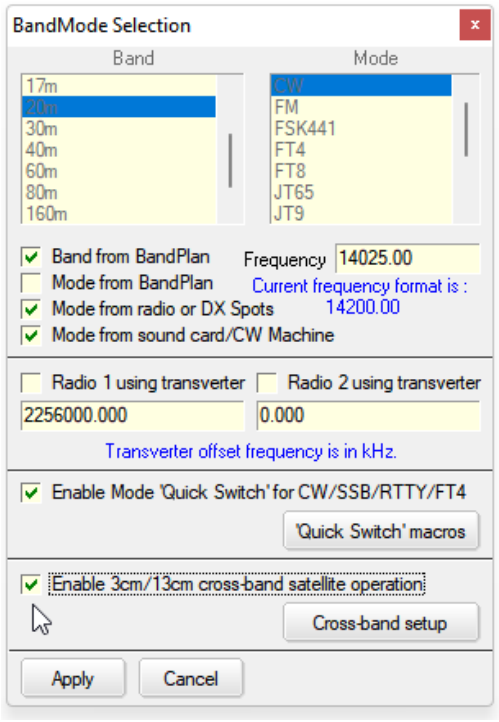
Band	Submode	Lower Freq	Upper Freq	Report	Radio Mode	Power	Stats	Aerial	Radio #	Rotator #	Rotator
3cm	SSB	10489.75500(	10489.99000(	59	USB		N		2	0	0
3cm	SSB	10489.74500(	10489.75500(	59	FM		Y		2	0	0
3cm	SSB	10489.65000(	10489.74500(	59	USB		N		2	0	0
3cm	RTTY	10489.58000(	10489.65000(	599	USB		N		2	0	0
3cm	FT8	10489.54000(	10489.58000(	+0	USB		N		2	0	0
3cm	CW	10489.50500(	10489.54000(	599	CW		N		2	0	0
3cm	CW	10489.50000(	10489.50500(	599	CW		N		2	0	0
13cm	SSB	2400.255000	2400.490000	59	USB		Y		2	0	0
13cm	SSB	2400.245000	2400.255000	59	FM		N		2	0	0
13cm	SSB	2400.150000	2400.245000	59	USB		N		2	0	0
13cm	RTTY	2400.080000	2400.150000	599	USB		N		2	0	0
13cm	FT8	2400.040000	2400.080000	+0	USB		N		2	0	0
13cm	CW	2400.005000	2400.040000	599	CW		N		2	0	0
13cm	CW	2400.000000	2400.005000	599	CW		N		2	0	0

2. Now enable 3cm/13cm cross-band logging:

Click the word "Freq" or "Mode" or "Band" in the log entry pane to open the BandMode Selection form ▶

Click to tick <**Radio 1|2 using transverter**> and enter the transverter offset frequency in kHz ▶

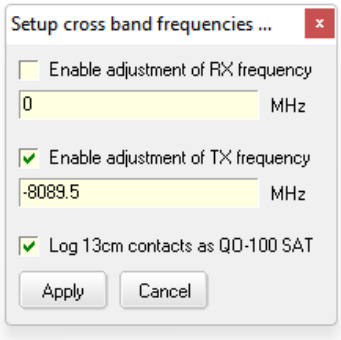
Click to tick <**Enable 3cm/13cm cross-band satellite operation**> ▶



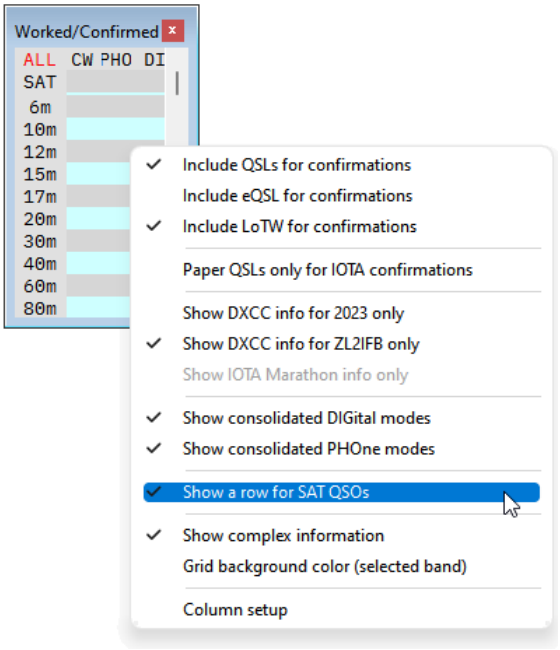
Don't close this form just yet: there's more to do …

Copyright © 2026 Robert C. Furzer K4CY

3. Click the <**Cross-band setup**> button near the bottom of the BandMode Selection form if you need to configure VFO adjustments for the download and/or upload [274] ▶

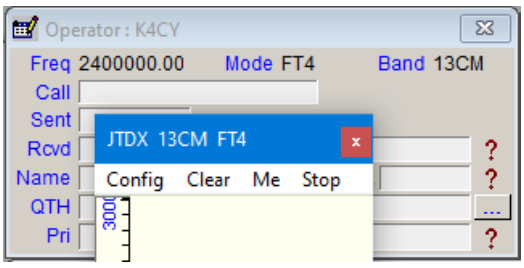


   - Tick <**Enable adjustment of RX frequency**> and type in your downlink frequency offset, if any[275] *i.e.* how many MHz to add to (a positive value) or subtract from (a negative) your reported receiving VFO frequency;
   - Tick <**Enable adjustment of TX frequency**> and type in your uplink frequency offset in the same way, also in MHz;
   - Tick <**Log 13cm contacts as QO-100 SAT**> to have Logger32 automatically tag what *appear* to be 3cm simplex QSOs as cross-band QO-100 satellite QSOs, while you are logging them.
   - Now click <**Apply**> to save the config settings.
4. Give your QO-100 QSOs their own special SAT row in the worked/confirmed table:
   - Right-click the worked/confirmed table to open its configuration menu ▶
   - Click to tick <**Show a row for SAT QSOs**>



Hinson tip: to log ordinary *non*-satellite simplex QSOs on 13cm correctly, open the BandMode Selection form and *un*-tick <**Enable 3cm/13cm cross-band satellite operation**> beforehand, and don't forget to *re*-tick it again before logging further QO-100 QSOs.

Having setup Logger32 correctly, the log entry pane and UDP BandMap header should look *roughly* like this ▶



Q. The magic is sporadic on 6m so decoded calls disappear too soon. What gives?

A. The UDP BandMap's timeout timer can be extended as far as 28,800 seconds (8 hours), so if you leave the radio monitoring FT8 on a mostly-dead band while doing Other Stuff elsewhere, when you return to the shack, callsigns decoded during your absence may still be shown on the BandMap. They should also still be listed in the decode log from JTDX|WSJT-X but the nice thing about the BandMap is that any 'new ones' are highlighted for you, so you know at a glance what magic you missed out on. Don't cry. The magic *will* be back. Probably.

[274] The values shown here are the defaults.

[275] Evidently SDR-Console only reports your 10 GHz (3 cm) receive VFO frequency to Logger32. Therefore, it *appears* to Logger32 that you are both receiving *and* transmitting on 10 GHz (the downlink band), whereas in fact you are transmitting on the 2.4 GHz (13 cm) uplink … so you need Logger32 to log the RX frequency as reported but convert your TX frequency down from 10 GHz to 2.4 GHz for the log.

Copyright © 2026 Robert C. Furzer K4CY

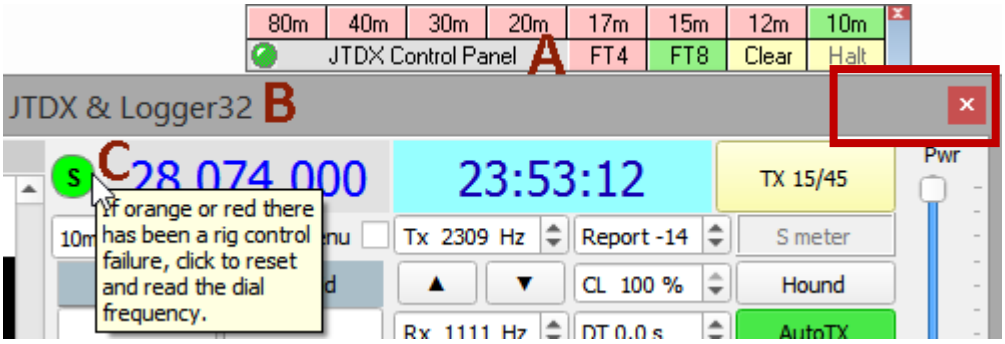
Q. The UDP band map pops up when Logger32 starts and I have to manually dismiss it. Any hints on how to stop this annoyance?

A. Normally, Logger32 automatically saves the positions and status of its windows when closing down normally, and recalls them when it re-starts. The idea is to start your next Logger32 session with the same screen layout that you had when you last closed Logger32.

There's a wrinkle with UDP however: the UDP BandMap is programmed to reappear if it was shown ***or*** if the UDP port was open, when Logger32 was last closed. So, to stop it automatically reappearing, simply close the UDP BandMap ***and*** the UDP port *before* you close Logger32. Next time you open Logger32, the UDP BandMap will not automagically open.

Q. Why don't the JTDX Control Panel band and mode buttons work as described?

A. There is probably a break in the communications linkage *somewhere* between the buttons and your radio. To help you locate the break, here are some clues from a working installation ▼



A: the JTDX Control Panel should *not* say "**Control Panel not connected**". If it *does*, the usual reason is that you have launched JTDX directly from an icon or the Windows start menu, instead of using **Start ⇨ JTDX** on Logger32's UDP BandMap. If your UDP BandMap is also not working (*e.g.* none of the callsigns recently decoded in JTDX are listed), check that both JTDX and Logger32 are communicating through the same UDP port.

B: the caption for the JTDX window should have "& Logger32" … and there should only be a X in the top right corner, not the minimize and restore icons like most other windows. The most likely reasons are that JTDX is simply not running at all (!), or else it was not started using **Start ⇨ JTDX** on Logger32's UDP BandMap. Again the UDP port settings may be wrong.

C: the blob to the left of the JTDX frequency readout should be green, not orange or red as the popup message says when you mouseover it. Try clicking the red or orange blob to reset JTDX's connection to the radio. If that doesn't work, check that your radio is on (!) and check the port settings in JTDX are correct for your radio *e.g.* the COM port number and speed.

If all those things are in order, you may have a problem in the software itself, so try closing and reopening JTDX (preferably using <**Stop**> then <**Start**> on the UDP BandMap menu). Next try closing and reopening Logger32. Finally, if nothing else works, try rebooting the PC.

Before you write to the Logger32 reflector desperately seeking help, confirm that JTDX can control your radio *e.g.* by selecting a different band or mode in JTDX. If that doesn't work, there is evidently a problem with JTDX's CAT capabilities … so fix that first.

Copyright © 2026 Robert C. Furzer K4CY

Q. I don't like the BandMap highlight colors. Can I change them?

A. No. You cannot. They are fixed, hard-coded, set in concrete. Resistance is futile … kidding! Of course you can! This *is* Logger32! Feel free to change the settings of <**DX spot highlight colors**> in the DX Spots pane. Conveniently for those of us who are easily confused, the same DX spot highlighting colors and tooltips can be used in the UDP BandMap. Find lots more coloring crayons in the appendix.

Q. Can I consolidate my log so that JTDX knows about *all* my digimode QSOs?

A. Yes. If your Logger32 logbook contains digimode QSOs that you want JTDX to know about (*e.g.* so that it correctly identifies 'new ones'), you can update the JTDX log with those QSOs:

1. In Logger32 export a *partial* log containing just your "JT-mode" QSOs: JT65/9, FT# *etc*.
2. Name the exported file *wsjtx_log.adi* and save it to a safe location.
3. Replace the default *wsjtx_log.adi* file located in *%LocalAppData%\JTDX* with the new file exported from Logger32[276].

Importing the data from your Logger32 logbook ensures that JTDX has updated DXCC/mode and QSO data and hence highlights any new ones in the decoded callsigns accordingly.

Q. When I log anyone on FT8 on 17m, the logbook band column remains blank. This only happens on 17m - other bands work fine. Any ideas?

A. UDP QSO messages sent from JTDX to Logger32 specify TX frequencies but not the bands. Logger32 looks-up the frequencies in your bandplan to determine the corresponding bands … so check the Bands & Modes table yourself using **Tools ⇨ Setup Bands & Modes**. Do any of the 17m rows cover the FT8 frequencies on that band? If not, define the 17m FT8 frequency range *e.g.* 18.0948 to 18.105 MHz (allowing for audio offsets and VFO errors).

> **Hinson tip**: the same problem may occur with other digimode QSOs if there are no rows covering the particular frequencies used, so it is worth defining a CW entry covering the entirety of each band, since (for historical reasons) CW is generally permitted anywhere on any band. That way, any frequency segment used by an as-yet unidentified mode will at least be associated with the correct band.

Q. Ripe cherries aren't being picked. I see "did not switch to transmit" in the event viewer. What can I do to fix this?

A. If you started JTDX[277] using <**Start**> on the UDP BandMap menu, Logger32 drops its CAT connection to the radio so it can't tell directly if the radio is transmitting. Instead, having sent a UDP message commanding JTDX to call a station, it waits for a response message from JTDX. If that response arrives within three seconds, Logger32 presumes that JTDX is indeed calling the station, commanding the radio into transmit and generating the audio signal. If the response does *not* arrive within three seconds, however, Logger32 presumes its message went astray and generates the "did not switch to transmit" event.

---

[276] File **E**xplorer knows where your *%LocalAppData%* folder is. Type it in just like that to see for yourself.

[277] The instructions and settings are *much* the same if you are using WSJT-X but may differ. Too bad.

Copyright © 2026 Robert C. Furzer K4CY

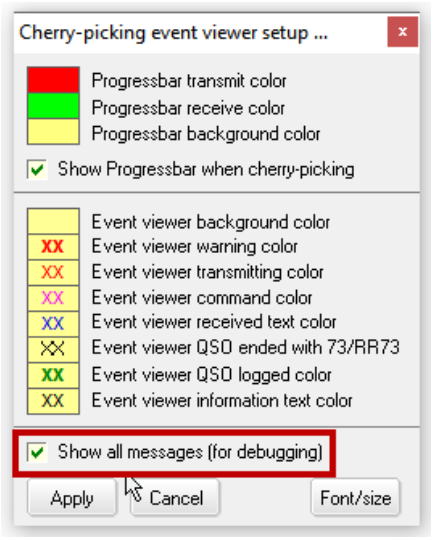
If this happens repeatedly, check your settings:

- In JTDX, open **File ⇨ Settings ⇨ Reporting** tab. <**Accept UDP requests**> must be ticked. If it isn't, click it.
- In JTDX, **Misc ⇨ Accept UDP reply messages ⇨ Any messages** must also be ticked.
- In JTDX, the <**AutoTX**> button near the power slider must be green.
- Finally, in Logger32's UDP BandMap, tick either or both of the top two options under **Config ⇨ Cherry-picking options** ▼



It *may* help to watch the cherry picker event viewer closely as the robot goes about its business.

Maybe turn on verbose mode in the <**Config**> menu ▼ to see its every move play out, blow-by-blow, in glorious Technicolor ▶



Hopefully that will give you sufficient clues to focus in on and solve the problem/s.

Good luck Jim.

```
17:10:44 Cherry-picking AA2MR/6 started
      AA2MR/6 sent:  ZL2IFB AA2MR/6
      AA2MR/6 not worked on 20m/FT8
      AA2MR/6 called me, I'm replying
17:10:44 Sent message to JTDX to call AA2MR/6
17:10:44 JTDX TxEnable on
17:10:45 JTDX Transmit on
17:10:45 JTDX is transmitting to AA2MR/6
17:10:46 JTDX Decoding off
17:10:58 JTDX Transmit off
17:11:14 JTDX Decoding on
17:11:14 AA2MR/6 replied:  ZL2IFB AA2MR/6 R-15
17:11:15 JTDX Transmit on
17:11:15 JTDX is transmitting to AA2MR/6
17:11:16 JTDX Decoding off
17:11:28 JTDX Transmit off
17:11:44 JTDX Decoding on
17:11:44 AA2MR/6 replied:  ZL2IFB AA2MR/6 73
17:11:45 JTDX Transmit on
17:11:45 JTDX is transmitting to AA2MR/6
17:11:45 JTDX Log QSO request
17:11:45 JTDX QSO Logged
17:11:46 JTDX Decoding off
17:11:48 AA2MR/6 QSO logged
17:11:58 JTDX Transmit off
17:12:14 JTDX Decoding on
17:12:15 JTDX TxEnable off
17:12:16 JTDX Decoding off
17:12:29 JTDX Decoding on
17:12:29 JTDX turned off Tx Enable
17:12:29 Sent Tx Halt message to JTDX
17:12:29 Cherry-picking AA2MR/6 ended +11
17:12:30 JTDX Decoding off
```

It is a cherry-picker,

not a vacuum cleaner

Bob, K4CY

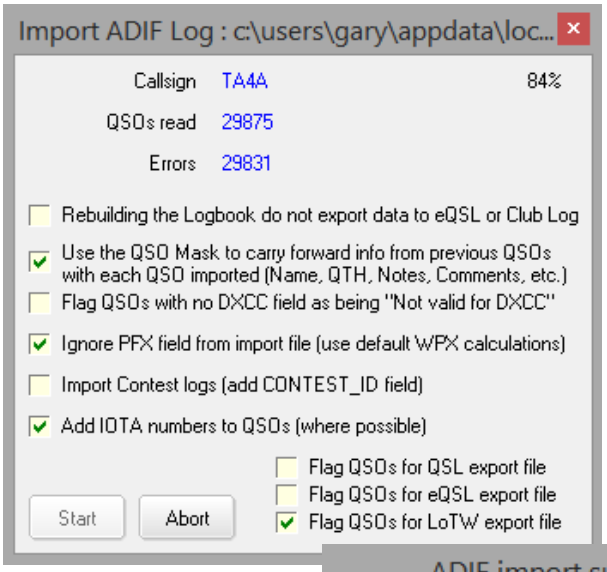
Copyright © 2026 Robert C. Furzer K4CY

Q. Some FT# QSOs are missing from my Logger32 log. How can I retrieve them?

A. Digimode QSOs occasionally go 'missing' in the course of being transferred from the digimode software (*e.g.* JTDX) into Logger32. That is always a possibility with UDP messaging, designed for speed and efficiency rather than communications integrity.

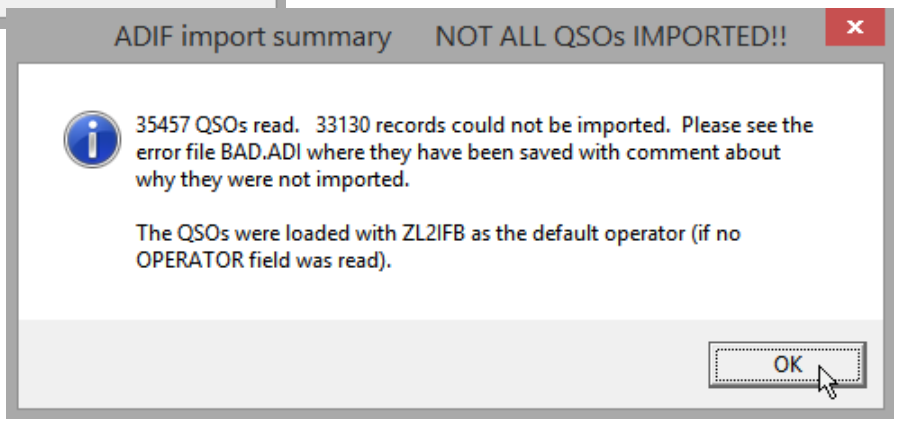
Assuming the QSOs were in fact completed and logged in JTDX or WSJT-X, they are most likely recorded by the digimode software in anADIF log. You might like to import that ADIF file into Logger32 from time to time. Any QSOs that had been successfully transferred and logged already *should* simply be skipped in the process … but any issues may result in your log being stuffed with unwanted QSOs or cause other problems … so first, *before you run the import*, export your log as an ADIF file using **File ⇨ Export Logs ⇨ Export ADIF (.adi) file**[278]. Save that ADIF file somewhere safe just in case.

Now use **File ⇨ Import file ⇨ ADIF (.adi) file** then find your JTDX ADIF file on disk: look in the *%localappdata%* folder (type that exact string complete with its percentages into the folder field of Windows Explorer: Windows will expand it to point at *your* local application data folder).



◀ Choose your settings on the import form *e.g.* to carry forward information from previously-logged QSOs with the same stations, add IOTA numbers for stations whose DXCC prefix unambiguously corresponds to a specific IOTA reference, and to flag the QSOs for export to LoTW.

Click <**Start**> to run the import. You will see the callsigns and QSO numbers rapidly updating, pausing occasionally for breath. The percentage shown top right shows how much of the ADIF file has been processed so far. When complete, a summary pops up ▼



Don't panic at the shouty warning banner: you didn't really expect *all* those QSOs to be imported, did you? Most of them had already been transferred to and logged by Logger32 … but some may be new.

[278] The file import function automatically backs-up your log and key *.INI* files for you before importing, saving the log in Logger32's internal database format. If necessary, you can restore those files to the *C:\Logger32* folder and/or wherever your log is stored, then recalculate statistics using **Tools ⇨ Database maintenance ⇨ Recalculate statistics** to get back to where you were … but recreating your log from a saved ADIF file is quicker and safer.

Copyright © 2026 Robert C. Furzer K4CY

In the example here, I found 2,327 (35,457 less 33,130) new QSOs – rather more than I expected but, hey, I had been busy on FT8 while beta testing both JTDX and Logger32 updates. Looking at my updated log, I found a lot of near-duplicate recent QSOs differing only slightly in the QSO times. It appears a little timing discrepancy has crept in between JTDX and Logger32, so I will need to work on that. Meanwhile, I restored my log from the pre-import backup, rather than manually locate, check and probably correct over 2,000 dubious QSOs in my log.

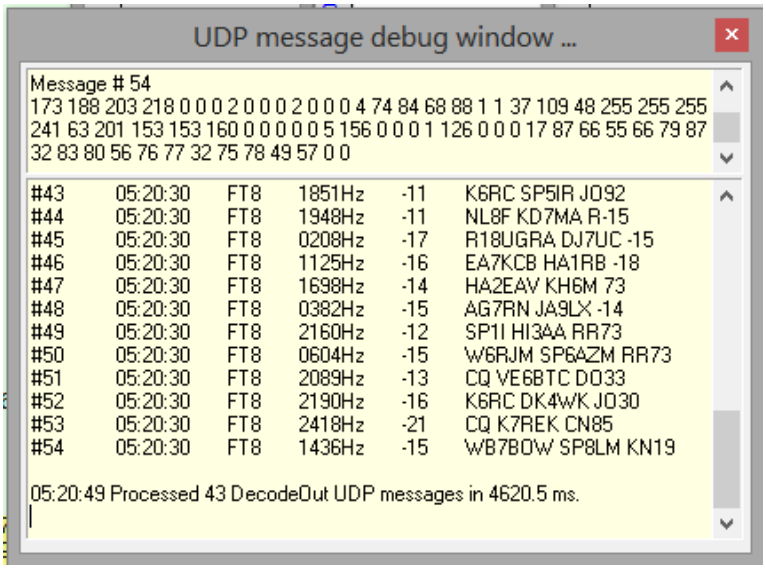
Q. My UDP BandMap is empty. What have I broken *this* time?

A. It *may* not be your fault, at least not *this* time. It's probably something on this list. Work your way down from the top, systematically fault-finding:

- JTDX isn't running. Doh! So start it.
- JTDX is running but, for some reason, the software isn't decoding digital signals, possibly because your radio is not tuned to one of the usual watering holes for whichever mode you are using, or the band you are currently monitoring is dead. **Is the waterfall displaying the expected signals?** If you are convinced the band is open, check your radio and antenna. Is your attenuator on turbo? Can you hear signals in the headphones? Do you have headphones? ARE YOU HARD OF HEARING? Maybe there has been a solar flare. Maybe squirrels or rats have eaten your coax. Maybe your receiver has packed up. Try a different band, ideally one that is almost certainly humming with digital activity. Maybe the cable from the radio to your PC is disconnected or broken. Maybe your PC sound system has gone to sleep, or naughty Windows has oh-so-helpfully reconfigured itself so JTDX is no longer using the sound system that is physically connected to your radio, or it has set the input level so low that you'd miss the EMP pulse from a nearby nuclear bomb. If there *are* signals on the waterfall but no decodes, double-check that JTDX is set to the appropriate digmode. Whatever, if there are no decoded messages for JTDX to send to Logger32 by UDP, there are none for Logger32 to receive via UDP and display on the UDP BandMap.
- JTDX has stopped sending decoded messages via the UDP port that Logger32 is listening on (port 2237 by default) – check under the **F2 settings** ⇨ **Reporting** tab that the correct UDP port is selected. If F2 settings isn't working properly either, or if JTDX has become unresponsive or is otherwise 'playing up', close and restart JTDX. See if that helps. If not, try reverting to a previous version.
- You have selected <**Show only highlighted callsigns/gridsquares**> on the UDP BandMap **Config** menu … and at present there is nothing interesting to show you, only boring callsigns and grids. De-select it and watch in awe as the BandMap soon fills … with boring stuff. [This one often catches me out: having set the highlight-only option on a busy band such as 20m, I QSY to a different, quieter band … and wonder why the UDP BandMap *appears* to have stopped working because there are no 'new ones' to highlight and show. Doh!]
- Logger32 has stopped listening for UDP messages from JTDX on the correct port (2237 by default). Check the UDP port by right-clicking the **UDP** panel on the status bar then click <**Change JTDX UDP port # for** [rig name]> and if necessary change the configuration to match the UDP settings in JTDX.

Copyright © 2026 Robert C. Furzer K4CY

- Logger32 is listening on the correct UDP port but for some reason the UDP traffic is indecipherable, or missing. On the UDP BandMap Config menu, click to enable <**Show UDP message debug window**> for a while. Soon after the end of a decoding period, JTDX sends a blast of UDP messages that you *should* see streaming through the debug window, roughly like this ▶



  If not, is your shack LAN still working properly? Are you still connected to the Interwebs? Is the Ethernet light flickering like normal – yes, that almost inaccessible little green LED on the rear panel of the PC, on the Ethernet card? How about your WiFi subsystem: does that show signs of life?

- Logger32 has packed up. Some users report sporadic issues with Logger32 – typically occasional, seemingly random crashes, but who knows what other funniness are going on between JTDX, Logger32, other apps and utilities on your computer, and the PC hardware? Stop and restart Logger32 anyway. Does that fix it? If not, look here.

- If Logger32 isn't running and positively refuses to run, Windows has packed up. Is there anything on the display? Is the screen on? Reboot your PC, preferably a cold reboot *i.e*. power it down completely, perhaps even unplug the power cable or remove the battery pack and then hold the power switch closed for a few seconds to drain that last few remaining joules keeping the motherboard from total CPU-arrest. Then reapply power and start it up. If nothing at all is happening, your PC has packed up. It may be something as simple as a pulled power cord or off-switch at the wall socket, a blown fuse/circuit breaker, or a failed PSU inside the PC. It could be a CPU, motherboard or memory failure, or overheating … or a fail-safe control that detected a potentially dangerous condition and shut things down before everything melted down. Is the PC still there, hidden away under the shack desk? Has it been abducted by aliens?

Q. Where in the world are all those digimode stations located?

A. When you mouseover an individual callsign on the UDP BandMap and hover the cursor there for a moment, a popup message tells you a little about the station, including its location.

To have Logger32 plot all the decoded stations on a world map, open the Tracking window (using **View** ⇨ **Show Tracking Window**), then click to open the **JTDX** tab.

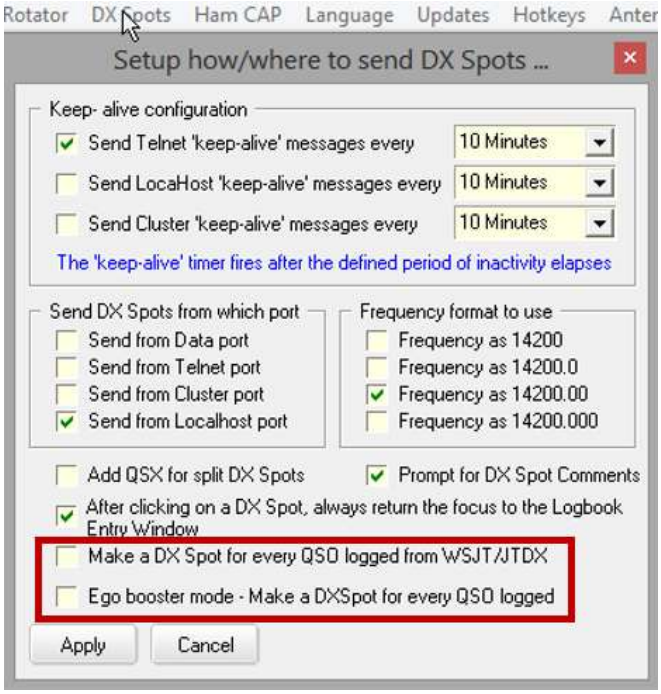
Right-clicking the JTDX map reveals a menu option to plot *all* the decoded stations, or just those stations that pass through your UDP BandMap filtering (typically just the ones you need or want to work).

Copyright © 2026 Robert C. Furzer K4CY

Q. I'm running Logger32 with JTDX, latest versions, with autologging turned on. I was politely informed that I was making a big nuisance by creating a DX spot for every FT# contact I made. I didn't realize this was happening. Problem is, I can't figure out how this behavior got started nor how to stop it ...

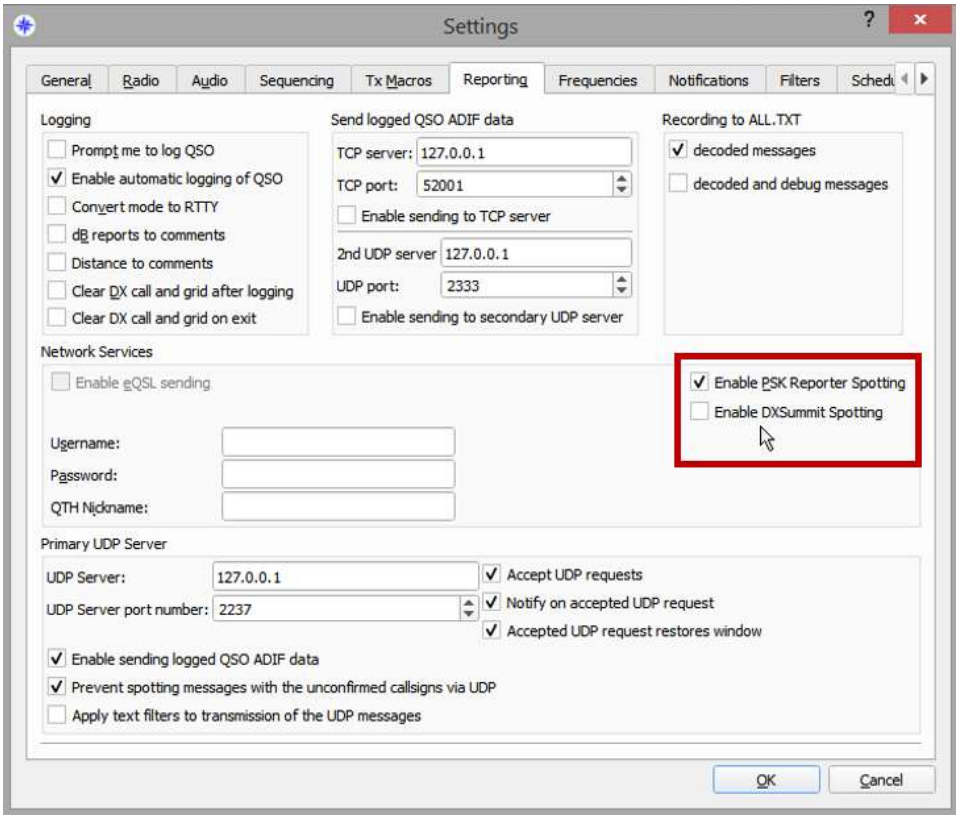
A.

Check your settings under
Setup ⇨ DX spots ▶



Logger32 can generate and send DX spots to the DX cluster network either for *every* QSO added to your log, or 'just' those made and logged in JTDX, then passed through to Logger32. Either way, the function is non-selective, whereas the DX cluster network is intended for spotting "DX stations" (whatever that actually means – presumably *not* every single QSO you log unless you are *very* lucky!).

In addition, JTDX can generate and send DX spots for all the stations you have decoded to the DX cluster network via PSK Reporter (which is designed to receive and process a tsunami of automatically-generated digimode spots) and/or DX Summit (which forwards them to the regular DX cluster network, intended for low-volume high-quality human-generated *DX* spots).



◀ In JTDX, check the configuration options under **F2 Settings ⇨ Reporting** tab.

Feel free to send decoded callsigns to PSK Reporter if you wish … but kindly avoid spamming the DX cluster network with digimode spots by *not* enabling DX Summit reporting.

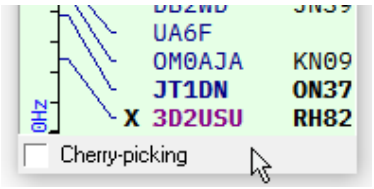
Copyright © 2026 Robert C. Furzer K4CY

Q. I use the DX spot filters to cut down on irrelevant spots, so why isn't the UDP BandMap filtered in the same way?

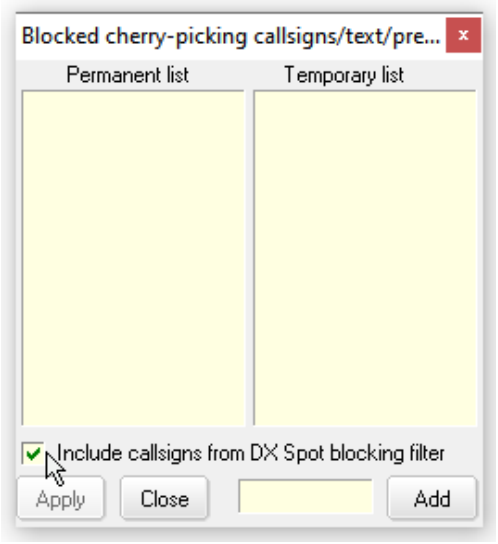
A. The clue is in the name: Logger32's *DX spot* filters are for *DX spots* arriving from the DX cluster network. Stations shown on your UDP BandMap are *not* DX spots from DX cluster!

You can still filter what appears on the UDP BandMap though, in any of three ways:

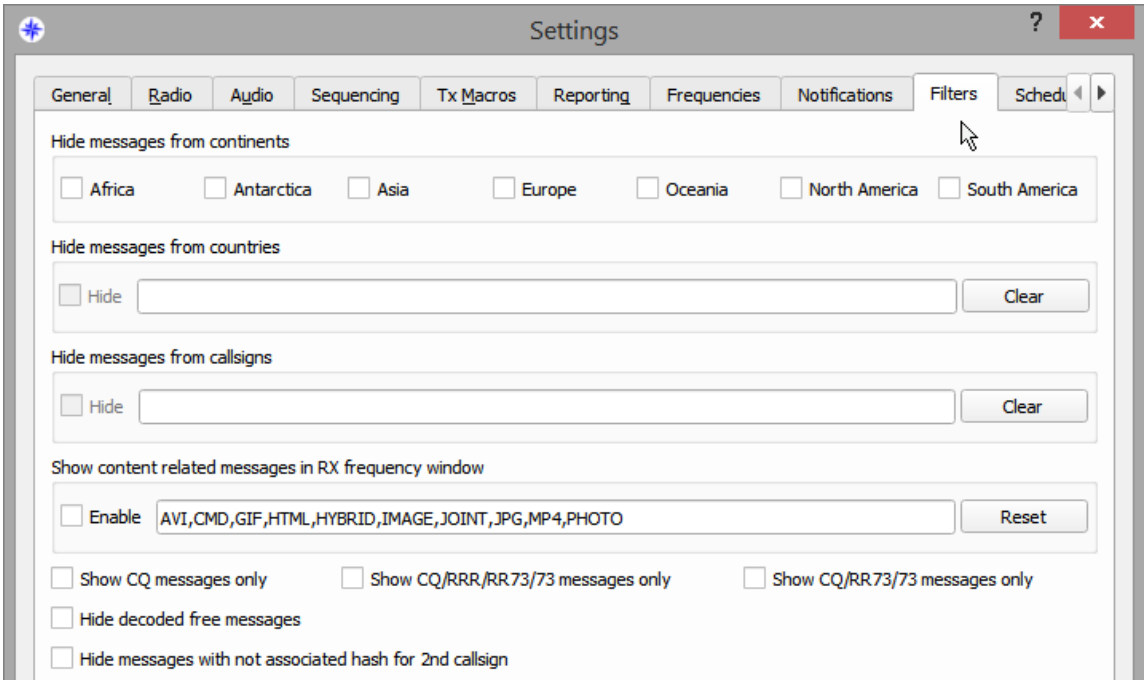
1. The UDP BandMap blocking filter can be configured to apply your DX spot filtering as well:
   - Open the blocking filter configuration form by right-clicking the bottom right corner of the UDP BandMap, to the right of "Cherry-picking" ▶



   - Tick <**Include callsigns from DX Spot blocking filter**> at the bottom of the form, then click <**Apply**> to save and start using the filter ▶



2. Apply the filtering at source in JTDX using the **F2 Settings ⇨ Filters** tab ▼



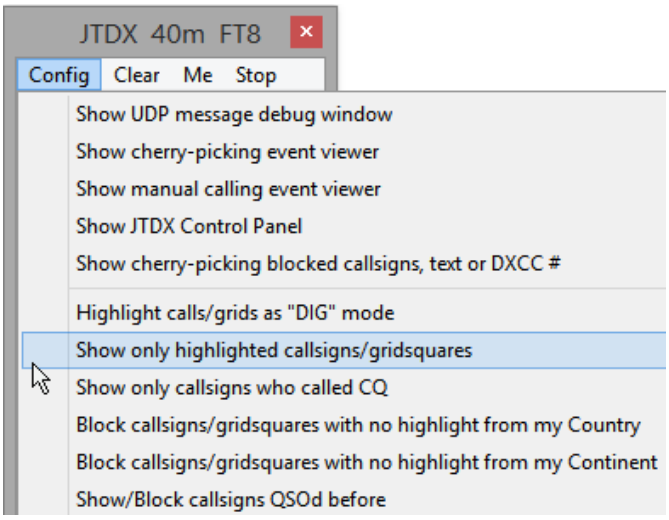
The filtering in JTDX is less effective than Logger32's. For example, if you are CQing, the JTDX autoresponder may respond even if a supposedly blocked station calls you.

Copyright © 2026 Robert C. Furzer K4CY

3. In the UDP BandMap's **Config** menu, you have the option to hide decoded callsigns that are not highlighted as new ones, or are not CQing.

Show only highlighted callsigns/gridsquares is a good place to start ▶

There are a couple of country and continent ▶ blocking options too ▶



Hinson tip: <**Show only highlighted callsigns/gridsquares**> can be useful on a busy band when the UDP BandMap is overflowing with stations, most of which are quite ordinary. Showing only stations that qualify as 'new ones' for you hides the clutter. It ensures that those 'new ones' are actually shown and don't get discarded with the overflow … unless there are too many 'new ones' to show in the available space, anyway. What a nice position to be in!

Q. After a spell on SSB, when I return to FT# with JTDX, why doesn't autologging work any more?

A. It sounds as if Logger32 is disconnected from JTDX.

Normally, if correctly configured, when Logger32 launches JTDX, it waits a few seconds for JTDX to start and get itself ready to accept inbound connection requests, then opens the data connection and changes the JTDX title bar to "JTDX **& Logger32**". Digimode QSOs are passed to Logger32 automatically after they are logged in JTDX. Using the JTDX Control Panel, we can command JTDX to change band on the radio (using *its* CAT connection) or swap to another JTDX-supported digimode. We can also close JTDX from the UDP BandMap's <**STOP**> menu.

If, after SSBing, JTDX starts up but its title remains stuck at "JTDX", try increasing the startup delay option on the UDP BandMap until it works reliably. Or get a faster PC.

Q. I don't want to launch JTDX from the UDP BandMap. In fact, I don't even want to use the UDP BandMap. Is it OK if I just start and run JTDX independently?

A. Yes!

Your radio's CAT connection cannot normally[279] be shared between applications, though, so if JTDX is using it (*e.g.* so you can select bands and send the mode-specific VFO frequencies to the radio from JTDX, or to use **F2 settings ⇨ Radio tab ⇨ Split Operation - Rig** to handle split operation within a mode segment), Logger32 cannot do so at the same time – and *vice versa*.

[279] Actually, it can, but additional software is required in the form of a 'port splitter' that opens and controls the radio's CAT port, communicating separately with the logging, digimode or other software, passing their CAT commands through to the radio and sharing CAT responses returned from the radio with all the other software. This adds complexity and delays … and is not in scope of this Logger32 User Manual.

Copyright © 2026 Robert C. Furzer K4CY

In fact, JTDX may refuse to start up if it cannot grab the CAT connection because Logger32 is hogging the port already.

Rather than closing down Logger32 in order to be able to start JTDX, simply tell it to drop its CAT connection by right-clicking the <**Radio**> panel at the bottom of Logger32's main window, then clicking <**Close port – [**Radio name**]**>. The text "Radio 1|2" in the panel is red when the CAT port is closed, and blue when it is open so you can see the status at a glance, provided you are not red-blue color-blind anyway.

Even if you don't want to use the UDP BandMap, provided you are running JTDX on the same PC as Logger32, or on the same shack LAN, you may find it useful to enable UDP messaging within JTDX and open the UDP port in Logger32 in order that:

- Any FT# QSOs that you complete and log within JTDX are passed via UDP to Logger32 to be incorporated in the open logbook. This means you don't need to transfer and import the ADIF file from JTDX into Logger32 to maintain your log.
- If you click FT# DX spots in Logger32, Logger32 tells JTDX via UDP messaging to QSY your radio accordingly.

In other words, Logger32's UDP port *can* be used *without* the UDP BandMap.

Q. Will Logger32 work with MSHV?

A. Yes, we believe so. Both MSHV and JTDX are built around the same core code as WSJT-X, sharing many features including UDP messaging to/from logging programs. A happy Logger32 user even confirmed that the cherry picker "is going very well" in conjunction with MSHV.

Beyond that, there are wrinkles here and there. All these programs, including Logger32, are experimental in nature, in keeping with the hobby … so feel free to experiment with your software and hardware configurations, and by all means share any innovative arrangements and issues through the Logger32 reflector. Although the Logger32 team can only directly address bugs, flaws and improvements *within* Logger32, we are interested in the bigger picture. We're also DXers and experimenters, happy to share interesting discoveries.

Q. Does Logger32 work with VarAC for VARA?

A. Sort of. Damian M0BKV reports via the Logger32 reflector that he is using VarAC to make and log VARA HF QSOs into Logger32 having:

1. Defined VARA HF as a submode of DYNAMIC in his *C:\Logger32\ADIFmodes.txt* file;
2. Added an entry to launch *VarAC.exe* from his UDP BandMap's <**Start|Stop**> menu; and
3. Been forgiving when VarAC *sometimes* launches blankly, without his information, and after closing, seems to prevent JTDX launching cleanly until Logger32 is restarted.

Q. What about FT2? It has been out for *days* and I'm clearly impatient.

A. As with VarAC just above ▲, the digimode-du-jour can be added manually to your *C:\Logger32\ADIFmodes.txt* file, ending up with a line similar to this:

MFSK =FSQCALL,FST4,FT4,**FT2,**JS8,MFSK4,MFS....

FT2 is defined in ADIF as a submode of MFSK. The order of the MFSK submodes in *ADIFmodes.txt* does not matter, but submodes such as FT2 should *not* be defined as modes.

Copyright © 2026 Robert C. Furzer K4CY

Add the appropriate frequency slots to your Bands & Modes table (at least one) and put a Y in the stats column for at least one FT2 entry, telling Logger32 to collect statistics for the new submode.

Changing *C:\Logger32\ADIFmodes.txt* only affects QSOs logged subsequently. Any QSOs already logged are *not* updated automatically, but you can edit your log manually, changing a logged QSO's mode to any mode or submode now defined in *C:\Logger32\ADIFmodes.txt*

After doing that, you might like to recalc your statistics to be *sure* all your logged QSOs on recognized modes are counted correctly.

Hinson tip: it takes months to debate and agree changes to the ADIF specification and then update the online QSL systems such as LoTW for new ADIF-defined modes/submodes, so meanwhile you may choose to upload your QSOs with a generic "DIGI" mode … and by all means continue experimenting with the new protocols in the amateur tradition. Once a new ADIF mode/submode is agreed and implemented, you can edit the mode for any logged QSOs and re-upload them to LoTW to get credit for the new mode/submode, if that matters to you.

Q. Cherry-picking works great with highlighted call signs but doesn't pick-up highlighted grid squares. Is it supposed to be that way or have I set it wrong?

A. If you have told the cherry-picker robot only to call highlighted callsigns, then it calls highlighted callsigns, regardless of grids. Likewise if you have told the robot to ignore non-LoTW users, it does so, even if they are in new grids. The highlighting and LoTW user checks take place before even thinking about the grids. The robot is stupid but obedient: it does as it is told. In short, don't rely on the robot to read your mind.

Copyright © 2026 Robert C. Furzer K4CY

16 QSLing

> "A postcard is
> a piece of happiness
> that fits in an envelope"
>
> *ChatGPT*

Here is the typical cascade process through which contacts with other stations flow towards awards such as DXCC. Confirmations (QSLs) are a critical part of the process because (believe it or not) some hams have been known to claim incomplete or even nonexistent/fake QSOs.



Initially, contacts with places such as North Korea are wanted, **needed** or *craved*, depending on just how desperate you are.

Once a QSO is made and logged (the station and the place has been **worked**), you then request and wait patiently for a **confirmation** – typically an electronic confirmation such as a match on LoTW, and/or a paper QSL card. This may take anywhere between seconds and decades to arrive, and may involve donating towards the costs of QSLing and/or making the QSO itself (*e.g.* transportation and licensing fees for some DXpeditions).

If you wish, one or more confirmed QSOs can be **submitted** (sent in, claimed) for an award.

Submitted QSOs are then checked and **verified** by the award administrators.

Provided the award administrators are satisfied that the QSOs are valid for the award (*e.g.* on permitted bands and modes), and that the confirmations are legitimate (*e.g.* properly licensed stations, not fake cards), they may be **granted**, locked in to the award.

Provided sufficient valid QSOs have been granted (the award rules generally define a minimum number *e.g.* at least 100 countries for DXCC), you can apply for the award. This typically involves paying a fee for the award certificate or plaque, packing and postage and a contribution towards the costs of promoting and administering the award.

Logger32 can help with that.

Copyright © 2026 Robert C. Furzer K4CY

16.1 Making QSL chores slightly less onerous

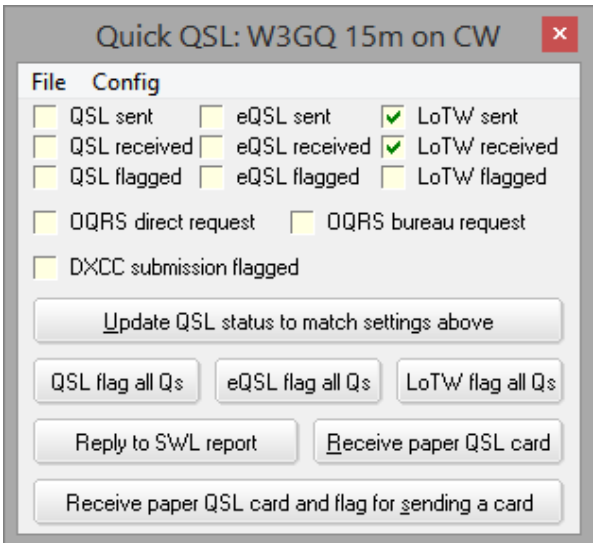
Administration of QSL cards and other forms of confirmation can be quite a chore, especially if you are an active DXer chasing awards or qualify as "DX" for many of the stations you work *e.g.*:

- Confirming all your QSOs routinely (individually or in batches) through online/electronic services such as Logbook of The World, Club Log and eQSL.
- Downloading online/electronic confirmations, updating your log to record receipt, and potentially updating your awards status accordingly.
- Picking out and arranging to prepare and send QSL cards for notable QSOs including 'new ones' that you still need to be confirmed for various awards.
- Checking, recording, responding to and filing away QSL cards received, including SWL reports.
- Looking up QSL routes for DX stations, such as their preferred QSL methods, QSL managers, online QSL systems, direct QSLing postal addresses *etc*.
- Correcting errors in your log, updating awards, and potentially re-confirming QSOs with the correct details.
- Following-up on missing QSLs, particularly those for which you have donated towards card and postage costs hence you reasonably expect a response ("Sorry, not in log" at the *very* least!).

Logger32 has functions to support/enable most of those activities, reducing the burden of QSLing[280], particularly the record-keeping and electronic aspects. Writing and posting QSL cards, collecting the mail, recalling memorable QSOs, admiring the artwork, considering the comments received, and filing them away in shoeboxes is still down to you though!

16.1.1 Quick QSL form

Right-clicking a QSO in the Previous QSO pane opens the <**Quick QSL**> form ▶



- **QSL|eQSL|LoTW sent|received|flagged**: the top half of the form lets you set or clear various flags for the QSO. Simply click the relevant boxes to tick or untick them and then …
- **Update QSL status to match settings above:** after changing any flags, don't forget to *click this button to save the changes*. Flags are not updated and saved automatically as you change them, unfortunately, whereas the buttons on the lower half of the form are actioned immediately.
- **QSL|eQSL|LoTW flag all QSOs**: these 3 buttons set the respective 'send QSL' flags unless the respective QSLs are already flagged as sent, *for all the QSOs currently displayed in the Previous QSO pane*. This is an easy way to confirm all your QSOs with someone at once, for instance if

[280] It *is* a burden, especially if you are not personally interested in confirmations. Please spare a thought for the people who ask you to confirm QSOs. As an accomplished DXer, your shack wall may be *plastered* with certificates and plaques, but think back to the early days when you were thrilled to get new ones confirmed.

Copyright © 2026 Robert C. Furzer K4CY

you have received 2 or 3 QSL cards from someone (suggesting that they like to QSL), but have logged say 5 or 10 QSOs with them. It helps if your QSOs are all on different band/mode combinations since none of the QSLs are duplicates … but you can always un-flag individual QSOs to avoid over-doing things – particularly for QSL cards. Superfluous electronic confirmations only mildly inconvenience a few electrons, whereas duplicate cards waste trees.

- **Reply to SWL report**: see below.
- **Receive paper QSL card**: sets the <**QSL received**> flag and puts today's date in the QSLRDATE field.

> **Hinson tip**: pressing <**Ctrl**+**R**> (as in 'QSL **R**eceived') from the log entry pane does the same thing directly, without having to open the Quick QSL form.

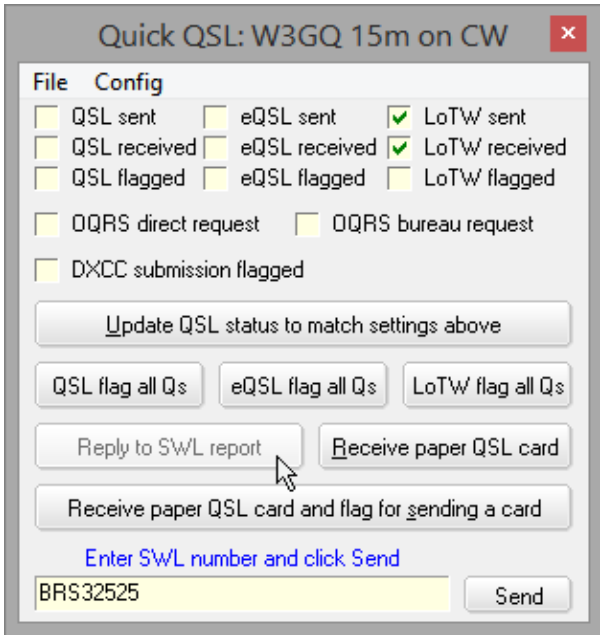
- **Receive paper QSL and flag for sending a card**: clicking this button or pressing <**Ctrl**+**S**> from the log entry pane does the same as above *and* flags the QSO record in your logbook to send a paper QSL. The next time you export a QSL file, this QSO will be included.

Any information already in the logbook QSL received fields (such as the previous date when a card was received) will be overwritten when using the Quick QSL features.

Closing the window *without* having clicked either <**Update QSL status to match setting above**> or the Quick QSL buttons makes no changes to the log. Any QSL flags you meant to set or clear will remain unchanged, as if you had never even opened the form.

16.2 Responding to SWL cards received

When you receive a QSL card from a **S**hort **W**ave **L**istener, the easiest way to look-up and check the QSO on the card is to type the callsign of the station you were reportedly heard working into the log entry pane which opens the previous QSOs pane. Look for the QSO and check the details on the card. If it all looks in order (meaning the SWL probably *did* hear you make the QSO as claimed), right-click the QSO in the previous QSOs pane, then click the <**Reply to SWL report**> button on the Quick QSL form to open a new section at the bottom of the form.



Enter the SWL's number or name, then click <**Send**> ▶

Clicking <**Send**> doesn't magically turn your PC into a QSL card-writing envelope-stuffing robot. It simply generates an ADIF QSO record and appends it to the following file:

C:\Logger32\[Log name]_SWL_dump_file.adi

Copyright © 2026 Robert C. Furzer K4CY

For example, here's mine after I clicked the Send button on the form shown above:

```
C:\Logger32\ZL2IFB_SWL_dump_file.adi SWL export file created by
Logger32 Version 4.0 Build 274. File created on 07/09/2021
08:01:29

<EOH>

<CALL:8>BRS32525 <DISTANCE:5>13367 <BAND:3>15m
<APP_LOGGER32_STATION_IN_QSO_WITH:4>W3GQ <CNTY:10>NC,Catawba
<CONT:2>NA <CONTEST_ID:10>ARRL DX CW <CQZ:1>5 <DXCC:3>291
<FREQ:9>21.025690 <GRIDSQUARE:6>EM95mn <ITUZ:1>8 <MODE:2>CW
<OPERATOR:4>ZM4G <PFX:2>W3 <LOTW_QSL_SENT:1>Y <LOTW_QSL_RCVD:1>Y
<QSO_DATE:8:D>20190216 <TIME_ON:6>203414 <RST_RCVD:3>599
<RST_SENT:3>599 <STATE:2>NC <STX:3>217 <TIME_OFF:6>203414
<LOTW_QSL_SENT:1>Y <LOTW_QSL_RCVD:1>Y
<APP_LOGGER32_QSO_NUMBER:6>101473 <FREQ_RX:9>21.025690 <EOR>
```

When you get a moment, use your ADIF file to generate labels or QSL cards with a suitable printing program or online service, update your online log, or compose emails to the SWLs … or simply display the SWL records on-screen (*e.g.* with ADIF Master) and physically write out QSL cards to send to them[281]. Afterwards, either *delete or archive the ADIF file* to avoid sending duplicate cards to the same SWLs the *next* time you have a spare moment to respond to SWL QSLs.

Hinson tip: although it is not appropriate to log an SWL's callsign in the same way as you log normal QSOs (since you didn't make contact with them on the air), feel free to add a note/comment to the QSO they reported and/or add an informational entry to your log immediately before or after that QSO to record the fact that you received and hopefully responded to an SWL report. Informational entries have the advantage of appearing automatically in the previous QSOs pane if you add another information entry later for the same SWL, whereas you cannot search the log for specific notes/comments using Logger32's log search function, unfortunately.

Hinson tip: be kind to SWLs. Many of us first got hooked on the hobby by listening on the bands.

I'm using Logger32 for a long time, I tried all others (paid or not) but I can't run away from Logger32. Hihihi. It has almost all that I need in a simple way, intuitive and very light. With all the new technologies being integrated to our HAM life, it would be good to keep up-to-date. Thanks for the good job keeping such a good Logging program up-to-date and free.

Alex PY2SEX

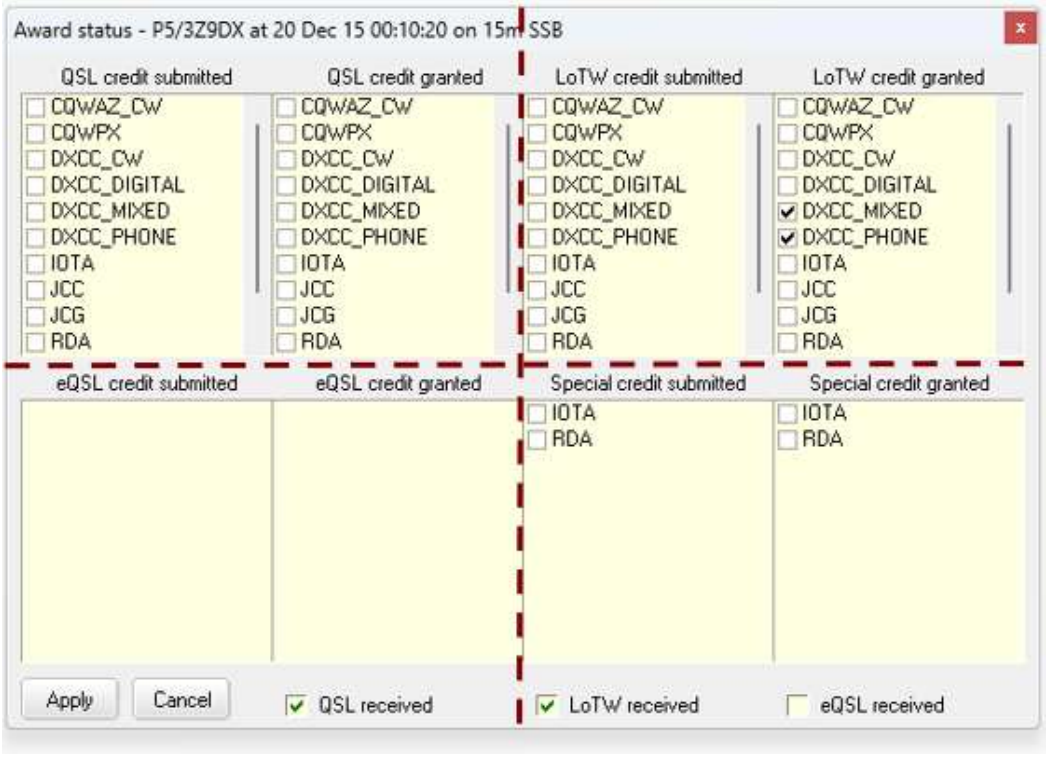
[281] Be nice! Remember that we *all* started out as SWLs: even if our first QSO was with the very first station we ever heard, we still had to *listen* on the short waves to make that QSO. Keen SWLs are a valuable source of newcomers to the hobby. Please encourage them - make them welcome.

Copyright © 2026 Robert C. Furzer K4CY

16.3 Tracking confirmations received

In order to track and record the QSLs that we have sent, received and had verified for awards such as DXCC, **right-click a QSO in the logbook**. Logger32 highlights that QSO and opens a menu.

Click <**Set/show award credits**> on the menu to open this table ▶



The caption reminds us which QSO this is.

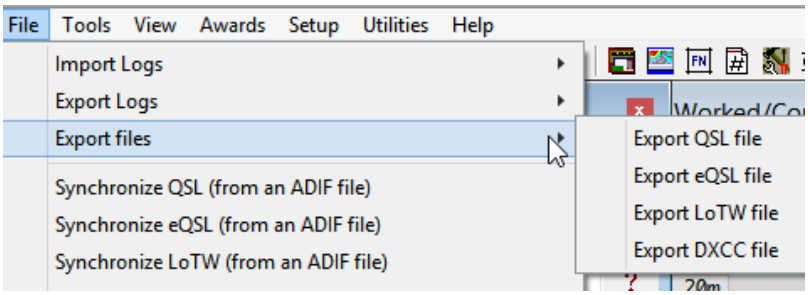
The four quadrants of the table show any QSL card, LoTW, eQSL and 'special' (other) confirmations or credits that we have **submitted** for any of the listed awards, plus any that have been **granted** (checked and accepted by the award administrators) for those awards. Click to tick or un-tick any of them – they are all toggleable.

Three selector boxes at the bottom of the form show whether any confirmations have already been received for the selected QSO: these too are toggleable.

When you're done with any updates, click <**Apply**> to commit any changes and close the form, otherwise <**Cancel**> or the top right corner X to scrap any changes and exit/abort the form.

16.4 Sending QSL cards and other QSO confirmations

Four types of QSL data files can be generated from your log using the respective functions under **File ⇨ Export files**[282] ▶



- **Export QSL file** lets you send information about logged QSOs flagged with "Send QSL" to QSL card printing services, online QSO maps *etc*. in ADIF or CSV formats.

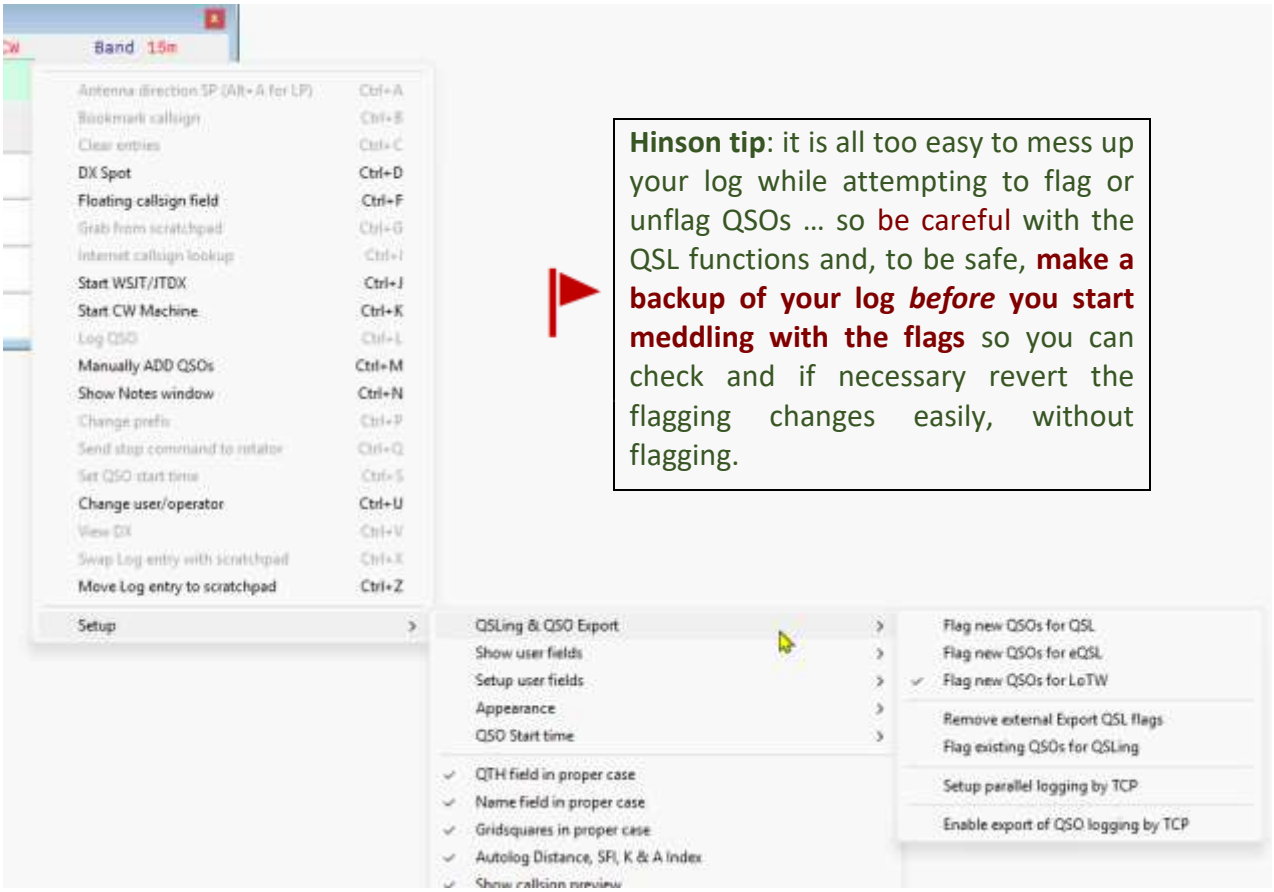
[282] The File ⇨ Export **Logs** function also generates ADIF files *regardless* of whether the QSOs have or have not been flagged to send QSLs. File ⇨ Export **Logs** is *not* intended for conventional QSLing with cards.

Copyright © 2026 Robert C. Furzer K4CY

- **Export eQSL file** extracts the details of QSOs flagged with "Send to eQSL" to an ADIF file on disk so you can then upload it to the **e**lectronic **QSL** system.
- **Export LoTW file** does essentially the same thing but for QSOs flagged (marked) "Send to LoTW". You will need to digitally sign the exported file and upload it to ARRL's **L**ogbook **o**f **T**he **W**orld system using the TQSL program. Read on to find out how to flag QSOs to be exported to LoTW, automatically, as you log them.
- **Export DXCC file** generates an ADIF file containing details of QSOs that have been confirmed in LoTW or by QSL cards and flagged as such in your log, that you intend to submit for one or more DXCC awards using ARRL's online DXCC system (also known as "easyDXCC").

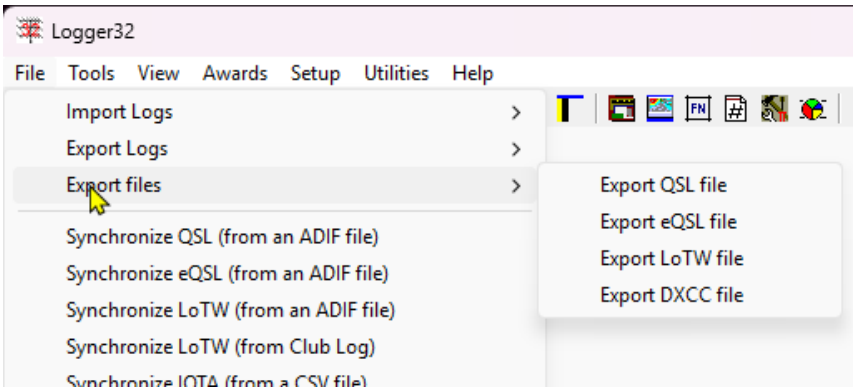
16.4.1 Flag logged QSOs *automatically* for QSLing

There are some powerful QSLing functions available by right-clicking any data entry field in the log entry pane, then clicking **Setup ⇨ QSLing & QSO Export ▼**



Hinson tip: it is all too easy to mess up your log while attempting to flag or unflag QSOs … so be careful with the QSL functions and, to be safe, **make a backup of your log *before* you start meddling with the flags** so you can check and if necessary revert the flagging changes easily, without flagging.

- **Flag new QSOs for QSL|eQSL|LoTW** tells Logger32 to flag your QSOs for QSLing as you log or edit them in your log. When you are ready, the flagged QSOs can be exported in ADIF format using **File ⇨ Export files ▶**



Copyright © 2026 Robert C. Furzer K4CY

- **Remove external Export QSL flags:** some external utilities (add-ons) rely on Logger32 automatically exporting new QSOs for further processing, such as uploading to online log and QSL services. They are *supposed* to enable the automated QSO exports only while running, then politely disable them when done. This function lets us disable the exports manually if add-ons rudely exit or crash without having disabling them.



◀ Click to un-tick any unwanted auto-exports, then click <**Apply**> to stop them exporting.

Of course, you *also* have to install, run and configure the add-ons to make use of the exports, such as L32LogSync which uses the ADIF Audit trail file to sign and upload new QSOs to LoTW, QRZ or eQSL.

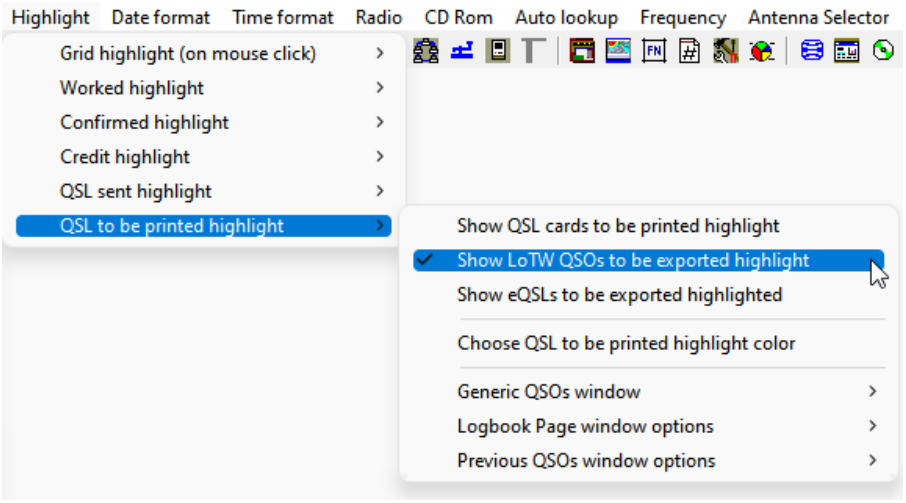
- **Flag existing QSOs for QSLing**: use this function to QSL-flag or un-flag ***ALL*** the QSOs in your logbook at once. Be careful! **Make an ADIF backup of your log *before* using this function**, particularly the risky 'remove ALL flags' options. You are presumably doing this because you screwed up already, QSL-flagging *loads* of QSOs by mistake. Try not to make a bad situation even worse!

Click to tick whichever options you want, then click <**Apply**> to make the changes ▶



Hinson tip: if you mistakenly QSL-flagged just a few QSOs, it is safer to locate and un-flag them individually, manually. One way to find them is with highlighting in the logbook ▼

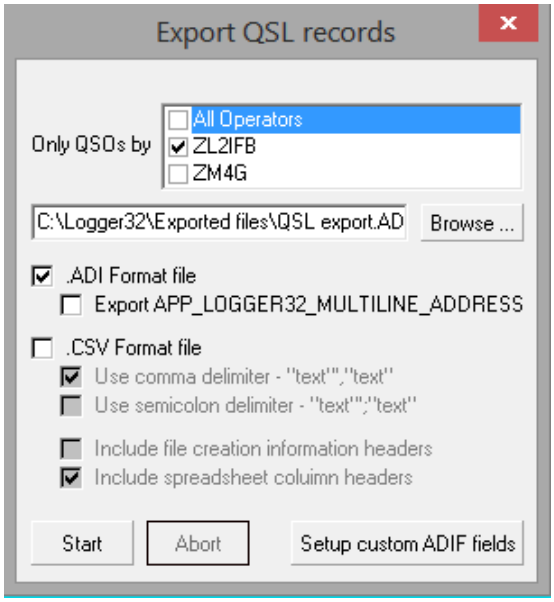
Using **Setup ⇨ Highlight ⇨ QSLs to be printed highlight**, QSOs are highlighted in the logbook if they are marked to send QSLs ▶

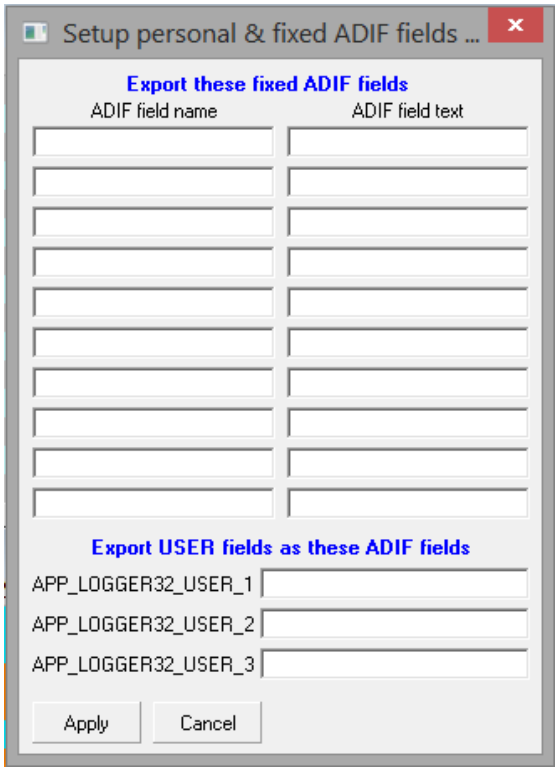


Once found, right-click each flagged and highlighted QSO in the log to check or adjust the associated QSL flags.

Copyright © 2026 Robert C. Furzer K4CY

16.4.2 Export QSL file





1. Use **File** ⇨ **Export files** ⇨ **Export QSL file** to specify which QSL records are to be exported[283], in what format, and to where[284] ▶
2. Select at least one operator. Only QSOs made and logged by that or those operators (callsigns), and flagged <**Send QSL**>, will be exported. Any others are skipped.
3. Click <**Start**> to proceed in a forwardly direction.
4. Check and if necessary amend the drive, folder and name for the output file. Click <**Browse …**> to open File **E**xplorer then navigate to a suitable folder for the file (*e.g. C:\Logger32\Export files*), and specify the file name.
5. Decide whether you want to generate an ADIF-formatted *.ADI* file (appropriate for most amateur radio apps including LogPrint and online QSL printing services) or a *.CSV* data file (suitable for spreadsheets and mail-merging into labels).
6. Select whichever output file options you require.
7. If, for some reason, you want the output file to include particular ADIF fields (such as "MY_SOTA_REF") populated with specific (fixed) data values, click <**Setup custom ADIF fields**> then enter or amend the relevant information on the form and click <**Apply**> ▶

   Likewise, Logger32 can export data saved in the USER fields as APP_LOGGER32_USER_1|2|3 or use different ADIF field names[285].
8. Click <**Start**> to set the export function going. If you change your mind, click <**Abort**> to stop the export in its tracks. Meanwhile, Logger32 dutifully scans the log for QSOs made by the designated operator/s and flagged "Send QSL", and generates ADIF or CSV data records for them.

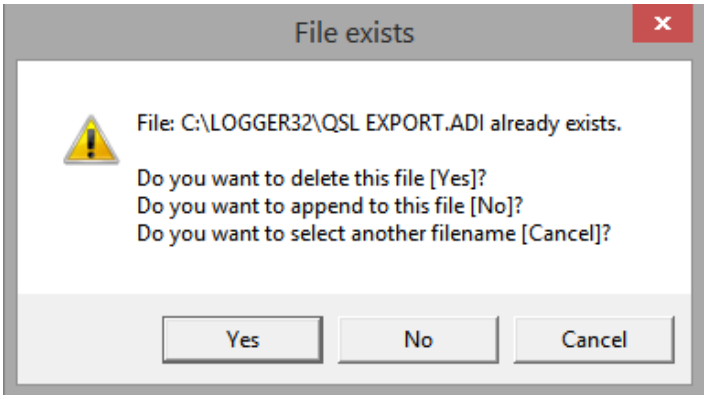
[283] If there are presently no QSOs in your log flagged with <**Send QSL**>, Logger32 informs you and aborts the export routine since it has no data to export. You need to right-click and flag (mark) at least one QSO in your log with <**Send QSL**> in order to use this function.

[284] While any of the export QSL functions are running, Logger32 temporarily suspends processing of DX cluster spots, so after a while you may notice your BandMaps emptying of spots. When you've finished exporting QSL info, the spots come through again in a rush, having been buffered.

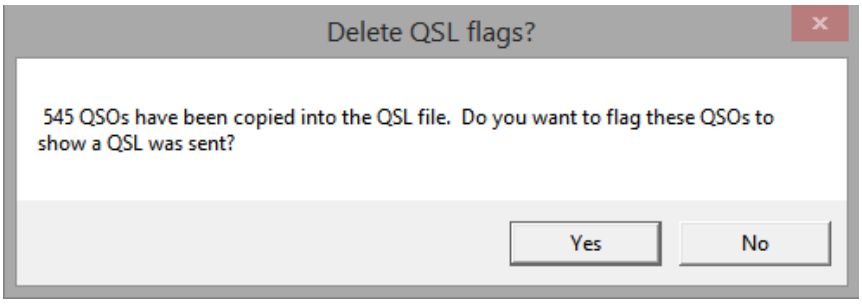
[285] This may be handy to start recording a new type of QSO information before the ADIF standard is updated for it. Once the ADIF update happens, export the saved data from your log with the new ADIF data field.

Copyright © 2026 Robert C. Furzer K4CY

9. If the chosen output file does *not* already exist, Logger32 simply creates it without further ado. If it *does* exist, Logger32 asks whether to **delete** (and replace) it, or open and **append** new QSOs to the end of it, or **cancel** the export (so you can choose a different file name). Go on then, be brave, click the appropriate button ▶



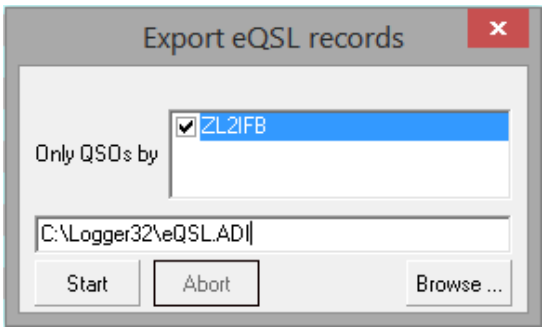
10. Having extracted and saved the QSO data, Logger32 kindly tells you how many QSO records it exported ▶ and gives you one final option to flag those QSOs in your log with "QSL sent" (click <**Yes**>), or not (you guessed it: click <**No**>).



11. Finally, do whatever you intended to do with the exported data. Send the ADIF to your online QSL printer or QSO mapping service. Import the CSV into a spreadsheet or database app to filter, sort and analyze it maybe. Run a mail-merge in a word processing app to populate a label template with the QSO data, then print sticky labels for your QSL cards. Carve wooden or stone plaques. Cast molten metal into molds … or whatever.

16.4.3 Export eQSL file for uploading

This function resembles **Export QSL file** but is much simpler, with hardly any user options to confuse us ▶



To use it:

1. Launch the function using **File ⇨ Export files ⇨ Export eQSL file**.
2. Tick the operator whose QSOs you intend to upload to eQSL.
3. Check and if necessary rename and/or move the data file. If you don't like the look of the suggested path and file name (which is either a default location, or the one you used last time you ran this function), edit the information shown or click <**Browse …**> to look through your disks and folders for something more suitable.
4. Click <**Start**> to generate the data file.
5. Read the messages and follow the prompts: it's not hard!
6. Login and upload the file to eQSL, and you're done.

Copyright © 2026 Robert C. Furzer K4CY

16.4.4 Import confirmations from eQSL

Browse to www.eqsl.cc/QSLCard/DownloadADIF.cfm to download your eQSL confirmations as an ADIF (*.ADI*) file.



Your ADIF log file has been built

There were 78 records

Click one of the following to download it to your computer:

- .ADI file
- .TXT file

Notes:

1. WARNING: Be careful using this file to overwrite an existing log!! It is NOT an exact duplicate of what you originally uploaded!!!
2. It will contain ONLY the following tags: CALL, QSO_DATE, TIME_ON, BAND, MODE, RST_SENT, RST_RCVD, SAT_MODE, PROP_MODE, QSL_SENT, QSLMSG
3. If your comments originally arrived in the QSL_COMMENT tag, they will now be found in QSLMSG instead
4. Since the online logbook at eQSL did not store RST_Rcvd until recently, this file might also NOT contain RST_Rcvd!!
5. If your browser does not pop up a dialog box that allows you to specify the directory and filename where you want the file stored, you may need to use the FILE - SAVE AS function from your browser to save it, or use Cut and Paste to save the file into Notepad or Wordpad.

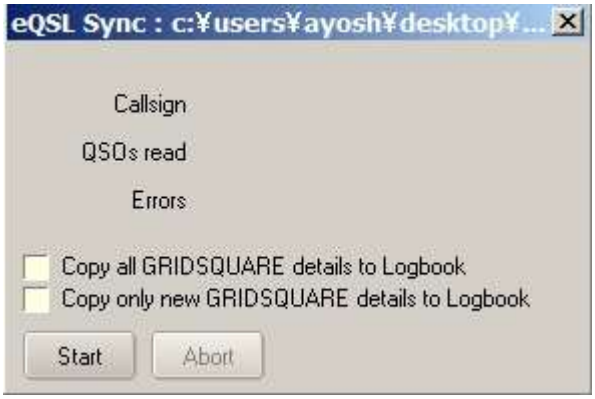
The file will look roughly like this:

```
ADIF 3 Export from eQSL.cc
For JA1NLX
Generated on Thursday, July 20, 2017 at 22:35:42 PM UTC
<PROGRAMID:20>eQSL.cc DownloadADIF
<ADIF_Ver:5>3.0.4
<EOH>
<CALL:4>NX5M<QSO_DATE:8:D>19981212<TIME_ON:4>0025<BAND:3>10m
<MODE:2>CW<RST_SENT:3>599<QSL_SENT:1>Y<QSL_SENT_VIA:1>E
<QSLMSG:7>Thanks!<EOR>
<CALL:5>F6IRG<QSO_DATE:8:D>19991107<TIME_ON:4>0841<BAND:3>10m
<MODE:3>PSK<SUBMODE:5>PSK31<RST_SENT:3>599<QSL_SENT:1>Y
<QSL_SENT_VIA:1>E<QSLMSG:7>Thanks!<EOR>
```

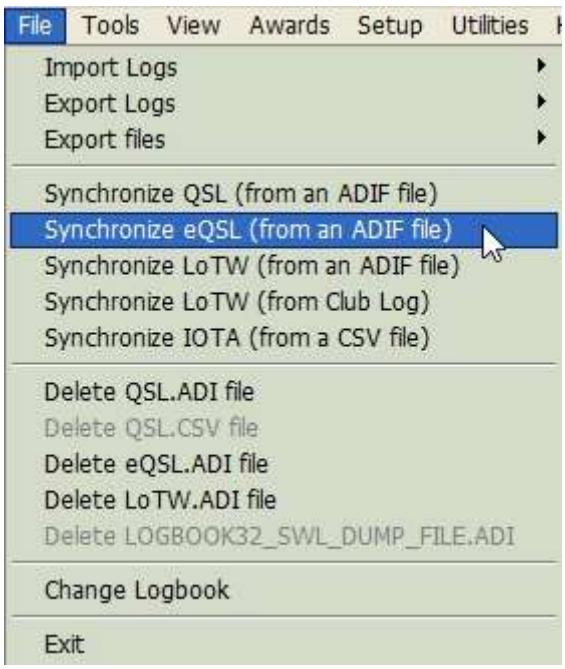
Copyright © 2026 Robert C. Furzer K4CY

Now open **File ⇨ Synchronize eQSL (from an ADIF file)** and select the downloaded file ▶

When the file has been loaded, the eQSL Sync form is displayed ▼



Click <**Start**> and it's off!



The eQSL ADIF file *must* have a <PROGRAMID:20> field in the header or it will be rejected.

Logger32 checks the file for obvious issues during synchronization, putting any invalid QSOs into the *BAD.ADI* file for you to review, and perhaps correct then reload.

The eQSL sync function searches for QSOs in the logbook that match the downloaded confirmations in respect of the callsigns, bands, modes and dates & times. To allow for clock variations, it allows +/- 30 minutes leeway on the <TIME_ON> field – the times when each QSO started. If a QSO with the same callsign, band and mode is found in your log at about the right time, and if the eQSL_SENT field is Y, a match has been found so the eQSL_RCVD field on that QSO is set to Y.

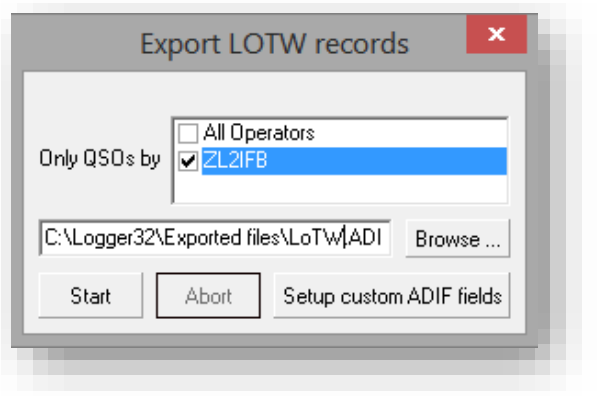
Hinson tip: internally (under the covers), Logger32 uses custom ADIF field labels eQSL_SENT and eQSL_RCVD field labels that were chosen before the ADIF standard catered for them. However, Logger32 uses the correct ADIF-compliant field names when exporting QSOs to ADIF files.

"eQSL has processed 265,720,812 QSOs from Logger32 users. 113 eQSL users gave Logger32 a 👍 and 1 gave it a 👎. In terms of volume of QSOs sent to eQSL the #2 ranked logging program sent 171,663,896 QSOs, just over half as many as Logger32. Looks like we might just be doing something right!"

Bob K4CY

Copyright © 2026 Robert C. Furzer K4CY

16.4.5 Export LoTW file



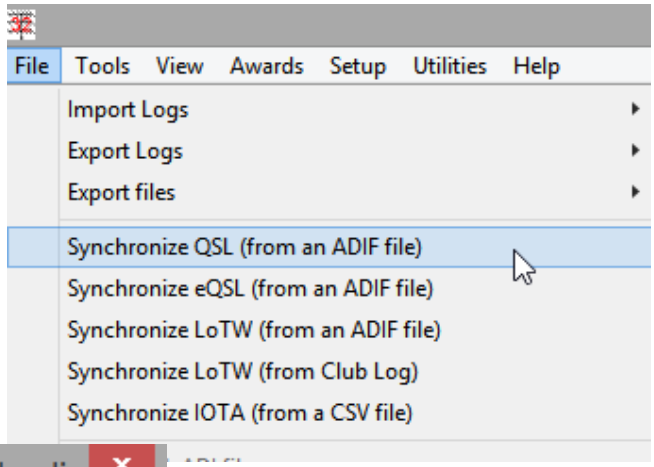
Here's another straightforward function, laying somewhere between **Export QSL file** and **Export eQSL file** in terms of complexity ▶

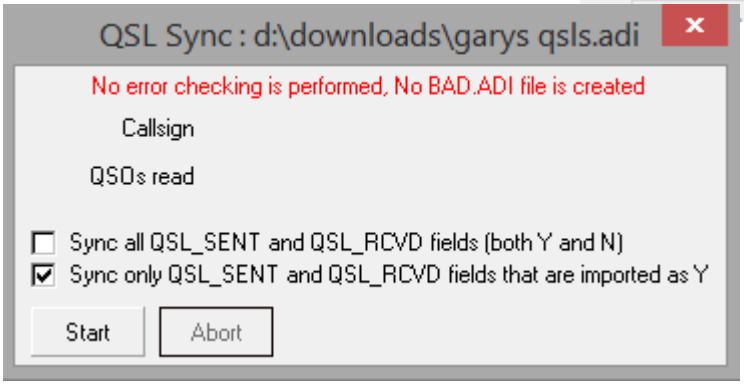
1. Launch the function using **Files ⇨ Export Files ⇨ Export LoTW file**.
2. Select (tick) the operator whose QSOs you intend to upload to ARRL's **L**ogbook **o**f **T**he **W**orld.
3. Check and if necessary rename and/or move the exported data file to a different folder.
4. Click <**Start**> to generate the data file.
5. Read the messages and follow the prompts: it's easy really!
6. Sign and upload the file to LoTW using TQSL, and you're done.

16.5 Synchronize QSL (from an ADIF file)

This function was designed for those who employ the services of a QSL manager. Keep your QSL manager updated by periodically:

1. Generating an ADIF log file export containing just your recent QSOs (made since the previous time you updated your QSL manager).
2. Sending the ADIF to your QSL manager *e.g.* by email, Dropbox *etc*.
3. Waiting until the QSL manager updates his/her copy of your log, and optionally sends you back an ADIF file containing just the QSOs that have been confirmed or corrected (*e.g.* as a result of QSL cards or emails received).
4. Importing the ADIF into your log to reflect the QSLs sent and received by your QSL manager using **File ⇨ Synchronize QSL (from an ADIF file)** ▶





◀ The Sync QSL function does not check for errors: use at your own risk!

Backups are your friends.

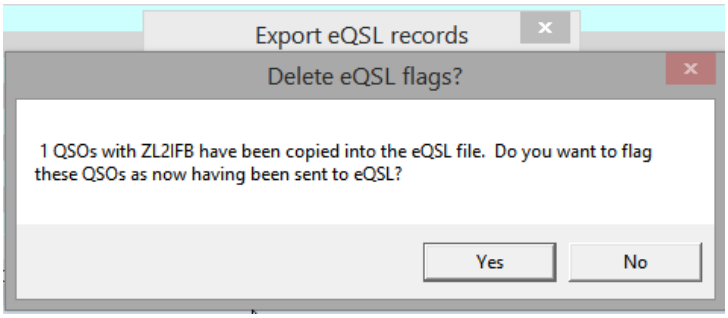
Backup early:
later may be *too* late.

Copyright © 2026 Robert C. Furzer K4CY

16.6 QSLing FAQs

Q. I exported a QSL file and sent it to the QSL printer. How do I now mark those QSOs to show I have sent the QSLs?

A. There is an important question at the end of the QSL export process ▼



The default answer[286] (whether you click it or simply press <**Enter**>) is <**Yes**>, meaning "Go ahead, clear the *send QSL* flag on those QSOs that I have just exported, set the *QSL sent* flag on them instead, and record today's date under the *QSL sent date*."

If instead you clicked the <**No**> button, the flags and sent date are *not* updated … in which case, if you run the same QSL export function again (soon, without logging any new QSOs in between[287]), the same QSOs will be selected for QSLing from your log and exported since the flags remain the same. However, if you click <**Yes**> to the final question this time around, the flags *will* be updated. That answers your question.

You may wonder why Logger32 even asks this question: why would anyone want to export the QSL info but *not* update the flags accordingly?

The answer is that:

- You have only generated the QSL info export file at this point: until/unless you complete the remaining steps in the QSLing process (*e.g.* uploading the QSOs to an online log-matching/awards application or QSL printing system), it is not quite complete.
- You *may* want to wait until the entire process is completed successfully before updating those flags, so that if you need to run the export again for some reason (*e.g.* if you lost the export file, forgot to upload it or if the upload and import steps failed), you can do so easily … whereas if you had already updated the flags, you would need to go through your log manually finding and flagging the same QSOs to be QSL'd again, assuming you can even identify the QSOs in question. Trust me, that is tedious and error-prone.
- Provided the exported QSL file is still available (not carelessly deleted), you ought to be able to [sign and] upload it again. Hopefully *this* time it will go through OK, so then you can run the QSL export again to update the flags and dates, then get back to the serious business of DXing.

[286] Notice the heavier shadow around the <**Yes**> button? That indicates the default response.

[287] If you logged and send-QSL-flagged any new QSOs since the first export, they would be exported in the second export and then marked as sent when you answer <**Yes**> to that update flagging question: if you only intended to update the QSL flags for the QSOs initially exported, you will now have to manually re-flag the new QSOs to send QSLs in the *next* run. It's easier to update the flags *before* you log any new QSOs!

Copyright © 2026 Robert C. Furzer K4CY

Q. I want to capture multiple consecutive QSO's so I can flag them for export. The typical Windows method of selecting a QSO in my log then shift-clicking another QSO doesn't seem to work. So how to I flag multiple QSOs?

A. Flagging consecutive QSOs in your log is not possible due to the risk of accidentally damaging the log. Logger32 works with just one QSO at a time. This is a conscious design decision.

Instead, try this: right-click any data entry field in the Log entry pane. Click **Setup ⇨ QSLing & QSO export ⇨ Flag existing QSOs for QSLing ▼**

Anything there that you could use? If not, you'll either have to right-click and QSL-flag each QSO in your log individually, or:

1. Export your log as an ADIF file from Logger32 – either all of it or just a partial export containing the QSOs of interest;
2. *Carefully* edit the ADIF to flag multiple QSOs *e.g.* using ADIFmaster or a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad. Check it and save it;
3. Import the edited ADIF into a new Logger32 log (if you exported the whole log) or into the same log having first made a safety backup (if you only exported and modified a section of the log).

Q. I'm about to send a direct QSL card to a DX station's QSL manager. I'm pretty sure the QSL manager manages other DX stations too. How can I tell which other DX QSOs might be worth QSLing in the same batch, to save postage?

A. If you routinely log QSL managers' callsigns, you can display that column in the logbook, then click the column heading to sort the logbook by that field.

Copyright © 2026 Robert C. Furzer K4CY

17 LoTW (Logbook of The World)

> "Imagine being able to submit evidence of a contact electronically, and have it count towards awards. No muss, no fuss - just a simple procedure under a system that ensures the validity of the QSO."
>
> *Wayne Mills N7NG*

ARRL's **L**ogbook **o**f **T**he **W**orld online QSO matching and confirmation system makes confirming contacts easier, quicker, cheaper and more trustworthy than traditional QSL methods, and is especially valuable for those of us chasing DXCC and a few other awards.

In essence, LoTW users upload our QSO records to the LoTW system which adds them to a large QSO database. If our QSO partners also upload the corresponding QSO records, LoTW matches them up and confirms the QSO for both of us, ready to be claimed and credited for the awards.

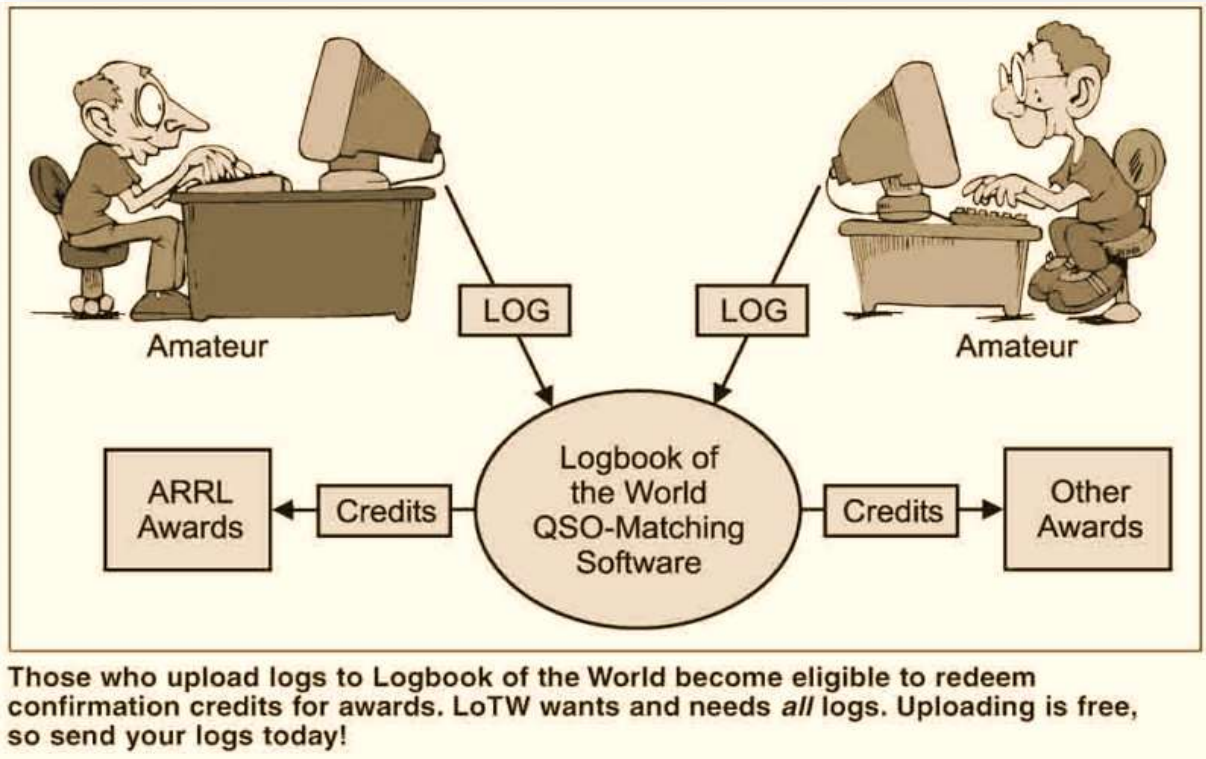


Diagram courtesy of Wayne Mills, N7NG
"Introducing Logbook of The World" in QST, October 2003.

Copyright © 2026 Robert C. Furzer K4CY

Read more about LoTW on the ARRL LoTW website and in the LoTW New User Guide.

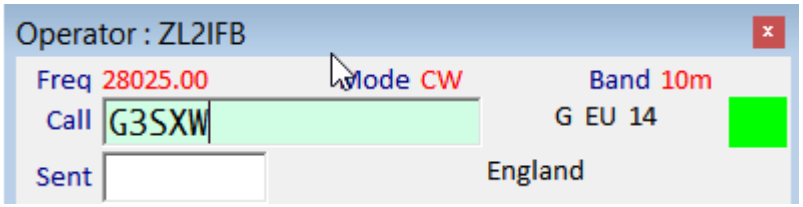
Logger32 supports the use of LoTW, providing program functions to:

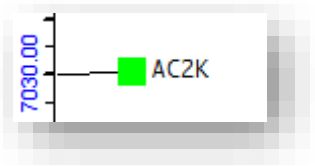
- Export QSO records to be signed and uploaded to LoTW through ARRL's TQSL program.
- Import LoTW confirmations and update the logbook to show that certain QSOs have been confirmed, and are thus eligible for the DXCC and other awards.
- Maintain details of DXCC and other awards showing which entities have been both contacted and confirmed.
- Generate ADIF files of QSOs that have been confirmed on QSL cards but *not* on LoTW, for use in online DXCC applications to supplement those confirmed on LoTW.
- Import details of confirmed QSOs that have been checked, approved and granted towards DXCC and the [few] other awards which accept LoTW confirmations.

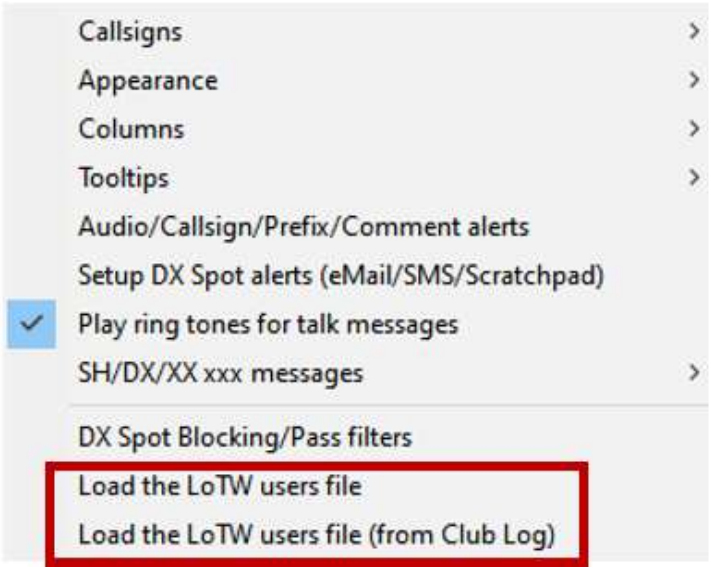
17.1 Identifying LoTW users

17.1.1 Load the LoTW user file

Since LoTW makes it so easy and cheap to confirm QSOs for awards such as DXCC, some hams preferentially contact other LoTW users. Logger32 can automatically identify which stations are known to use LoTW whenever they are spotted on DX cluster, decoded on digital modes, or entered into the log entry pane as you are working them. It shows distinctive square blobs ▼ highlighting the relevant stations.







For this purpose, Logger32 lets you download a data file containing callsigns and (optionally) the dates when those stations last uploaded to LoTW[288].

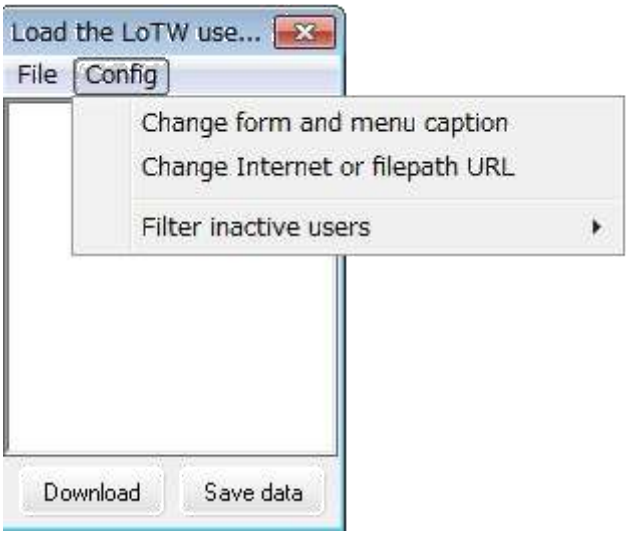
Logger32 defaults to the LoTW user file containing just the callsigns of LoTW users. In fact, as we'll see in just a moment, the data file can contain *any* list of callsigns.

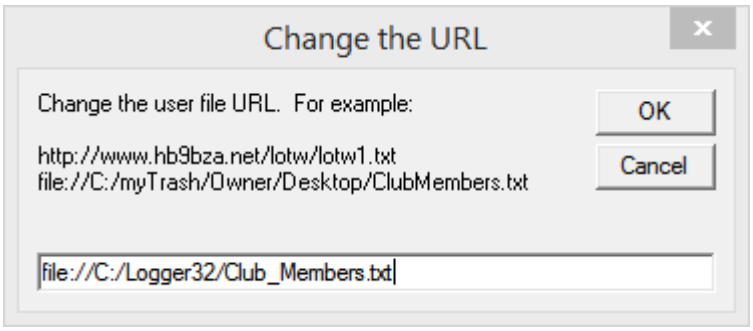
◀ To download a simple list of LoTW users *without* their last upload dates, right-click the DX Spots pane then click **Setup ⇨ Load the LoTW users file**.

[288] The last upload dates are a big clue as to whether they are still actively using LoTW or may have long since given up/forgotten how to do it.

Copyright © 2026 Robert C. Furzer K4CY

If you wish, the <**Config**> menu lets you choose a *different* data file using <**Change Internet or filepath URL**>[289] ▶

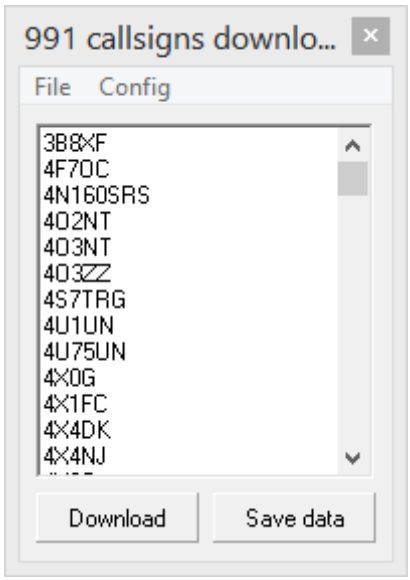




◀ The URI naming format - especially for files on your computer - can be tricky, so look closely and follow the format of the examples given in the window.

Click <**OK**> when you're ready.

To import the list of LoTW users from the online or local data file, click <**Download**> ▶



Callsigns stream through the window as the data file is imported to a temporary cache. When the streaming ends, click <**Save data**> to commit the callsign list to Logger32's database ▶

Hinson tip: if you close the form prematurely *without* clicking <**Save data**>, the downloaded data are simply discarded … and you'll probably have to start over by downloading once more.

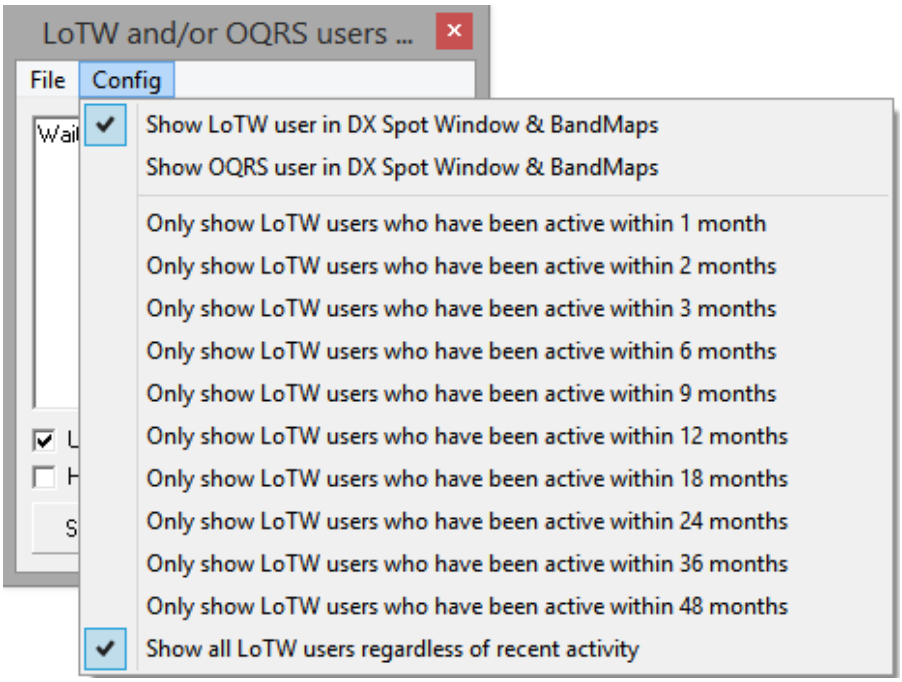
17.1.2 Load the LoTW user file *from Club Log*

Through a special *weekly* hook-up to the LoTW system, Club Log maintains a list of LoTW users *with* the dates of their most recent LoTW uploads and (for many stations) their grid squares. It also knows when Club Log users last uploaded their logs to Club Log. Recent log uploads suggest those hams are actively making and confirming QSOs.

[289] If you are not that bothered about LoTW and DXCC, you might prefer a list of "Address Guaranteed" eQSL users instead: use the URL http://www.eqsl.cc/qslcard/DownloadedFiles/AGMemberList.txt Lists of club members' callsigns work in the same way *e.g.* FOC members. Only one user database can be active in Logger32 at a given time though, so choose wisely.

Copyright © 2026 Robert C. Furzer K4CY

To download the LoTW users file from Club Log into Logger32, right-click the DX Spots pane then click **Setup ⇨ Load LoTW users file (from Club Log).**



The Club Log data file contains not only callsigns but LoTW upload dates, recent Club Log upload dates, OQRS status (user or not) *and* their grid squares. **Club Log updates this file once a week.**

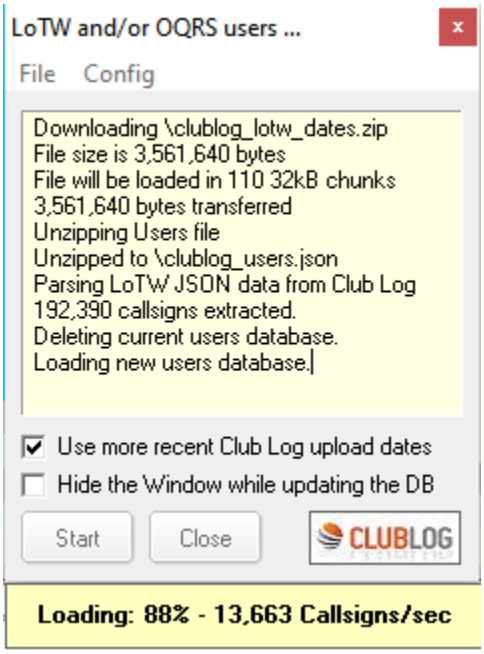
Logger32 shows whether each station is an LoTW and/or OQRS user on the DX Spots pane, the BandMaps and the log entry pane.

▲ If you wish, Logger32 can show the colored blobs only for users who have uploaded to LoTW or Club Log's **O**nline **Q**SL **R**equest **S**ervice within the past 4 years or less, *implying* that they are actively uploading.

<**Use more recent Club Log upload dates**> overrides the LoTW upload date if they have since uploaded their logs to Club Log ▶

Since it takes a while to request, extract, download, unzip, parse and load the data file, so you can <**Hide the Window while updating the DB**> to let the last phase happen in the background … ▶

… since status messages near the system clock still show the progress and rate anyway ▶



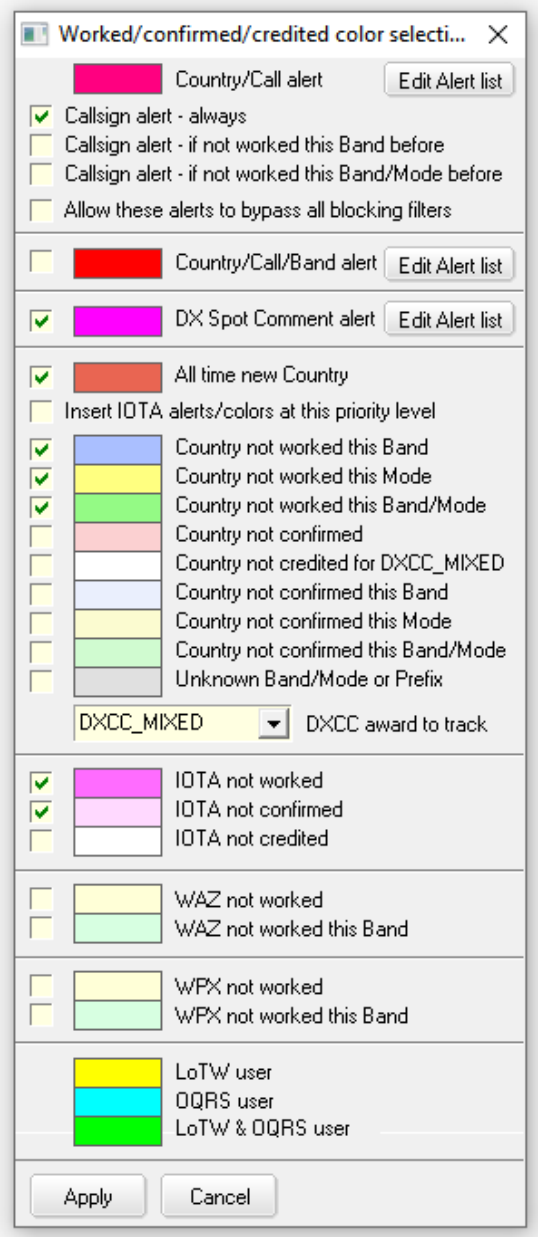
Click <**Start**> to download the file, and if you didn't hide the window while updating the database, click <**Close**> when it tells you it is done.

The download is saved as *C:\Logger32\clublog_lotw_dates.zip* and the unpacked JSON file is *C:\Logger32\clublog_users.json*, in case you want to take a look at the raw data using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad.

Copyright © 2026 Robert C. Furzer K4CY

17.1.3 LoTW & Club Log OQRS user blobs and tooltips

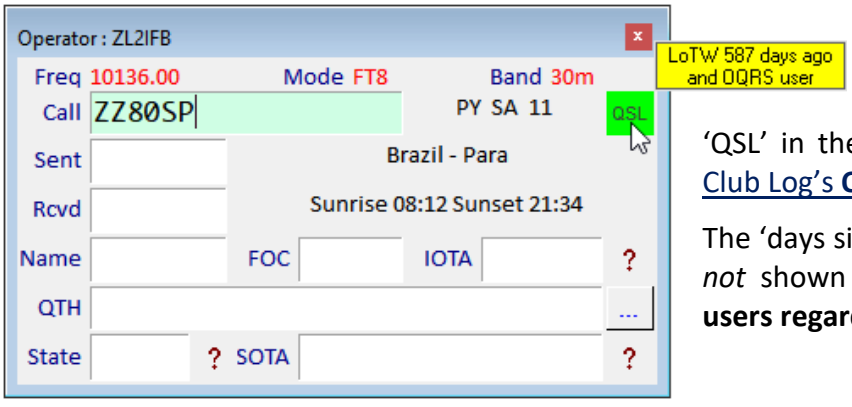
Blobs indicating stations known to use **L**ogbook **o**f **T**he **W**orld and/or Club Log's **O**nline **Q**SL **R**equest **S**ystem on the log entry, DX Cluster and DX Spot panes and on the BandMaps can be colored to your liking, using possibly *the* most obscure and hard to locate color configuration setting within Logger32 – at least *I* struggle to find it.



To open this menu,
right-click the DX Spots pane,
then click **Setup ⇨ Appearance ⇨ DX Spot highlight colors**.

Down here at the bottom are three rectangles to color spots for stations that use LoTW, OQRS or both ▶
Simply click a rectangle then pick your crayon.

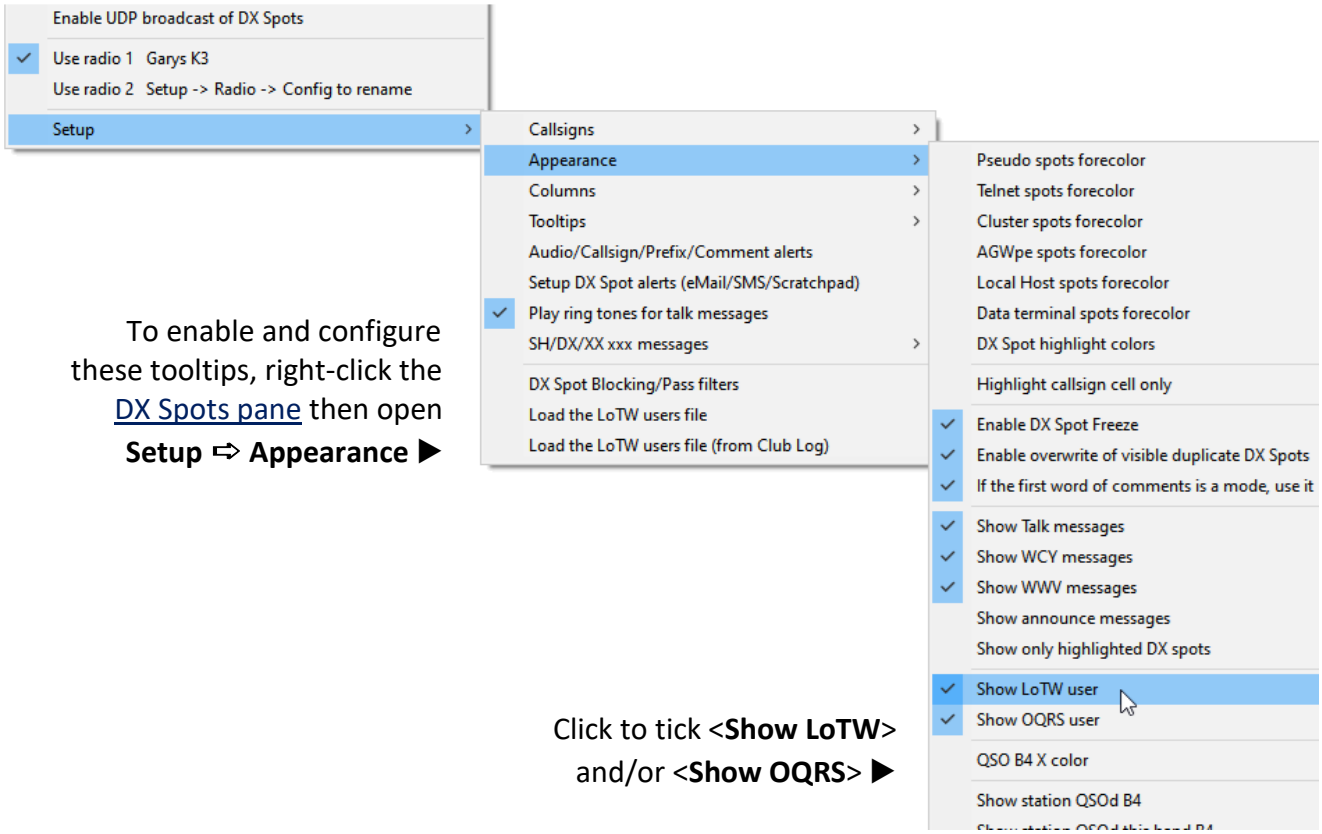
Configurable tooltips showing the last LoTW upload data and whether the station uses OQRS can be displayed when we mouseover the colored blobs on the DX Spots pane or BandMaps, and in the log entry pane. Here is a typical log entry pane colored blob tooltip with *both* <**Show LoTW users**> *and* <**Show OQRS users**> selected ▼



'QSL' in the blob tells us the station uses Club Log's **O**nline **Q**SL **R**equest **S**ervice.

The 'days since LoTW was updated' value is *not* shown if you select <**Show all LoTW users regardless of recent activity**>.

Copyright © 2026 Robert C. Furzer K4CY



To enable and configure these tooltips, right-click the DX Spots pane then open **Setup ⇨ Appearance ▶**

Click to tick <**Show LoTW**> and/or <**Show OQRS**> ▶

17.1.4 Grid squares for stations spotted on DX cluster

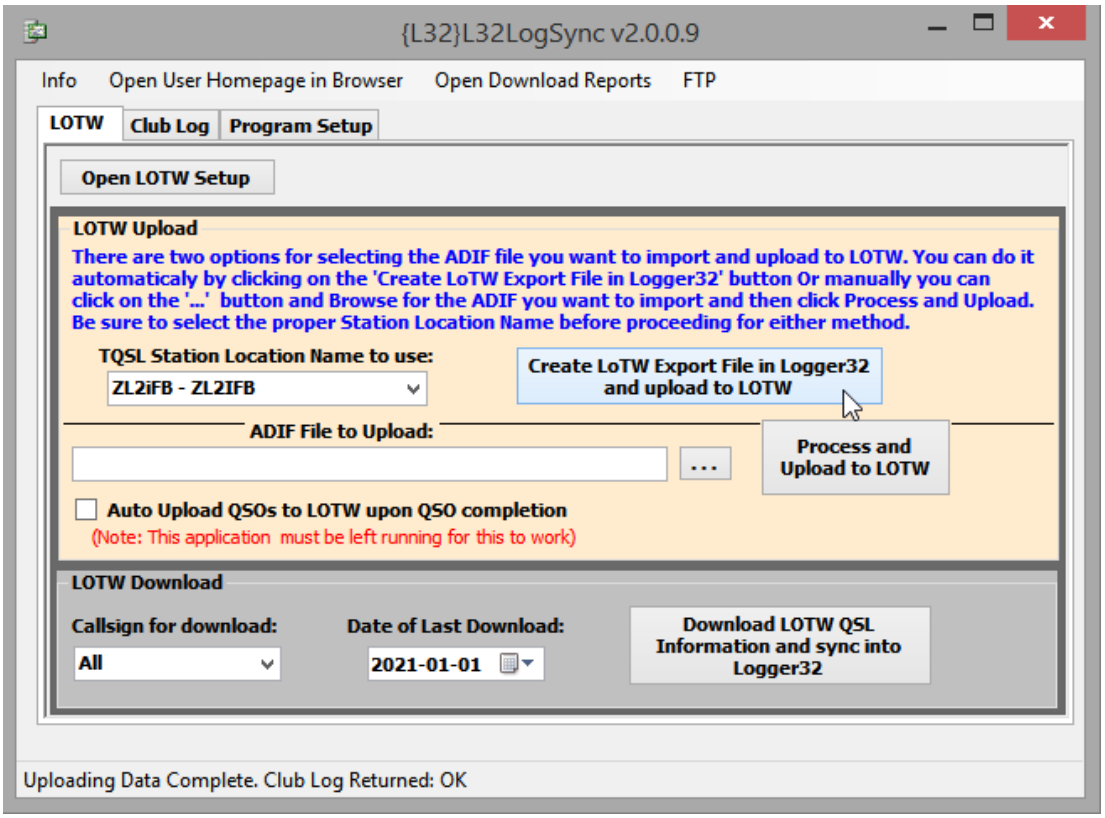
If Club Log knows a DX station's grid square, and if you have downloaded that information from Club Log as just explained, when you click a DX spot for that DX station, his grid square automatically populates the log entry pane's Grid field (if shown) *provided* Logger32 has not already determined the grid square by some other more reliable or at least current means (*e.g.* if he sent his grid in CQ messages on FT#).

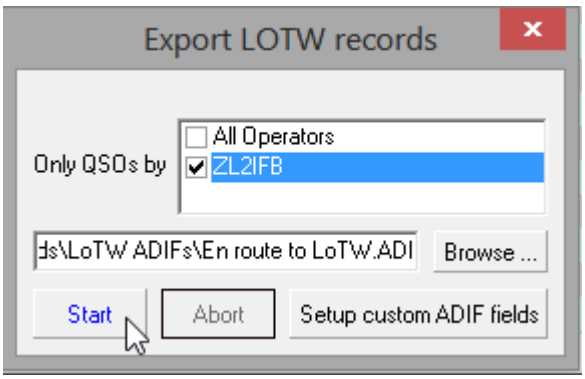
17.2 Upload your QSOs to LoTW

The process of uploading QSOs from your logbook to LoTW is covered in the QSLing chapter. In summary, flagged QSOs are exported as an ADIF file that gets signed and uploaded through the TQSL program. The 'Send to LoTW' flag is reset (un-set) so that the same QSOs are not exported, signed and uploaded again the *next* time you run the process … unless you change any of the basic QSO information (*i.e.* the date/time, callsign, band/frequency or mode) for a logged QSO, which magically sets the 'Send to LoTW' flag once more.

Copyright © 2026 Robert C. Furzer K4CY

The L32 LogSync utility automates the uploading process, making it easy to update LoTW as often as you like. Here I am about to click the <**Create LoTW Export File in Logger32 and upload to LOTW**> button ▼





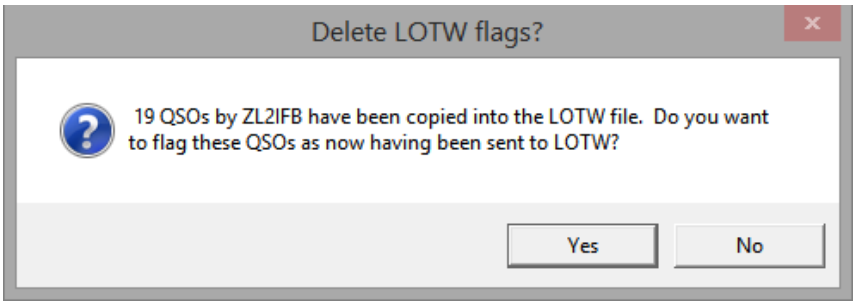
◀ I'm prompted to choose whether to upload QSOs made with my regular and/or other callsigns (if any) … but having just selected the <**TQSL Station Location Name to use**> (and hence the callsign certificate in TQSL) for my ZL2IFB callsign, I *must* choose the same callsign here.

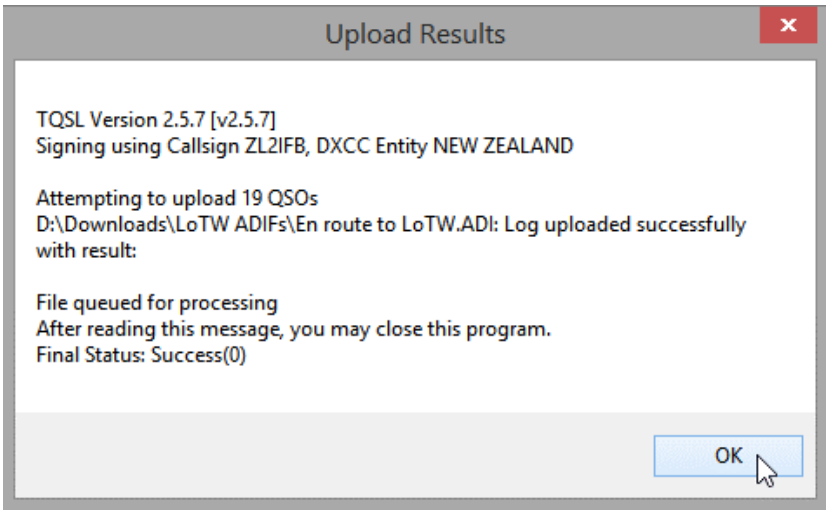
◀ I could pick a different folder and/or file name … but it's easiest to let the system re-use the same one each time: I just click <**Start**>, quietly muttering to myself "Oh get on with it!".

Hinson tip: for system security reasons, if you run Logger32 as administrator, Windows insists that you also run any programs it calls (such as L32 LogSync and hence TQSL) as administrator too. Conversely, if you do not run Logger32 as administrator, it cannot call any programs that run as administrator.

Copyright © 2026 Robert C. Furzer K4CY

Immediately, L32 LogSync asks me whether to flag the QSOs with LOTW_SENT, so naturally I click <**Yes**>, now gnashing my teeth[290] ▶





◀ Barely a second later, TQSL notifies me that the QSOs have been signed and uploaded successfully.

Job done! Well nearly …

◀ Just the OK button click remains! We're some way short of a one-click process.

L32 LogSync also automates downloading and updating your logbook with LoTW confirmations through the gray section on the same window. Read on for more …

17.3 Download LoTW confirmations to update your log

LoTW can generate and export ADIF files listing our QSOs that have been matched and hence confirmed in LoTW, and may also have been claimed and credited towards various DXCC awards.

Hinson tip: make a backup of your log *before* updating it. In the unlikely event the update doesn't go entirely to plan, you'll be ~~glad~~ relieved you did this.

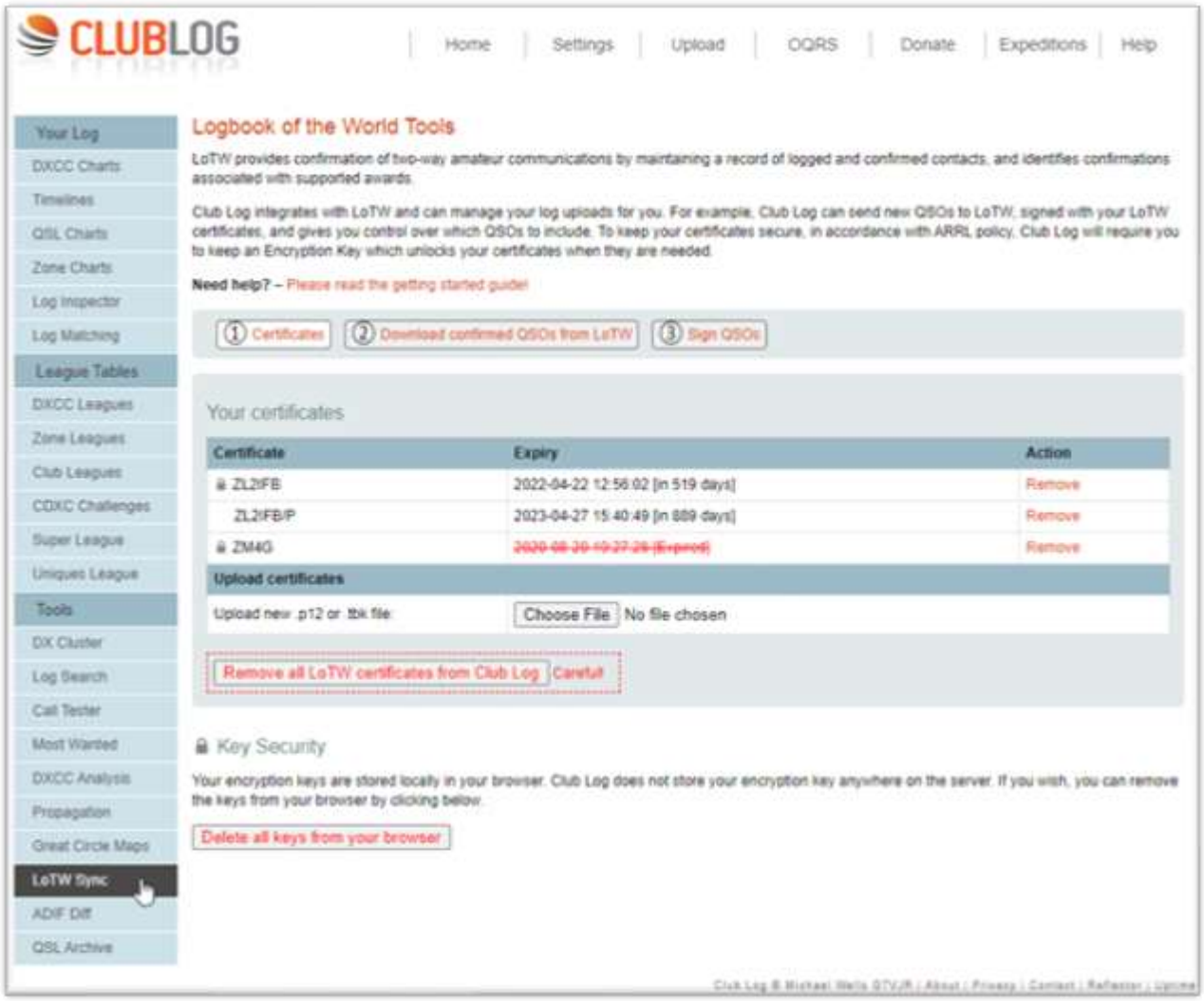
There are *four* ways to generate and download the ADIF file from LoTW. In descending order of popularity, check out the following notes on using Club Log, L32LogSync, manual LoTW downloads or (last resort!) the ARRL Scraper.

[290] At this precise point in the process, the QSOs have *not* yet been signed and uploaded to LoTW, in fact: they are pending. L32LogSync is a little premature in asking to update the flags since the signing and uploading step *may* not complete as planned *e.g.* if my Internet connection or the LoTW server is down. L32LogSync should really wait until *after* the successful upload message from TQSL that follows before updating the flags – and it wouldn't need to ask for my permission, saving a click. But, hey, it's only a hobby, right?

Copyright © 2026 Robert C. Furzer K4CY

17.3.1 LoTW and DXCC synchronization *via* Club Log

If you are registered for both LoTW and Club Log, Logger32 can update your logbook to record your DXCC credits painlessly thanks to Club Log's handy <**LoTW Sync**> function that synchronizes your Club Log log with your LoTW account ▼



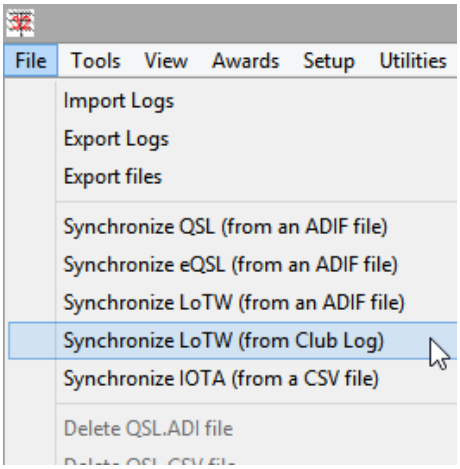
Club Log can generate and export a JSON (**J**ava**S**cript **O**bject **N**otation) data file containing details of your logged QSOs that have been confirmed on LoTW and perhaps granted towards DXCC awards, which Logger32 can download and then use to update the LoTW and DXCC flags in your log accordingly in batches of 10,000 QSOs.

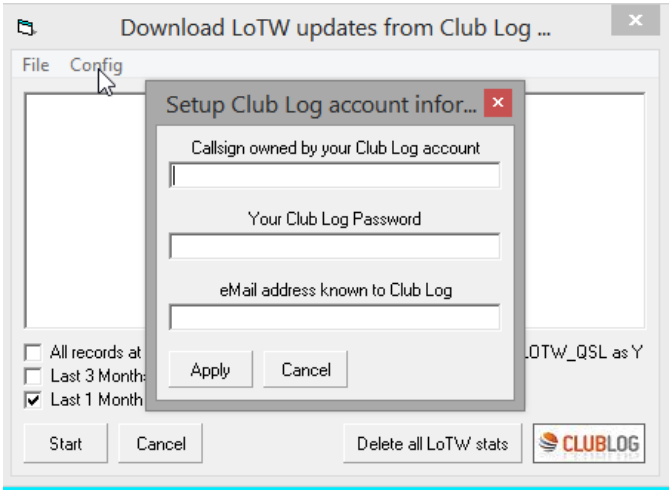
Here is the entire process, step-by-step:

1. Login to Club Log.
2. Unless you routinely upload all your QSOs to Club Log as they are logged using the excellent L32 LogSync add-on, update your log in Club Log by exporting recent QSOs from Logger32 as an ADIF file and uploading that to Club Log. Wait a while for the upload to be processed.
3. Run Club Log's LoTW Sync function to grab all your LoTW confirmations and DXCC credits, updating your log (or logs, if you have logs under your different callsigns) in Club Log. This process also takes a while so either wait for a confirmation email from Club Log (if you have configured Club Log's settings to email you a summary after each upload) or take a break to do something else for maybe an hour or so.

Copyright © 2026 Robert C. Furzer K4CY

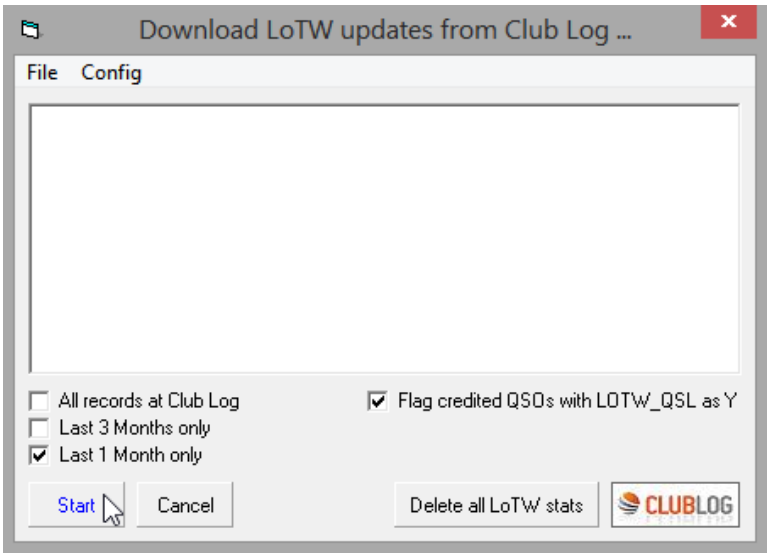
4. In Logger32, click **File ⇨ Synchronize LoTW (from Club Log)** to open the Club Log synchronizer function ▶





◀ Logger32 needs to login to Club Log to get your information, for which it needs your Club Log username and password. Click <**Config**> on the menu, type in your callsign and Club Log login credentials, then click <**Apply**> to save and use them[291].

5. Select the appropriate options at the bottom of the form ▶



- <**All records at Club Log**> fully synchronizes your Logger32 log with your DXCC records at Club Log[292]. Do this the first time you run the synchronizer.
- The <**Delete all LoTW stats**> button wipes all the LoTW flags from your log if they have got into a mess, so you can start afresh by synchronizing using the <**All records …**> option.
- <**Last 3 Months only**> and <**Last 1 Month only**> are useful to update your DXCC status periodically, grabbing the information you have recently updated at Club Log from LoTW. If you remember to do it monthly or at least once a quarter[293], you can avoid having to run the slower <**All records …**> update.

[291] Your Club Log credentials are saved insecurely, so either avoid using this function altogether (*especially* on a shared club or public computer) or login to Club Log and change your password as soon as practicable after doing it.

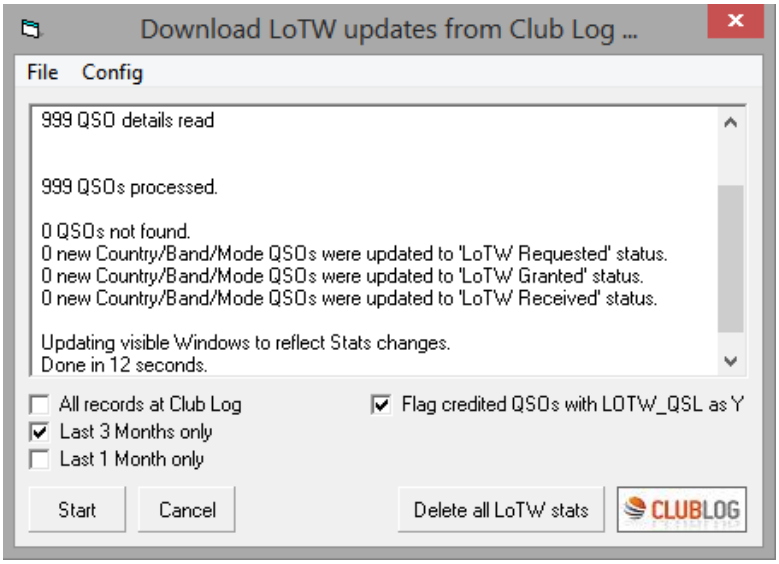
[292] It takes about half an hour to run on my system. Yours may be faster … or not.

[293] Stick a note in your diary!

Copyright © 2026 Robert C. Furzer K4CY

- <**Flag credited QSOs with LOTW_QSL as Y**> marks all QSOs that have been credited for DXCC in your LoTW records, with LOTW_QSL = Y. This includes any QSOs that were actually confirmed on QSL cards, and were credited to your DXCC account.

6. Click <**Start**> when you are ready to roll.

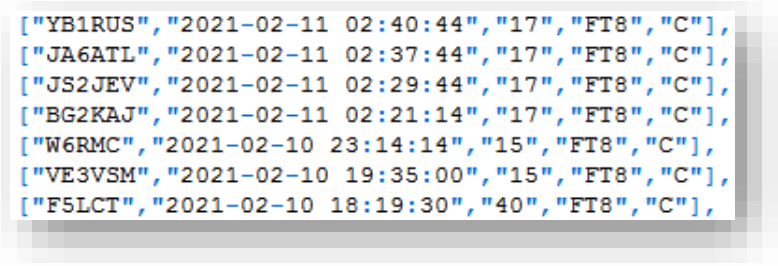


◀ Logger32 tells you what's going on with a scrollable progress window and a yellow pop-up notification down near the system clock ▼



Hinson tip: if you get bored waiting or need to get on with logging QSOs, you can abort the Club Log LoTW sync process and re-start it later on. It remembers where it got up to and picks up from that same point next time.

Behind the screen, Logger32 authenticates and requests the information from Club Log, downloading the data as a JSON-formatted file *C:\Logger32\LoTW records from ClubLog.txt*



```
["YB1RUS","2021-02-11 02:40:44","17","FT8","C"],
["JA6ATL","2021-02-11 02:37:44","17","FT8","C"],
["JS2JEV","2021-02-11 02:29:44","17","FT8","C"],
["BG2KAJ","2021-02-11 02:21:14","17","FT8","C"],
["W6RMC","2021-02-10 23:14:14","15","FT8","C"],
["VE3VSM","2021-02-10 19:35:00","15","FT8","C"],
["F5LCT","2021-02-10 18:19:30","40","FT8","C"],
```

◀ JSON is a simple comma-delimited text format, more succinct but less flexible than ADIF.

Then Logger32 locates the QSOs in your log, updating their LoTW and DXCC status accordingly.

Hinson tip: within the L32 LogSync utility, the <**Download LOTW QSL Information and sync into Logger32**> button does a similar job without involving Club Log. It accesses your LoTW account directly then updates your Logger32 log to record all LoTW confirmations received lately and (if selected) it lets you update the logged counties and zones *etc*. accordingly. However, the L32 LogSync function does not reflect your DXCC account *e.g.* if you have applied to credit confirmed QSOs for DXCC, or if confirmed QSOs have been granted for any of your DXCC awards. It purely reflects your LoTW confirmations for matched QSOs.

While Club Log may know of QSOs that are *eligible* for DXCC credit, it cannot definitively determine *which* particular QSOs have in fact been credited for QSOs dating back to 2017 or before due to limitations of the data provided from the DXCC system via LoTW. For most of us, that doesn't particularly matter: the important thing is to know which DXCC *entities* have been worked, confirmed, claimed and credited for DXCC on a given band and mode.

Copyright © 2026 Robert C. Furzer K4CY

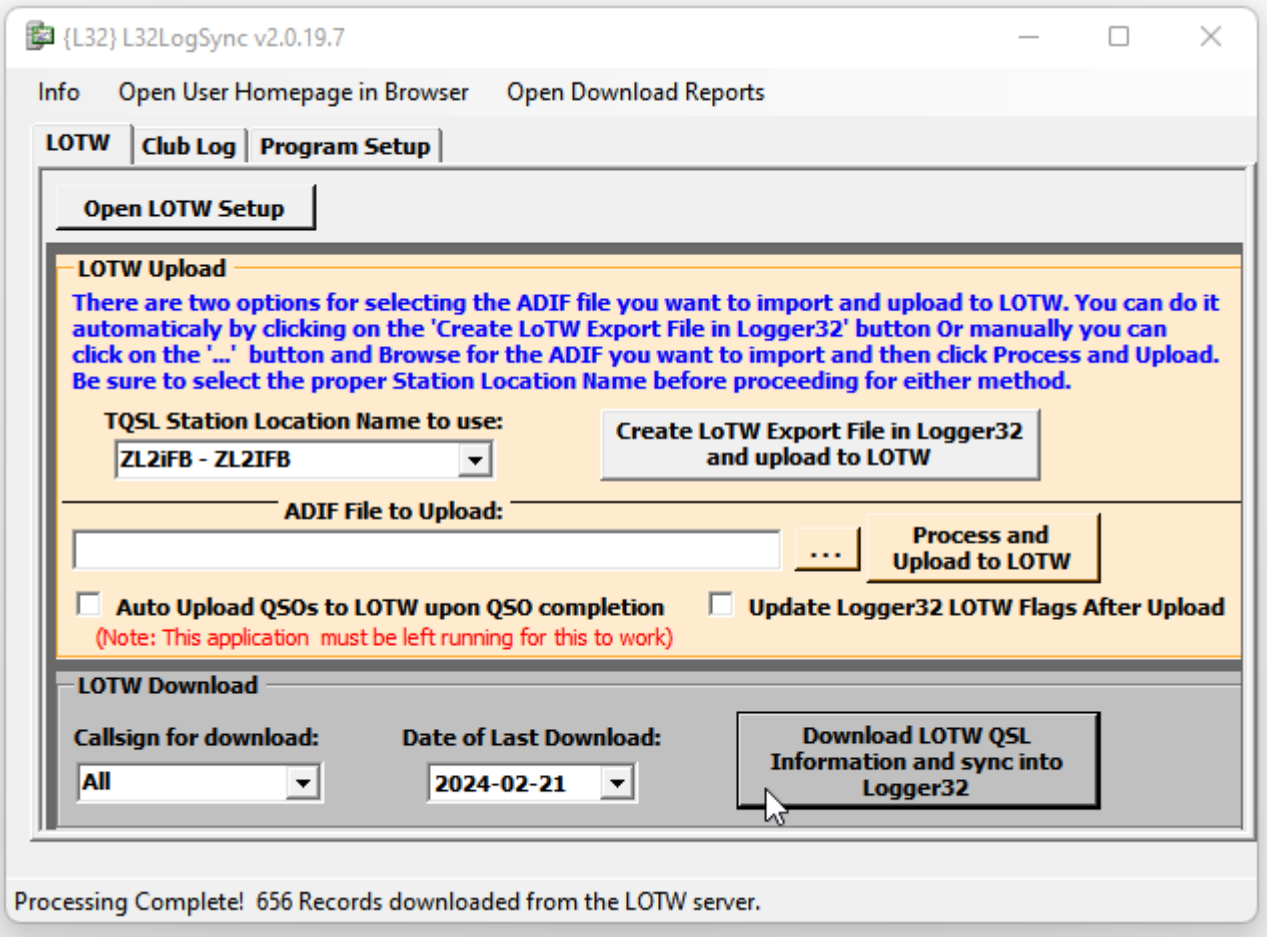
Hinson tip: if for some reason you need more definitive information about the respective QSOs, the <**Account credits**> left-menu option on the LoTW <**Awards**> tab lists the callsigns credited towards your DXCC awards. Click any callsign on the list to see the QSO details – either a short report for QSOs confirmed by QSL cards, or a longer report for QSOs matched in LoTW.

You may see "QSO not found" messages, typically resulting from mismatched modes between your logbook and Club Log/LoTW. For example, you may have logged a QSO on "FT2" in your logbook whereas Club Log/LoTW considers it a "DATA" QSO … in which case FT2 should be configured in your Bands & Modes table, and defined as a digital mode in the **Setup phone/digital mode** table. Logger32 then automatically translates mismatched modes if configured properly.

Hinson tip: jot down details of those "QSOs not found" *before* closing the information window, in order to figure out why they weren't found and hopefully fix your log (which may take some sleuthing). Once the information window is closed, the only way to have Logger32 list those unfound QSOs is to run the process again.

17.3.2 LoTW -> log synchronization using L32 LogSync

A simple way to grab a batch of confirmations from LoTW and update the QSOs in your logbook to show they are confirmed is to use the L32 LogSync add-on ▼



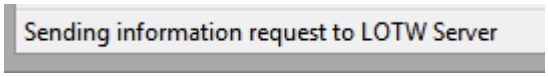
The gray panel has two data-entry fields and a button:

- **Callsign for download**: if you have uploaded logs from multiple callsigns into your LoTW account (*e.g.* your regular and personal contest callsigns), you can download confirmations for the individual callsigns separately or all at once. As part of identifying which QSOs in your

Copyright © 2026 Robert C. Furzer K4CY

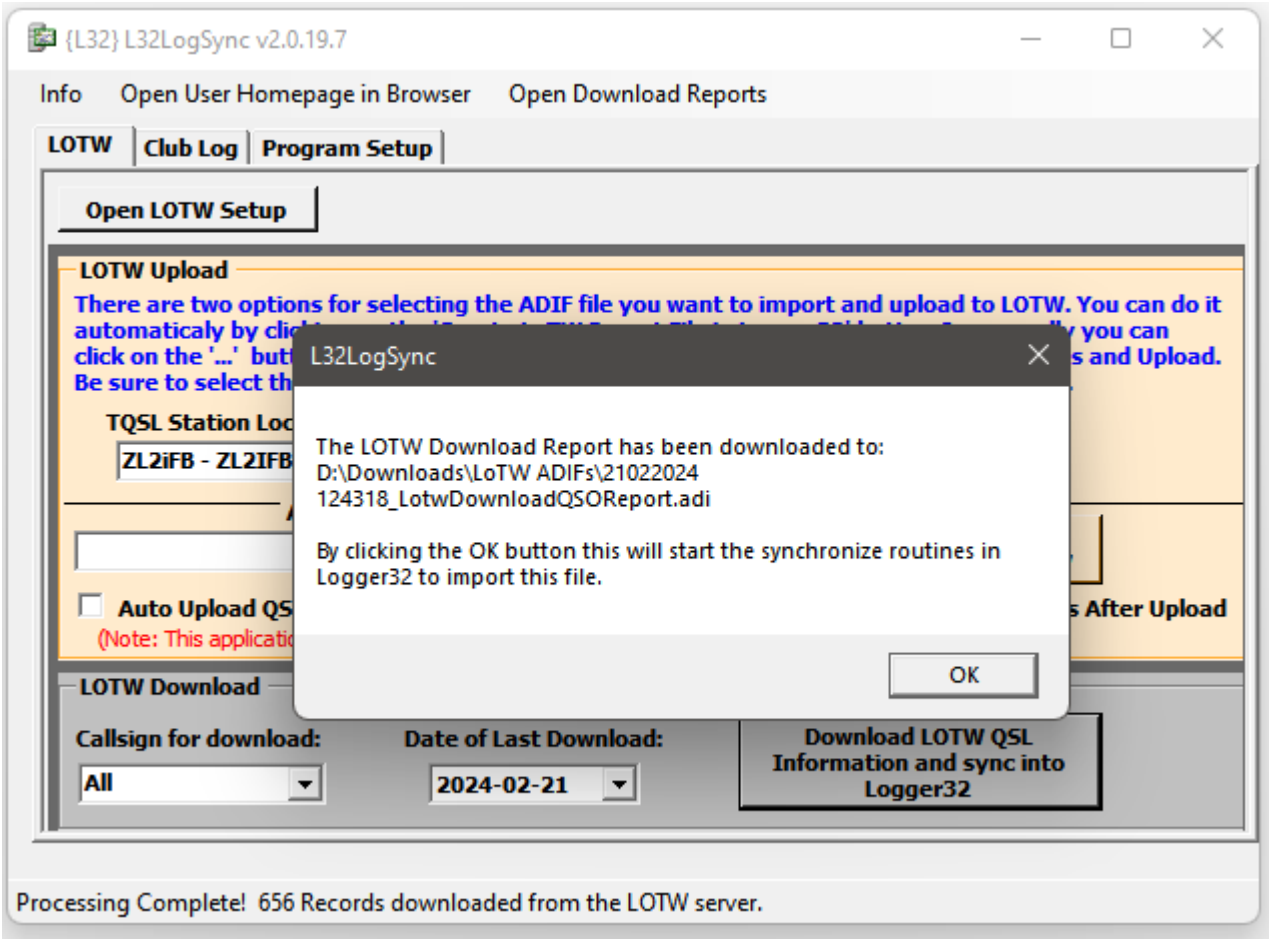
logbook to update, Logger32 checks that the OPERATOR field in each downloaded QSO record matches the OPERATOR field recorded for the corresponding QSO in your logbook.

- **Date of Last Download**: LoTW routinely records the dates and times at which QSOs are confirmed (matched), as well as the dates and times the QSOs were logged by both parties. It also records which confirmation you downloaded last, with the option to start downloading only confirmations received *after* your last download. L32 LogSync lets you choose any start date for confirmations by typing it in or selecting it from the calendar[294].
- **Download LOTW QSL Information and sync into Logger32**: having completed those two data entry fields, click the go button!



◀ Messages at the bottom of the L32 LogSync window keep us informed of progress.

Shortly afterwards, the confirmations have been extracted from the LoTW database and downloaded as an ADIF file ▼



[294] Although both LoTW and L32LogSync only show the *date* of the last download (not the date and time), it is OK to start the next download from the <**Date of Last Download**> as shown in L32LogSync. Any confirmations generated in LoTW *after* your previous download, including later on the same UTC day, *will* be included in the next download. Occasionally, however, you may like to pick a start date months or years earlier in order to grab and synchronize *all* the intervening confirmations, just in case any were somehow missed along the way (*e.g.* if you downloaded a batch of LoTW confirmations but got interrupted while updating your log). You'd better be patient, though. With a lot of data, It takes a while to turn the cogs.

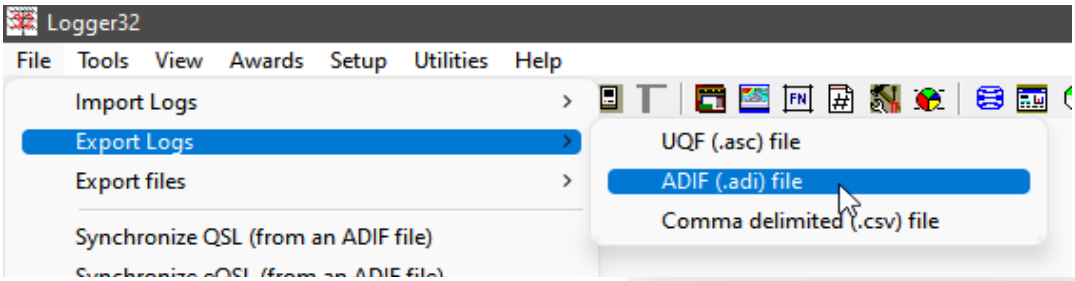
Copyright © 2026 Robert C. Furzer K4CY

The downloaded LoTW file is an ADIF file containing details of confirmed QSOs *i.e.* the essential fields for any QSO (*e.g.* both callsigns, the date and time, the band and/or frequency and mode/submode) and the following location fields:

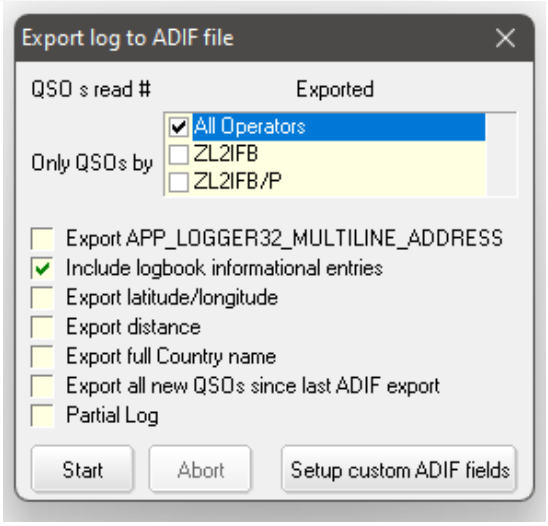
- **County**: a Secondary Administrative Subdivision (*e.g.* US county, JA Gun).
- **Grid square**: the ADIF standard permits 2, 4, 6 or 8-character grid squares.
- **ITU zone**: one of the 75 areas of the word defined by the ITU.
- **CQ zone**: one of the 40 areas of the word originally defined by CQ Magazine.
- **IOTA**: an Islands On The Air reference (*e.g.* OC-036 for North Island, New Zealand).
- **State**: a Primary Administrative Subdivision (*e.g.* US State, JA Island, VE Province).

Click <**OK**> to proceed, then click to select the ADIF file that has just been downloaded (the most recent file in the folder shown) and <**Open**> to launch Logger32's LoTW sync function.

5. While you wait, make a complete log export in ADIF format, and save it somewhere safe using **File ⇨ Export Logs ⇨ADIF (.adi) file ▼**



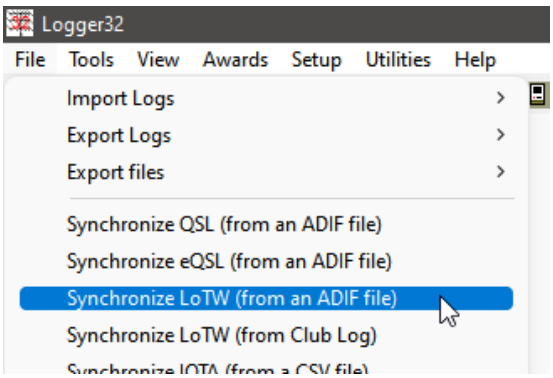
To export a copy of everything, choose the entire date range of your log (*not* a <**Partial Log**>) for <**All Operators**> and tick to <**Include logbook informational entries**> ▶

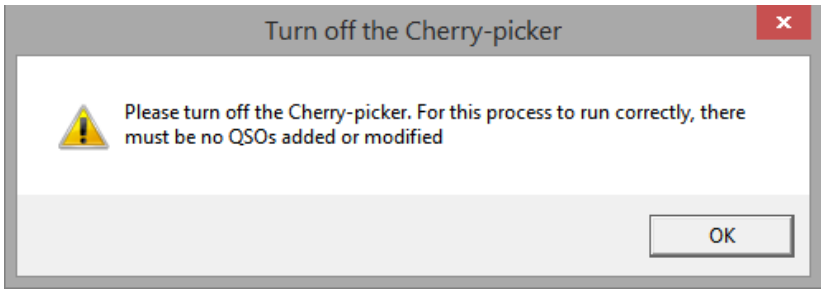


Hinson tip: you *probably* won't need it but if you ever do, you'll be glad you took the trouble. Trust me. The other information offered on the file export form isn't important to me – I can readily recreate it from the basic QSO info – but if it is important to you, tick those options as well.

Copyright © 2026 Robert C. Furzer K4CY

6. When the download is complete, run Logger32's LoTW synchronization function using **File ⇨ Synchronize LoTW (from an ADIF file)** ▶





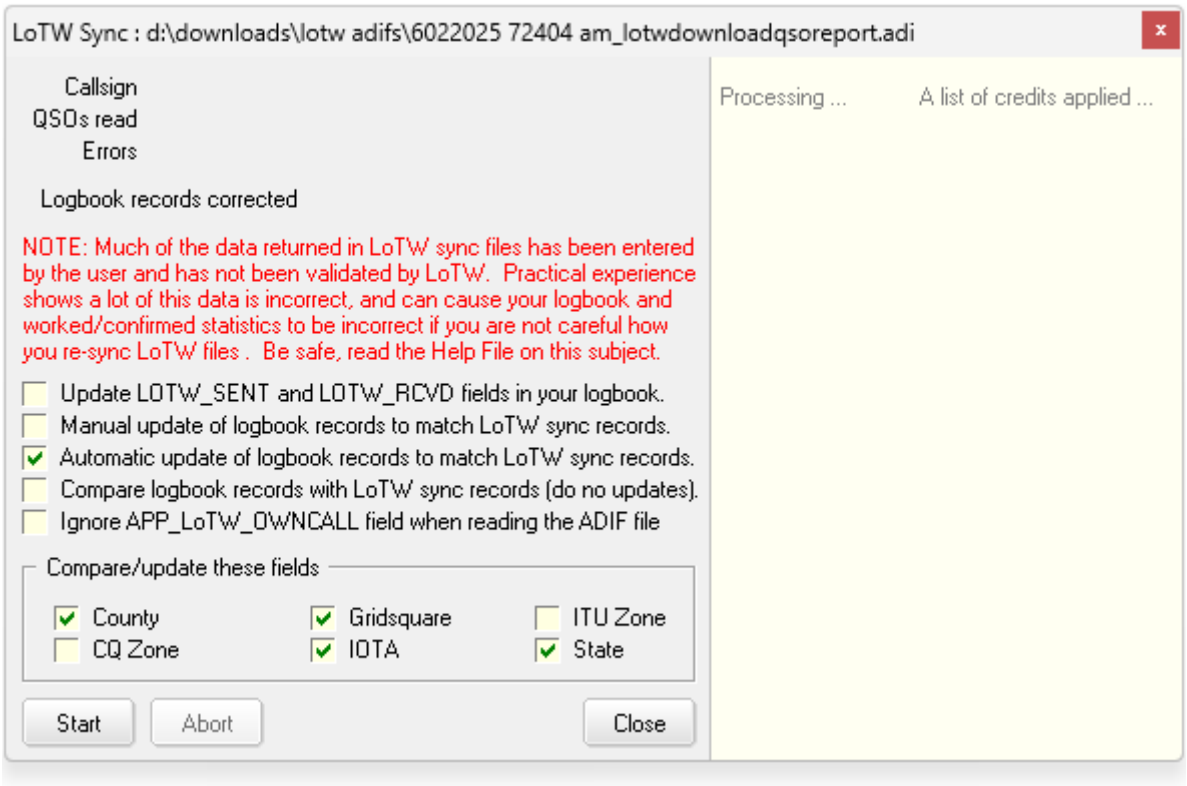
◀ Do not attempt to log additional QSOs *while* Logger32 is busy updating your log – which explains the warning message if the UDP BandMap's cherry-picker function is running. If you see this message, click <**OK**> to acknowledge and dismiss it, then click to un-tick and disable the cherry picker, and finally start the LoTW synchronizer again.

7. Find and click to select the .ADI file downloaded from the LoTW website, most likely in your browser's default download folder, then click <**Open**>.

> **Hinson tip**: check the configuration settings/options in your browser if you're not sure where to look. It varies across browsers and is usually user-configurable. In Chrome, for example, the shortcut <**Ctrl+J**> opens a list of recent downloads. Top of the list will be a file similar to **lotwreport.adi** and beneath that <**Show in folder**>: click that link to open the download folder … and discover where on disk it is!

Copyright © 2026 Robert C. Furzer K4CY

8. The LoTW synchronization function opens a dual-purpose form to select various sync options and then show you stuff happening once the sync is started ▼



Notice the dire warning in red. Despite improved guidance and data validation in TQSL since LoTW was released, there are *often* errors in the location fields – particularly the ITU and CQ Zones. Automated updates can apply *in*correct data, automatically.

I recommend the manual process.
The garbage that users enter is scary.
K4CY

Hinson tip: once the synchronizer is <Start>ed, callsigns will start appearing in the yellow panel on the right as Logger32 takes successive QSO records from the LoTW ADIF, then tries to find them in the open logbook, checking and if appropriate updating the log accordingly.

9. Click to tick your choice of options on the LoTW sync form:
   - <**Update LOTW_SENT and LOTW_RCVD fields in your logbook**> checks and if necessary updates those logbook flags to show that certain QSOs have been sent to LoTW then matched/confirmed and returned back from LoTW in the download file.

Hinson tip: this option does *not* change the DXCC granted status for those QSO or your DXCC award statistics or any other fields: all it does it set both the LoTW sent and LoTW received flags for QSOs that have been confirmed on LoTW. It does what it says on the tin.

Copyright © 2026 Robert C. Furzer K4CY

- <**Manual update of logbook records to match LoTW sync records**> is a fairly safe option that lets us check each proposed log status change, one at a time, and either accept or reject it[295]. See ▶
- <**Automatic update of logbook records to match LoTW sync records**> is a little less safe since Logger32 makes all the QSO status changes it determines are needed from the download file. See ▶

Note: if we elect to update the logbook manually or automatically, we can use the lower panel on the form to choose which of six station location ADIF fields to update in our log with the data from LoTW *i.e.* the QSO partner's **county** (secondary administrative subdivision), **gridsquare**, **ITU zone**, **CQ zone**, **IOTA** reference and **state** (primary administrative subdivision). Any un-ticked fields will be ignored: any data already in your log for these fields and QSOs will remain unchanged.

Hinson tip: despite being risk-averse and loath to corrupt my precious logbook, I usually select the automatic update option, safe in the knowledge that I made a full log backup way back at step 1 of this process … plus all my usual daily, weekly, monthly and annual just-in-case data backups. If you are worried about losing control, *don't* choose automatic updates!

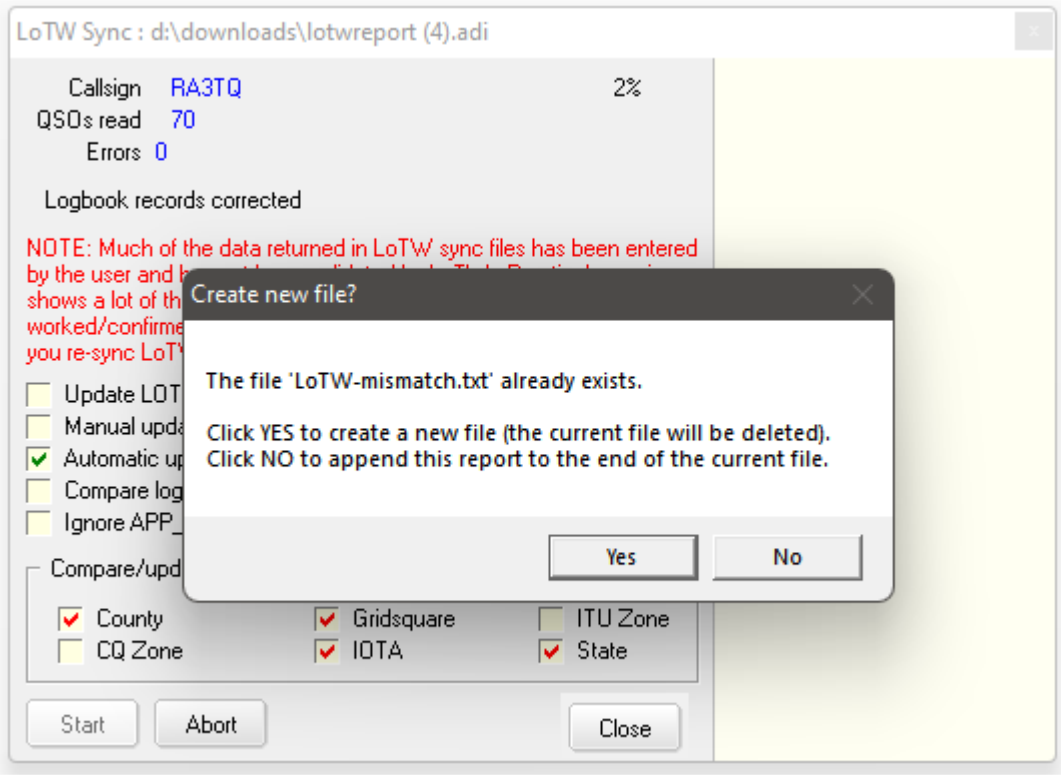
- <**Compare logbook records with LoTW sync records (do no update)**> is a safe option that does the comparison of QSL records from the LoTW download file against the QSO records in your log and tells you what it finds: it makes no changes to your log.
- <**Ignore APP_LoTW_OWNCALL field when reading the ADIF file**> tells Logger32 to ignore this deprecated field, saving a few CPU cycles. This is recommended for all.

Hinson tip: in my experience, the CQ and ITU zones in LoTW ADIF downloads are the least trustworthy, whereas Logger32 can usually determine the zones accurately using other location information such as that from Club Log – so I un-tick both zone fields, but tick the other four, despite knowing that there may be errors there too. Too bad! If you are worried about data errors in *any* of the location fields, simply un-tick them on this form to ignore the corresponding ADIF data fields from LoTW.

[295] In fact, <**Manual update**> *automatically* updates the LoTW_RCVD flags in the log *provided* all the selected elements match for a given QSO. It only asks for permission to update QSOs *if* the data from at least one selected field differs between the LoTW confirmation and the QSO in the logbook.

Copyright © 2026 Robert C. Furzer K4CY

10. Click <**Start**> to set the synchronization process running. It will start reading-in and looking-up the downloaded QSOs, comparing the data for any differences, showing the percentage complete and error count ▼



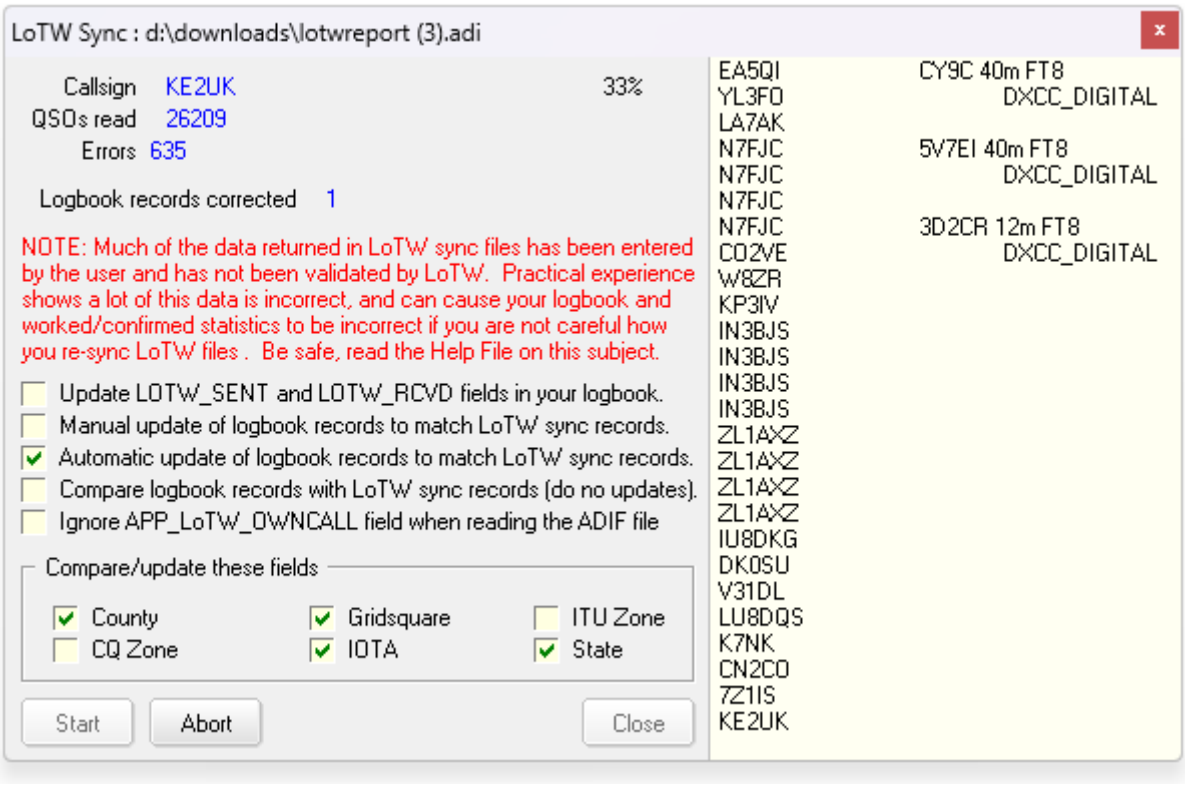
If the comparison finds any differences, Logger32 may warn you that you already have a file called *LoTW-mismatch.txt* in your Logger32 folder from some previous LoTW checks/updates, giving you the option <**Yes**> to delete that original file and start afresh, or <**No**> to keep the original file but append new information to it, in case you decide to check it out later.

Hinson tip: the information that Logger32 adds or appends to *LoTW-mismatch.txt* records those discrepancies between QSL records in the LoTW downloaded file, and the QSO records in your log. It is not particularly exciting but can be used to figure out why the synchronization process doesn't always proceed as expected – for instance when no corresponding QSO can be found in your log because you deleted it or lost it, somehow, after having signed and uploaded it to LoTW at some point prior.

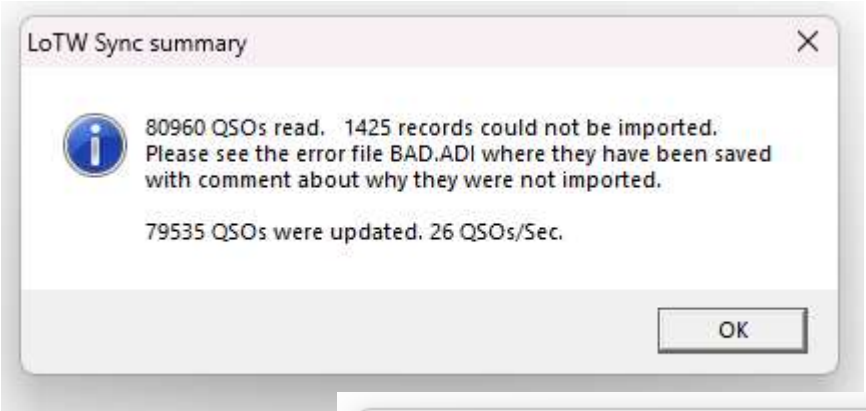
Hinson tip: an entry added to my Utilities menu lets me view or edit *LoTW-mismatch.txt* easily with just one click.

Copyright © 2026 Robert C. Furzer K4CY

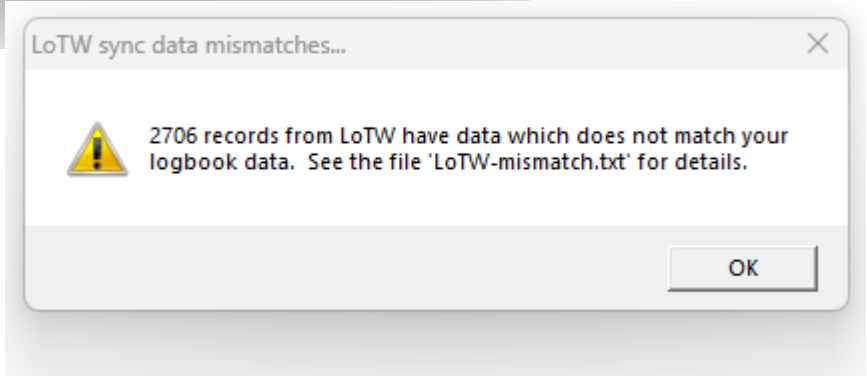
11. Watch in *awe* as Logger32 zips rapidly through the LoTW confirmations, looking up the QSOs in the open logbook, checking for discrepancies and (depending on which options you chose) informing you and/or updating the logbook accordingly. On the right of the form, the yellow section shows if any QSOs have been flagged as granted for various DXCC awards ▼



12. Eventually, you will see a pair of completion messages, something like this ▼



◀ OK that for this ▼



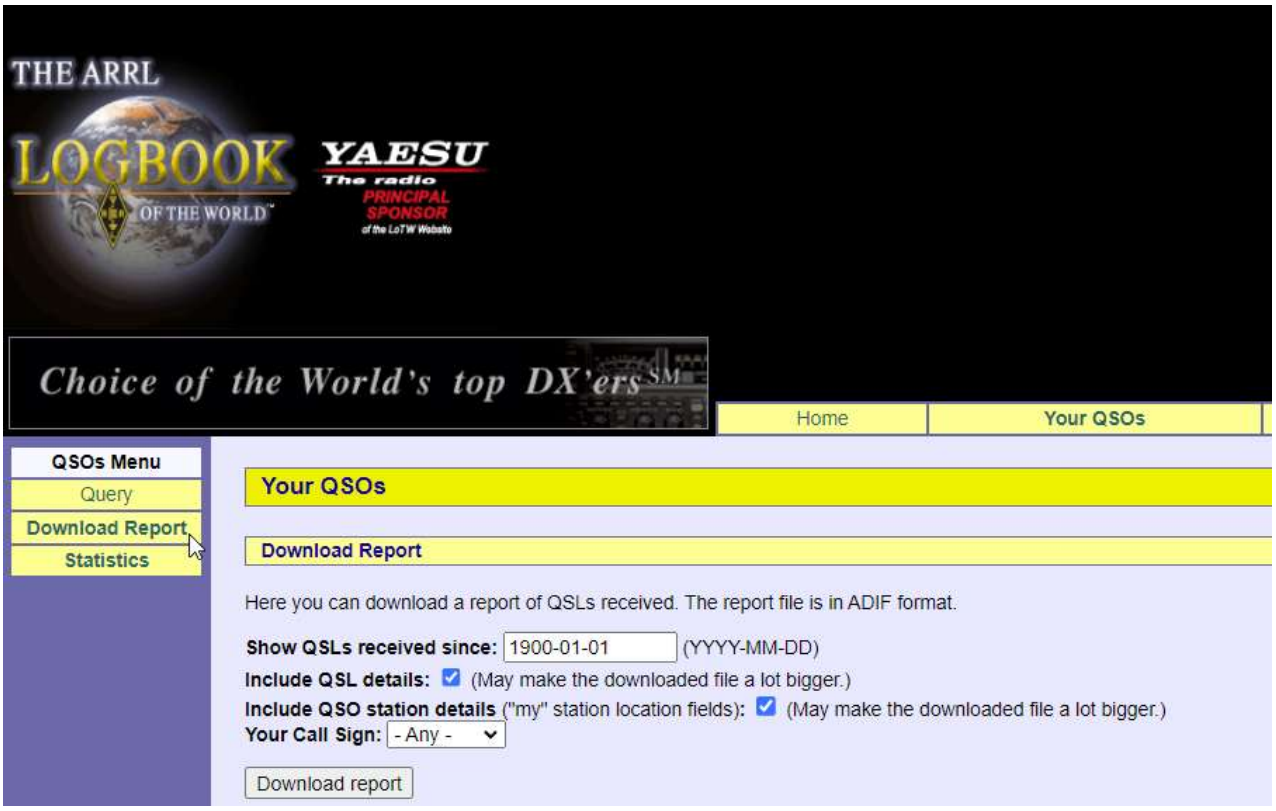
Check out *C:\Logger32\BAD.ADI* for details of the data errors detected (blocking any changes to those QSOs in the log) and *C:\Logger32\LoTW-mismatch.txt* for details of log changes made.

Copyright © 2026 Robert C. Furzer K4CY

17.3.3 Automatic LoTW -> log synchronization using manual downloads

If you can't or don't want to use the L32LogSync add-on, you can download the ADIF file of confirmed (matched) QSOs from LoTW manually:

1. Login to LoTW.
2. Click to open the yellow <**Your QSOs**> tab ▼



- <**Show QSLs received since**> defaults to the last day you downloaded your LoTW confirmations (if ever), making it easier to download only newly received confirmations each time you run the process.

> **Hinson tip**: the LoTW system has no idea whether you have actually updated your Logger32 logbook with the downloaded details: if you ran the download but then forgot to update Logger32, or if Logger32's LoTW synchronization process failed or was interrupted for some reason, LoTW will not send you those confirmations again *unless* you pick an earlier date. Every so often (maybe on your birthday?), you might like to download ***all*** your LoTW confirmations by deleting the date from the date field. Leave it blank to get *all* your confirmations, ever.

- Tick <**Include QSL details**> to receive additional location information for your confirmed QSOs If you do not tick <**Include QSL details**>, the basic QSO information is only sufficient for Logger32 to identify the confirmed QSOs and update LoTW_RCVD flags in your logbook[296].
- Click <**Download report**>. If given the option, choose a folder in which to save the downloaded *lotwreport.adi* file *e.g. C:\Logger32*.

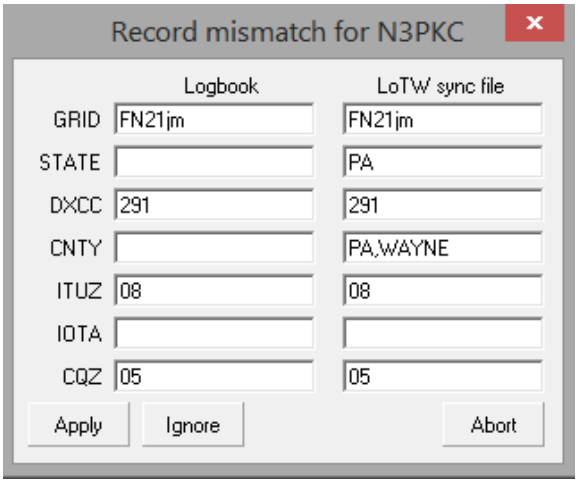
3. Now go to step 6 in the previous section to run Logger32's LoTW sync function.

[296] Logger32 also ensures that the LoTW_SENT flags are set as well: you might, for example, have signed and uploaded a contest log from N1MM+ through TQSL, then imported that log into Logger32, without Logger32 flagging the QSOs with LoTW_SENT. When the LoTW confirmations arrive, however, Logger32 makes the obvious connection and sets *both* the LoTW_SENT *and* LoTW_RCVD flags for the confirmed QSOs.

Copyright © 2026 Robert C. Furzer K4CY

17.3.4 Manual LoTW synchronization

If you simply don't trust the automatic synchronizers or have a problem that requires intervention, select the <**Manual update of logbook records to match LoTW sync records**> option on the **File ⇨ Synchronize LoTW from an ADIF file** form to check and update your log one change at a time.

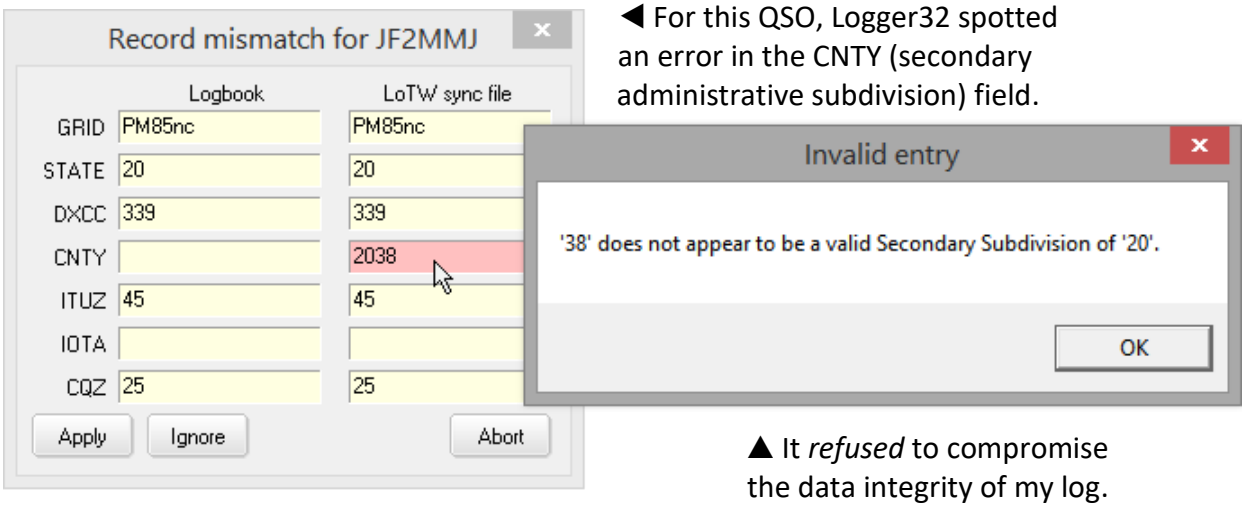


Logger32 displays each mismatched QSO in two columns showing the data presently in your logbook (on the left) and in the downloaded LoTW sync file (on the right) ▶

If you are not happy with the details from LoTW, you can skip directly to the next QSO without updating the mismatched QSO in your log by clicking <**Ignore**>.

To update this QSO in your log with *all* the data from the LoTW sync file column and move on to the next QSO, simply click <**Apply**>. In the example above ▲, the state and county fields would be updated for that QSO in my log using information supplied by N3PKC.

You can *selectively* update specific fields in this QSO record in your log by editing or deleting any of the LoTW sync file field contents *before* clicking <**Apply**>. For example, if I know N3PKC had been on holiday in Florida when we worked, I could overtype PA with FL, and delete or replace the CNTY and GRID data, regardless of what he signed and uploaded to LoTW.



◀ For this QSO, Logger32 spotted an error in the CNTY (secondary administrative subdivision) field.

▲ It *refused* to compromise the data integrity of my log.

I could not <**Apply**> this change without first correcting the error[297].

If there are no notable differences between a QSO in your logbook and in the downloaded LoTW sync file, that is deemed a match: the LOTW_SENT and LOTW_RCVD fields for the QSO are both set in your logbook, and there is no need for you to check or approve anything.

[297] For me, 'correcting the error' normally means simply deleting the data from the problematic county field … which leaves some confirmed QSOs without counties in my log. Too bad. Never mind.

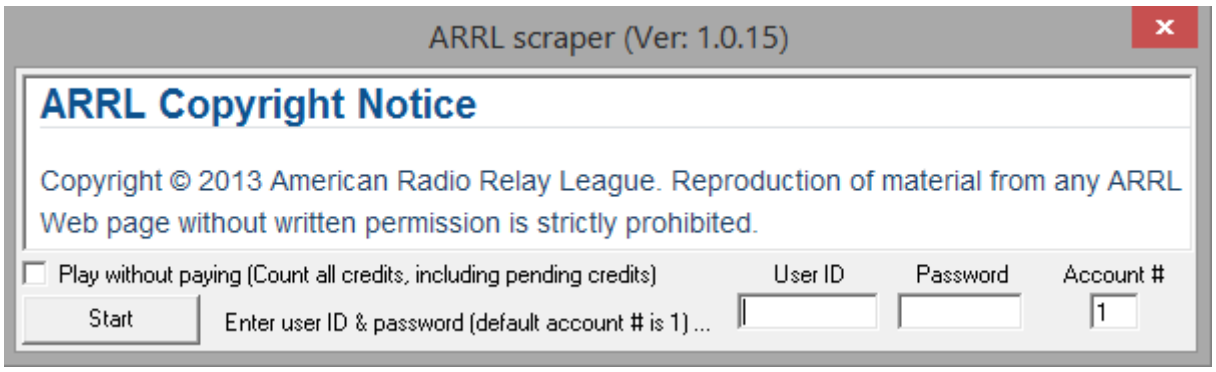
Copyright © 2026 Robert C. Furzer K4CY

17.3.5 Updating your DXCC award status with ARRL Scraper

Hinson tip: although the ARRL scraper still works, it is slow and no longer recommended.

ARRL Scraper is a standalone program provided with Logger32, to 'screen-scrape' your DXCC QSO credits (potentially *both* pending *and* granted) from the LoTW website into a data file.

Double-click *C:\Logger32\ARRL scraper.exe* to run it, then input your LoTW credentials ▼



The ARRL Scraper program:

1. Logs in to your LoTW account (account #1 by default – you can choose other accounts if you have several) using the User ID and Password you supplied.
2. Requests and downloads DXCC summary reports showing which country/band/mode slots have been filled for various DXCC awards.
3. Requests and downloads detailed reports on each of the QSOs that fill the slots: this step takes quite a while.
4. Parses the detailed reports, compiling the detailed QSO data with the applicable DXCC fields and saving it on disk as a data file (named *C:\Logger32\[Callsign]_[Account #]_granted_credits scraped data.txt*) containing bar-delineated QSO records of this form ▼

```
3B7FQ|004|2009/02/09||CW|17m|QSL|https://lotw.arrl.org/lotwuser/carddetail?card=858850017

3B8/EA5IDQ|165|2015/05/16|05:41:39|SSB|12m|LoTW|https://lotw.arrl.org/lotwuser/qsodetail?qso=676325161

3B8/SP2FUD|165|2010/04/13|06:47:58|RTTY|17m|LoTW|https://lotw.arrl.org/lotwuser/qsodetail?qso=277827414
```

Then, in Logger32, run **Awards ⇨ QSL update for DXCC/IOTA awards.** Browse and select the scraped data file, then click <**Start**>. Logger32 systematically searches your logbook to find the respective QSOs, setting the DXCC flags accordingly.

If matching QSOs are not found (*e.g.* if the record in LoTW showed you had been credited with working "P5/3Z9DX" whereas you logged, signed and uploaded a QSO with "3Z9DX/P5", or if the confirmed DXCC entity is different from your logged DXCC entity), you will have the chance to find, select and flag the relevant QSOs as confirmed on LoTW.



You may like to setup *ARRL Scraper.exe* as one of your utility programs ▶

Copyright © 2026 Robert C. Furzer K4CY

Run ARRL Scraper and you will see ▼



- **User ID and password**: enter the username and password you use to logon to the LoTW website.
- **Account #**: this is your DXCC account number as determined by the LoTW system.
  - If you have only one DXCC account, it is number 1.
  - If you have multiple DXCC accounts, you need to determine their numbers. First, logon to the LoTW website through your browser ▼



Then click to open the <**Awards**> tab ▼

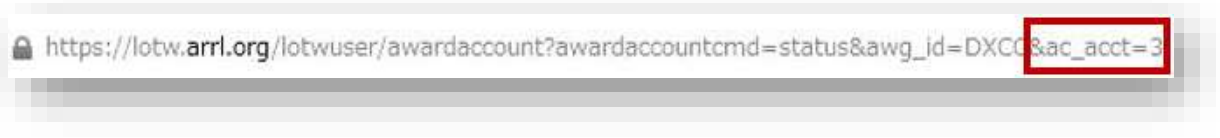




◀ Under <**Your LoTW ARRL DXCC (DX Century Club) Account(s)**> on the left side, click to select the account for which you want the number, then click the <**Select DXCC Award Account**> button.

Copyright © 2026 Robert C. Furzer K4CY

In your browser's address field, look along the URL for "ac_acct=N", N being the account number you need. In this example, the account is number 3 ▼

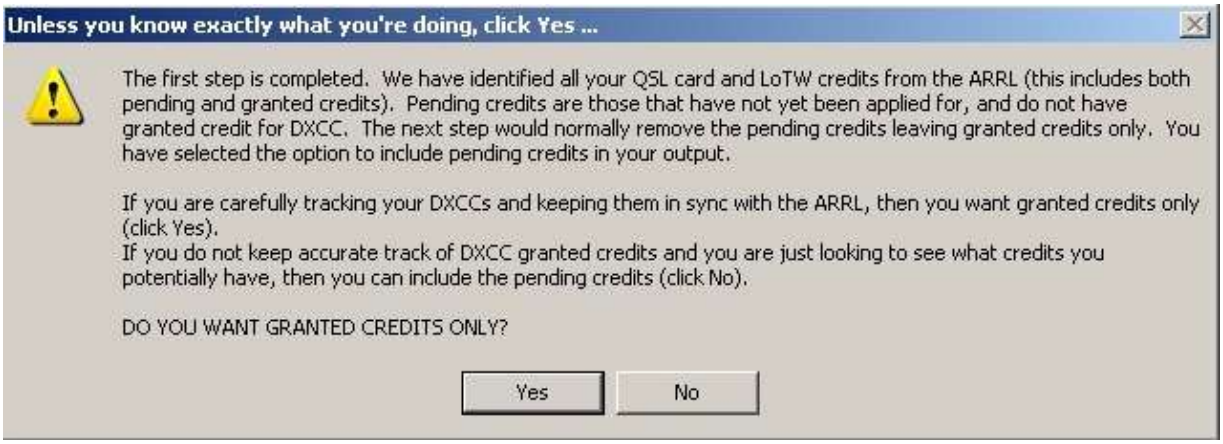


Repeat the process for any further DXCC accounts to get their numbers.

- **Play without paying**: ticking this option scrapes any pending credits (which are free), plus any that have been formally granted by ARRL towards your DXCC awards (which costs money), and reports them as if they were *all* granted ▼



Read the warning message carefully ▼



To count Granted credits only, either click <**Yes**> or simply (since the Yes button's bold outline indicates it is the default option) hit <**Enter**>. To count *all* credits, both Pending and Granted, click <**No**>.



Click <**Start**> ▲ and, as the processing takes place, its status is shown on a progress bar while scraped QSOs flash through the form. Taking approximately 20 minutes to scrape 1,000 records, the process may drag along for an hour or more if you are scraping a large number of records, especially on a slow/unreliable Internet connection.

Copyright © 2026 Robert C. Furzer K4CY

Look out for any error messages *e.g.* ▼



Scraped records are saved in a text file in the same folder as *ARRL Scraper.exe* (usually *C:\Logger32*) and the scraper closes automatically. The text file name is dependent on whether the <**Play without paying**> option has been selected:

- If so, the file name is like JA1NLX_3_all_credits scraped data.txt
- If not, the file name is like JA1NLX_3_granted_credits scraped data.txt

  Callsign ⮥ ⮤ Account number

17.4 Handling USACA county/LoTW discrepancies

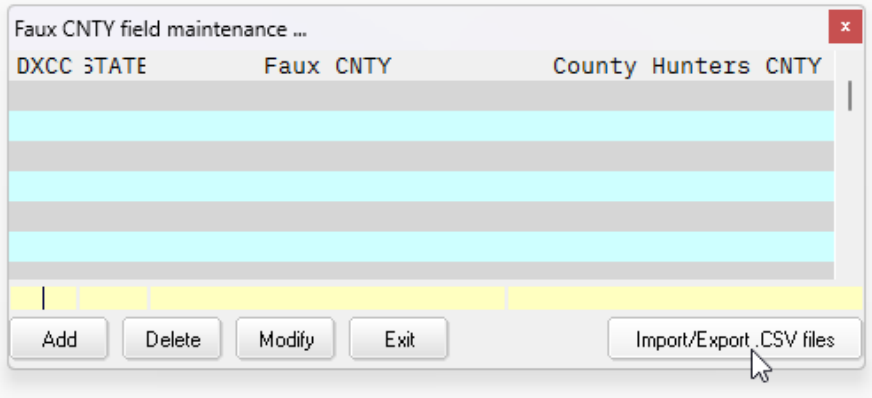
Discrepancies between the US counties that TQSL and hence LoTW accepts, and the 'official' list of US counties that qualified for the **USA C**ounties **A**ward (which Logger32 uses to validate the LoTW confirmations), can make the process of updating our logs to record LoTW confirmations quite tedious.

If we simply elect to ignore the counties and synchronize our log with LoTW automatically, the process flows nicely … but Logger32 does not save the confirmed counties nor update its USACA statistics to reflect any new ones now confirmed.

Logger32 has a function to translate certain US counties from the TQSL/LoTW county list into their equivalents on the USACA county list, such that LoTW synchronization flows painlessly, the counties are recorded in your log in the USACA format, *and* your USACA award statistics are updated appropriately.

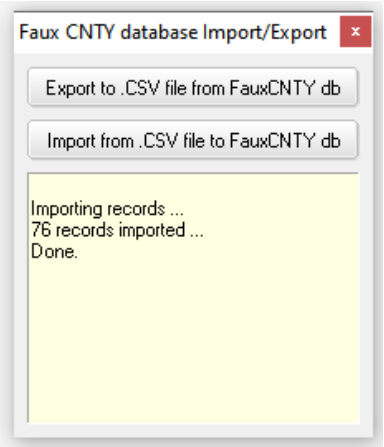
17.4.1 Load the CNTY-equivalence database

1. If it is not already available, copy *FauxCNTY.CSV* from the *C:\Logger32\updateFiles* folder to *C:\Logger32*.
2. In Logger32, launch the CNTY-equivalence database maintenance function using **Tools ⇨ Database maintenance ⇨ Faux ADIF defined cnty maintenance**.



3. Click the <**Import/Export .CSV files**> button to open a form ▲ then click the <**Import/Export CSV files**> button.

Copyright © 2026 Robert C. Furzer K4CY

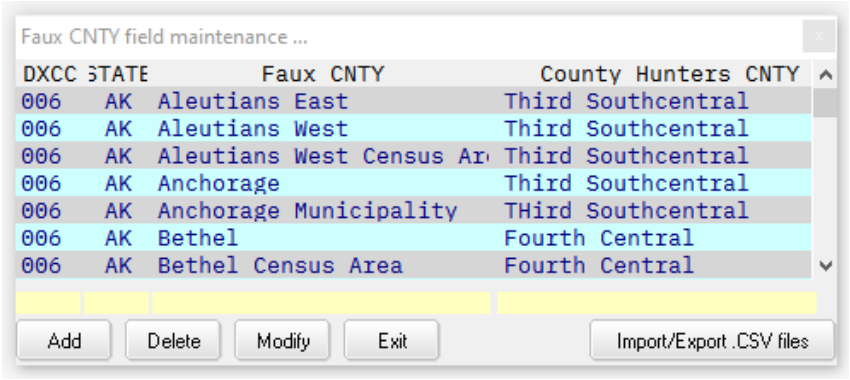


4. Click the <**Import from .CSV file to FauxCNTY db**> button ▶

5. Find and then click to load the *C:\Logger32\FauxCNTY.CSV* file. Read the messages in case there are any import errors.

6. Close the database import/export function by clicking the the top right corner X.

7. Take a moment to browse the Faux CNTY database and explore its maintenance function ▶

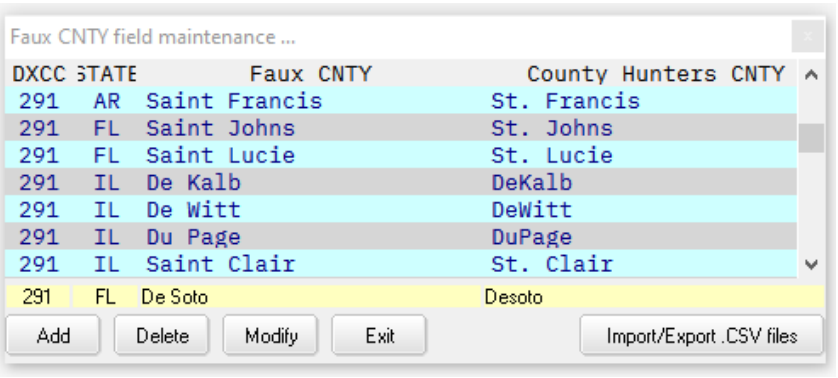


   Records in the Faux CNTY database have the following fields:

   - DXCC entity number.
   - State (primary administrative subdivision code).
   - Faux CNTY – the secondary administrative subdivision name used by TQSL and LoTW.
   - County Hunters CNTY – the equivalent US county name used by the USACA award scheme, and in future by your Logger32 log.

8. If you wish, the <**Add**>, <**Delete**> and <**Modify**> buttons let you edit records in the database, perhaps adding further equivalents – for instance, QSOs with a few 'independent cities' can be counted for the adjacent USACA counties: Logger32 automatically assigns those cities that are entirely surrounded by a single USACA county to that county, but for cities bordered by two or more adjacent counties, *you* can choose which *one* to use for each independent city (ideally one that would otherwise be unconfirmed and perhaps unworked).

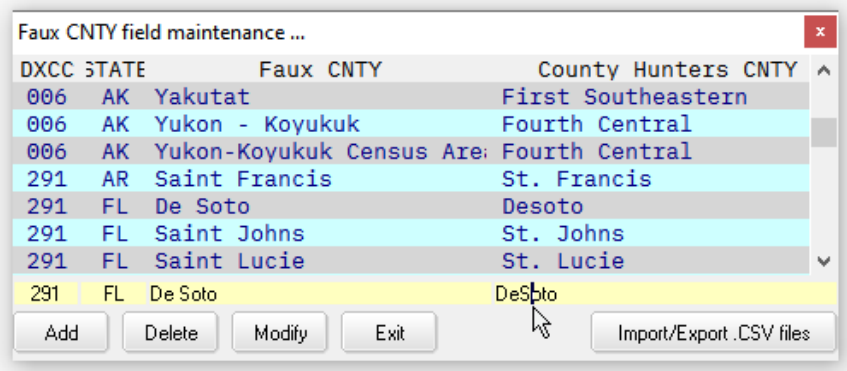
   To add an entry, type into the yellow boxes ▶ the following info:



   - DXCC entity number;
   - 2-letter US state abbreviation;
   - County name as defined in TQSL and reported in LoTW confirmations;
   - Equivalent county name as defined in the MARAC USACA rules.

   Then click the <**Add**> button.

Copyright © 2026 Robert C. Furzer K4CY



◀ To modify (correct) an existing entry, first find and click the current entry. The details appear in the yellow boxes, ready for you to edit. When you are ready, click the <**Modify**> button to save the changes.

9. If you have made any changes to the database, export the data to a *.CSV* file on disk as a backup, and potentially to share with the wider Logger32 user community.

> **Hinson tip**: if you import an *FauxCNTY.CSV* file at some later point, your edited database will be lost - deleted and replaced by the imported *FauxCNTY.CSV*. Therefore, to retain your edits (*e.g*. the equivalents you have chosen for those independent cities with multiple neighboring counties), you should either avoid importing, or export and manually merge your custom data with the updated *FauxCNTY.CSV* file first. Alternatively, keep a written note of your edits and re-apply them manually to the database after updating it.

10. Close the CNTY-equivalence database maintenance function by clicking the <**Exit**> button or the X in the top right corner.

> **Hinson tip**: since the CNTY-equivalence function takes account of the DXCC entities, the same mechanism is technically capable of translating the names of secondary administrative subdivisions for other countries besides mainland USA and Alaska as specified in LoTW confirmations, into whatever names you want to use for them in your Logger32 log *e.g*. for other similar award schemes. However, this has not been tested and proven, yet. If you venture down this route, be *sure* to have a good backup of your log *before* leaping into the unknown … and please let us know how you get on.

17.4.2 Enable the CNTY-equivalence function

1. Right click any data-entry field in the log entry pane.
2. Click <**Setup**> at the bottom of the menu.
3. Click to tick <**Translate non-standard CNTY data to award standard**>
4. That's it, you're done! This is a set-and-forget action. The configuration is saved to *C:\Logger32\Logger32.INI* (if you are using the default configuration at the time).

17.4.3 Use the CNTY-equivalence function

There is nothing much to do or see. When you import/export ADIF data that includes counties (*e.g*. downloading and synchronizing your LoTW confirmations), Logger32 silently changes any US counties, Alaskan boroughs *etc*. that are *not* specified in the USACA award scheme to their equivalents that *are* specified, and you'll never know the difference … except that (hopefully!) LoTW synchronization now proceeds without US county errors and your USACA award records end up more accurate and complete.

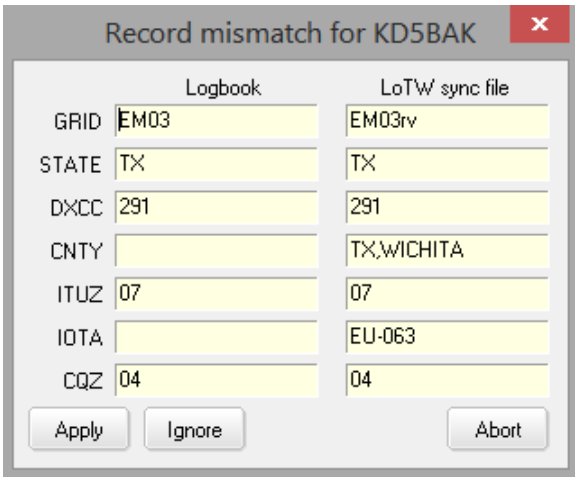
Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: some **independent cities** and other areas cannot be automatically translated to their USACA equivalent counties, so you may need to update such QSOs and/or edit your CNTY-equivalence database manually if you work and intend to claim them for the award. Check the award rules for details: at the time of writing this tip, MARAC's USACA rule C5 says *"When a station is operating in an independent city, a national park, other federal enclave, or the District of Columbia, where the neighboring county lines do not extend into the area, then the contact with that station may be counted for any adjacent county. Only one adjacent county may be counted for the award."*

17.5 Correcting IOTA errors

Information from LoTW is not entirely accurate, particularly if the original QSO data was generated by a logging program with inadequate data validation, or if mistakes were made when defining locations in TQSL. LoTW is a legacy system with poor input validation.

In this example, a Texan has accidentally (?) signed and uploaded his log to LoTW with the European IOTA island reference EU-063 ▶



Logger32 gives you the option to spot and correct errors manually during the synchronization process.

If you are synchronizing automatically to a downloaded LoTW confirmations, Logger32 will do its best to repair simple formatting errors in the IOTA information on import. For example, NA26, NA026 or NA-26 would become NA-026. Of course, if the IOTA reference should in fact have been NA-260 or OC-036, the *format* will be corrected but the *data* could still be wrong.

Hinson tip: Logger32 doesn't even notice errors such IOTA references from a different continent.

17.6 Gridsquares from LoTW

Logger32 accepts 4-, 6- or 8-character gridsquares (also known as Maidenhead locators).

Using automatic LoTW synchronization, your log retains the most detailed grid reference available. Logger32 does *not* truncate a logged 6-character grid reference (*e.g.* RF80hl) to 4 characters (RF80) if an LoTW confirmation for that QSO has the shorter form. However if only 4 characters were logged (as often happens with FT#), the reference *is* updated if the LoTW confirmation has a 6-character grid.

With manual synchronization, you can patiently edit the data as you wish, one QSO record at a time. Logger32 merely notifies you when grid references differ between QSOs in your log and the corresponding LoTW confirmations.

Copyright © 2026 Robert C. Furzer K4CY

17.7 General notes about LoTW synchronization

The LoTW synchronization routine searches the logbook for a QSO that exactly matches on date, time, callsign, band and mode. If no exact match is found, it checks for a matching QSO within ±10 minutes.

If Logger32 warns you that it has generated a *BAD.ADI* file during the synchronization process[298], check:

- The primary and secondary administrations. These are key fields in Logger32, so QSO records downloaded from LoTW must match your logbook entries *exactly* on these fields or they are rejected by the synchronizer.
- The UTC dates and times of the QSOs. They *must* be ±10 minutes of what appears in your log. The ADIF fields have the format *<QSO_DATE:8>20211023* (meaning 23rd of October 2021, UTC) and *<TIME_ON:6>004956* (that's 49 minutes and 56 seconds past midnight, UTC).
- Any IOTA island group references in the ADIF-compliant format xx-nnn *e.g. <IOTA:6>NA-033.* On air and in DX spots, hams often omit the hyphen and/or the leading zero (*e.g.* "NA33"), and some give out the wrong references: although you may have logged IOTA references accurately as they were sent, they may still be wrong.
- ADIF-compliant grid squares must be of the format XXnn or XXnnxx or (rarely) XX or XXnnxxnn *e.g. <GRID SQUARE:6>FM72cd* or *<GRID SQUARE:4>FM72*. Ignore the case.

If you are using more than one callsign with LoTW, and if you are not paying attention, LoTW may download confirmations for *all* your callsigns in one consolidated *lotwreport.adi* file. Luckily, during import, the LoTW field STATION_CALLSIGN in *lotwreport.adi* is checked against the **operator** field in the open logbook for each QSO, updating only those records that match.

By the way, the program-specific LOTW_SENT and LOTW_RCVD fields will probably not be recognized by other logging software, as these are not ADIF compliant.

17.8 Diagnosing mismatched confirmations

If Logger32 cannot locate a QSO in your log corresponding to a confirmation from LoTW, it writes the QSO as an ADIF record to *C:\Logger32\BAD.ADI* … and there the fun begins. You now have to try to locate the QSO in your log manually, and work out how come it was not found by Logger32.

Going back a step, receiving the confirmation from LoTW means you *must* originally have signed and uploaded a QSO to LoTW, and so did the person you contacted. So, if you have received an LoTW confirmation, why wasn't the QSO readily identified in your log by Logger32?

You can simply view the QSO record in *BAD.ADI* by eye, picking out the key parameters such as the date, time, band and callsign of the person you worked, then checking the log for a likely match. Although ADIF is a plain text format, it is quite tricky to make out the details since it is intended to be machine-readable … but it is possible. You might even take the trouble to break out each ADIF field and value to a new line in the file to make it easier to separate them.

LoTW allows some leeway on the time of QSOs, so check your log a little before and after the supposed time of the QSO. Also check the band, mode and *both* callsigns in the confirmation. If

[298] You might like to add an entry to your Utilities menu to make it easier to locate, open, review and edit *BAD.adi* in a plain ASCII text editor such as MS Notepad or WordPad, or TED Notepad.

Copyright © 2026 Robert C. Furzer K4CY

you find a QSO with the station concerned, you can simply mark it confirmed on LoTW, or you can carry on exploring why Logger32 failed to match the confirmation to the QSO.

One way to do that is to sort both ADIF records by the field names and compare them side-by-side, adding blank lines to line them up ▼

LoTW info in BAD.ADI	QSO in Logger32 log
	<APP_LOGGER32_LAT:16>52.7291666666667
	<APP_LOGGER32_LNG:17>-41.4583333333333
	<APP_LOGGER32_QSO_NUMBER:2>83
<APP_LoTW_2xQSL:1>Y	
<APP_LoTW_DXCC_ENTITY_STATUS:7>Current	
<APP_LoTW_MODEGROUP:4>DATA	
<APP_LoTW_OWNCALL:5>OK1TK	
<APP_LoTW_QSO_TIMESTAMP:20>2020-01-18T15:53:00Z // QSO Date & Time; ISO-8601	
<APP_LoTW_RXQSL:19>2021-01-08 07:45:03 // QSL record matched/modified at LoTW	
<APP_LoTW_RXQSO:19>2021-01-08 07:45:03 // QSO record inserted/modified at LoTW	
<BAND:3>40M	<BAND:3>40M
<CALL:6>UA3RLE	<CALL:6>UA3RLE
	<COMMENT:10>/P, OK9ZAM
	<CONT:2>EU
<COUNTRY:15>EUROPEAN RUSSIA	<COUNTRY:15>European Russia
<CQZ:2>16	<CQZ:2>16

In the example above, the matching fields present in both the LoTW confirmation and the log are shown in black text. Upper or lower case doesn't matter. I have colored the fields that are present in only one of those in brown for you. The bold red line shows why these two records were mismatched: the confirmation was for a QSO with OK1TK (in the APP_LoTW_OWNCALL and STATION_CALLSIGN fields), whereas my log recorded a QSO made by OK1TK/P (the OPERATOR field). LoTW has evidently stripped off the /P portable designator from at least one of the uploaded logs and declared this QSO a match – which supports the policy of logging callsigns in full as they were sent on air, including any location modifiers and suffixes.

17.9 Satellite LoTW

If you are making, logging and uploading satellite QSOs to LoTW, two particular ADIF fields are required, namely **PROP_MODE** (specifically, <PROP_MODE:3>SAT) and **SAT_NAME** (*e.g.* AO-7 but *not* AO7, AMSAT Oscar 7 *etc*.: see the list of satellite names that LoTW accepts). You probably also want to log SAT_MODE, BAND_RX and FREQ_RX, so add all these columns to your logbook.

Most satellite QSOs are crossband. LoTW uses the BAND field to match satellite QSOs so enter the *transmit* band and frequency in the BAND and FREQ columns, using the BAND_RX and FREQ_RX columns to log the receive side.

Copyright © 2026 Robert C. Furzer K4CY

Here are some correctly-logged satellite QSOs ▼

Logbook page (C:\PROGRAM FILES\LOGGER32\N4GM)

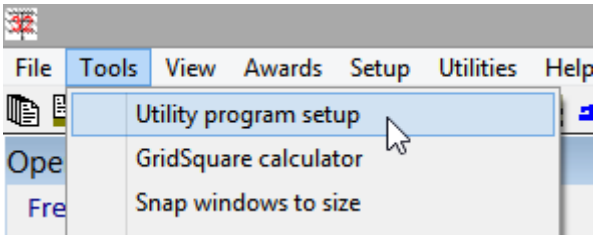
MODE	FREQ	BAND	SAT_MODE	SAT_NAME	PROP_MODE	BAND_RX	FREQ_RX
CW	145950.00	2M	V/u	FO-29	SAT	70CM	435848.00
CW	432149.00	70CM	U/v	AO-7	SAT	2M	145953.00
CW	145966.00	2M	V/u	FO-29	SAT	70CM	435836.00
CW	432239.00	70CM	U/v	VO-52	SAT	2M	145901.00
FM	145923.00	2M	V/u	AO-51	SAT	70CM	435296.00
FM	145923.00	2M	V/u	AO-51	SAT	70CM	435296.00
CW	435250.00	70CM	U/v	VO-52	SAT	2M	145900.00
CW	435250.00	70CM	U/v	VO-52	SAT	2M	145900.00
CW	435250.00	70CM	U/v	VO-52	SAT	2M	145900.00

17.10 LoTW FAQs

Q. Where *are* BAD.ADI and LOTW-mismatch.txt?

A. If they exist, you will find them in the Logger32 folder - usually *C:\Logger32*

If you don't use them often enough to remember where they are, you might like to set up entries in your Utility program menu:



Click **Tools ⇨ Utility program setup** ▶

Add lines for editing each of the files, typing a suitable menu caption on the left, and on the right a command of the form *editing program name [space] file to edit e.g.*

C:\Windows\notepad.exe C:\Logger32\BAD.ADI

and

C:\Windows\notepad.exe C:\Logger32\LOTW-mismatch.txt

Now, all you need do in future is open the <**Utilities**> menu and click one of those entries to review and maybe edit the files.

Hinson tip: although Notepad is quite easy to use and conveniently provided as part of Windows, there are more powerful plain ASCII editors out there. Additional functions may include scripting of compound editing commands, more sophisticated find-and-replace, automatic line numbering and formatting of key words *etc*. If you find yourself doing a fair bit of plain text editing, it's worth checking out potential replacements for Notepad. I prefer TED Notepad, but there are others. *Your* preference depends on the kinds of things *you* need to do and hence your functional requirements. Unfortunately, ADIFmaster seems to skip and hide Logger32's comments explaining various issues in the ADIF file – at least it did when I tried it.

Q. Having run the LoTW synchronization function and closed it, I'd like to double-check those DXCC credits granted … but I've forgotten what they were …

A. Check the temporary file *C:\Logger32\New LoTW Credits Granted.txt* before you next re-open Logger32 (it is deleted automatically as Logger32 starts).

Copyright © 2026 Robert C. Furzer K4CY

Q. I've lost the brightly-colored markers for LoTW users to the left of their callsigns on the UDP BandMap. How I can get them back?

A. Click <**Config**> on the UDP BandMap menu. Is <**Show LoTW user**> unticked? If so, click it to make the LoTW user blobs reappear. If <**Show LoTW user**> is ticked but there are no blobs, maybe there are simply no LoTW users in your decodes at the moment … but if you think there are, you might like to reload the list of LoTW users.

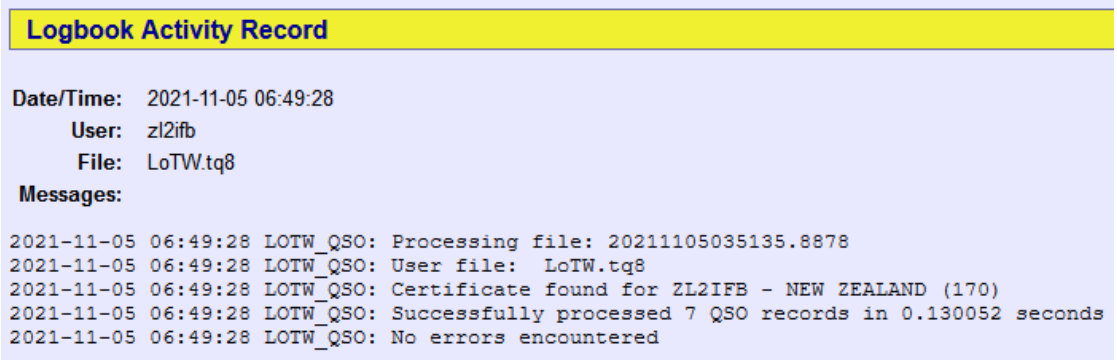
If the blobs are shown now but you don't like their color, reconfigure them. Hint: *try* to avoid choosing a blob color too similar to the background color …

> *There were black controls labelled in black, on a black background, with a little light that lit up black, which made it difficult to control the ship.*
>
> *Hitchhikers' Guide to the Galaxy*

Q. If I type my own callsign into the log entry pane, Logger32 says I haven't uploaded to LoTW for several days, but in fact I upload my log daily. What gives? If mine is wrong, can I trust anyone's last-upload dates?

A. The 'last upload' values in Logger32 are clearly not updated in real time, because:

- LoTW uploads – including yours – may take a while to process if the LoTW system is busy digesting a backlog of logs submitted following a major contest or DXpedition. Also, uploaded logs may not be processed at all if there are problems with the digital signatures or QSO data: LoTW automatically identifies and handles common issues but occasionally uploaded logs get 'stuck' in the system, holding up processing of the entire queue of logs until someone in ARRL IT notices and manually aborts the stuck log.
- Once an uploaded log is completely processed, LoTW dutifully records the date and time of completion. You can check your own upload history by logging in to LoTW, opening the <**Your Account**> tab, then clicking <**Your Activity**> on the left-side menu. Click the **Details** link for any record to see how many QSOs were processed, when, and for which of your callsigns, along with any issues such as duplicate uploads ▼



Logbook Activity Record

Date/Time: 2021-11-05 06:49:28
User: zl2ifb
File: LoTW.tq8
Messages:

```
2021-11-05 06:49:28 LOTW_QSO: Processing file: 20211105035135.8878
2021-11-05 06:49:28 LOTW_QSO: User file:  LoTW.tq8
2021-11-05 06:49:28 LOTW_QSO: Certificate found for ZL2IFB - NEW ZEALAND (170)
2021-11-05 06:49:28 LOTW_QSO: Successfully processed 7 QSO records in 0.130052 seconds
2021-11-05 06:49:28 LOTW_QSO: No errors encountered
```

You can also check someone else's last upload date through an LoTW query.

- Club Log queries the LoTW server for details of the recently-processed logs just once a week, so the data currently held on Club Log may be up to a week old.
- It may have been a while since you ran the 'last upload' function.

Copyright © 2026 Robert C. Furzer K4CY

In summary, the 'last LoTW upload' dates are simply a guide, an indication that the holder of a given callsign has signed and uploaded their log to LoTW whereupon it has been processed at that point. They *may* have uploaded subsequently as well. Either way, they are clearly LoTW users – which is nice to know.

Q. A QSO that has been confirmed on LoTW has the wrong data. What can I do?

A. Essentially you cannot change a QSO record in LoTW, especially one that has been matched and confirmed to both parties. Only LoTW administrators have the power to make such changes to the LoTW database system, and in practice they do not make individual corrections for several reasons. In short, too bad.

However, you can sign and upload a corrected QSO, hoping that the station you contacted has done or will do the same. That will generate a new matched QSO record in LoTW, this time with the correct data.

Copyright © 2026 Robert C. Furzer K4CY

18 Awards

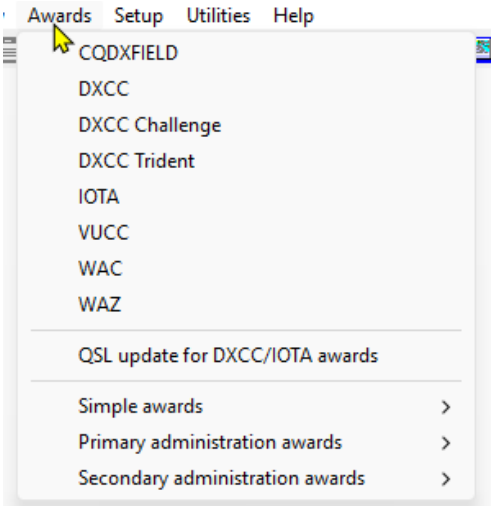
> "In the end it's about the work,
> not an award you get
> for the work"
>
> *Linda Fiorentino*

A lot of thought and effort has gone into helping you chase, track and apply for various operating awards. Logger32 sports a flexible and powerful suite of award functions and reports.

For major awards such as DXCC, Logger32 can identify and highlight potential 'new ones' for you (either all-time new ones or new this year, and new on this band, new on this mode or both) as the applicable DX stations are spotted on DX cluster, decoded by your digimode software and shown on the UDP BandMap, or entered into the log entry pane as you chase and hopefully work them.

Logger32 can monitor your progress simultaneously on multiple awards, flagging qualifying QSOs in your log and maintaining the statistics showing their worked, confirmed, claimed and granted status.

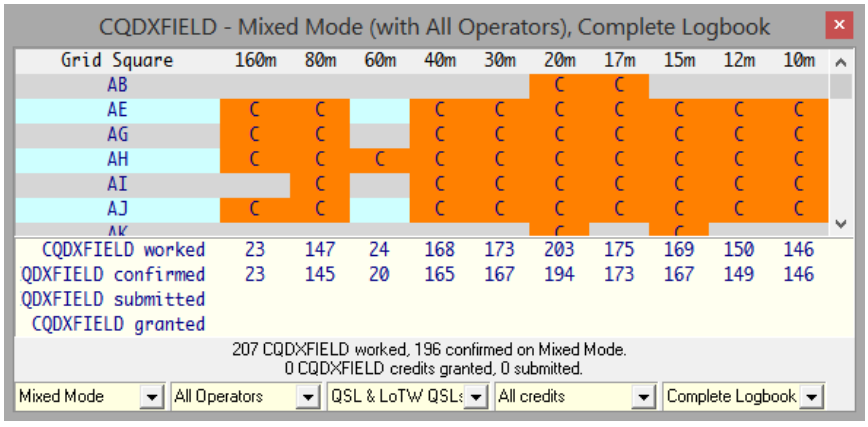
Access the awards tables/reports through **Awards** on the main menu ▶



18.1 Supported awards

The following popular awards are fully supported in Logger32 by default[299]:

- **CQDXFIELD**: the DX field award involved making contacts with Maidenhead locators around the globe, specifically at least 50 of the 324 10°x20° "fields" labeled AA through RR ▶



[299] 'By default' means these awards are readily available on a new installation of Logger32, fresh out of the box.

Copyright © 2026 Robert C. Furzer K4CY

About half of the world's surface is oceanic or covered by the shrinking polar icecaps, hence (as with DXCC), the challenge gets progressively harder as you chase contacts with the remaining fields. Mixed- and single-mode DX field awards are supported[300]. See also the grid squares simple award.

- **DXCC**: ARRL's **DX C**entury **C**lub award involves contacting at least 100 entities (countries) on the official DXCC list. The single-band and mode DXCC awards are also supported. More below.
- **DXCC Challenge** involves filling at least 1,000 band-slots. Contacting, say, Canada on 5 of the 10 bands from 160m to 6m (excluding 60m) earns 5 points towards the Challenge, even if you happen to live in Canada. You needn't qualify for DXCC on *every* band, luckily, but it is definitely challenging to amass at least 1,000 points in total. More below.
- **DXCC Trident** requires confirmed contacts with at least 100 DXCC entities on each of CW, speech and data modes. More below.
- **IOTA**: an award scheme for contacting **I**slands **O**n **T**he **A**ir within 'some 1,200 island groups' recognized for award purposes. More below.
- **VUCC**: this is the VHF/UHF version of DXCC, involving contacts with grid squares rather than entities. The ARRL VUCC awards require various numbers of confirmed QSOs with different grids on different bands - at least 100 grids on 6m, 2m or via satellite, for example, or 'just' 5 grids on the SHF bands at 3.4 GHz and above (no mean feat given the technological and practical challenges at those frequencies – not least, finding hi-tech playmates to contact).
- **WAC**: the IARU's **W**orked **A**ll **C**ontinents award involves confirmed QSOs with 'all six' continents (meaning North America, South America, Oceania, Asia, Europe and Africa: you don't *need* to work Antarctica for this award) on any mode. There is a five-band WAC for working 'all six' continents on each of the bands 10m, 15m, 20m, 40m and 80m, and endorsements for also doing so on other bands. Although not actually mentioned in the WAC rules, there are single-mode awards (CW, phone, image, digital and satellite) and QRP endorsements too (see the award application form). QSL cards (but *not* LoTW confirmations) are required.
- **WAZ**: CQ Magazine's **W**orked **A**ll **Z**ones award involved working 40 CQ zones covering the whole world. Since CQ Magazine folded, José N4BAA has taken on the administration of WAZ. There are mode, band and technology variants, offering certificates and fancy wooden and acrylic plaques.

18.1.1 Simple awards[301]

Unlike, say, DXCC, "simple" awards have no definitive predefined reference lists of valid entities. Typically for club-based awards, the club membership list is open-ended and continues to grow while you are busy hunting down, contacting and logging club members, recording and accumulating their membership numbers. For simple location-based awards, the complete list of locations would cover the whole world, including places so inaccessible that they are *never* going to be activated (such as the craters of active volcanoes!).

Simple awards mostly rely on data from one of the ADIF "USER" fields, except for the grid squares, CQWPX, 10-10 and All Africa awards which use the corresponding standard ADIF fields.

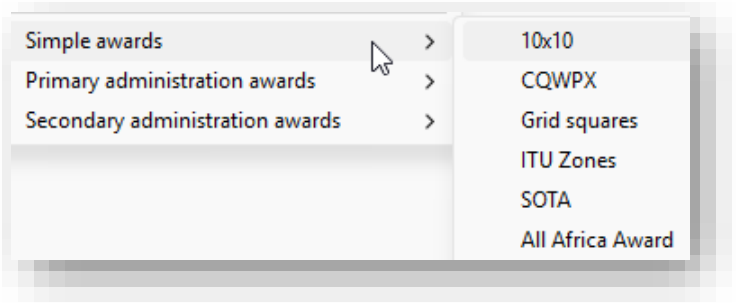
---

[300] Most awards require some form of assurance that you have, in fact, made valid contacts that qualify for the awards – typically a confirmation by some acceptable means such as a QSL card.

[301] The terms "simple", "primary" and "secondary" concern the way Logger32 handles the database tables internally, *not* the difficulty, quality, stature or nature of the awards!

Copyright © 2026 Robert C. Furzer K4CY

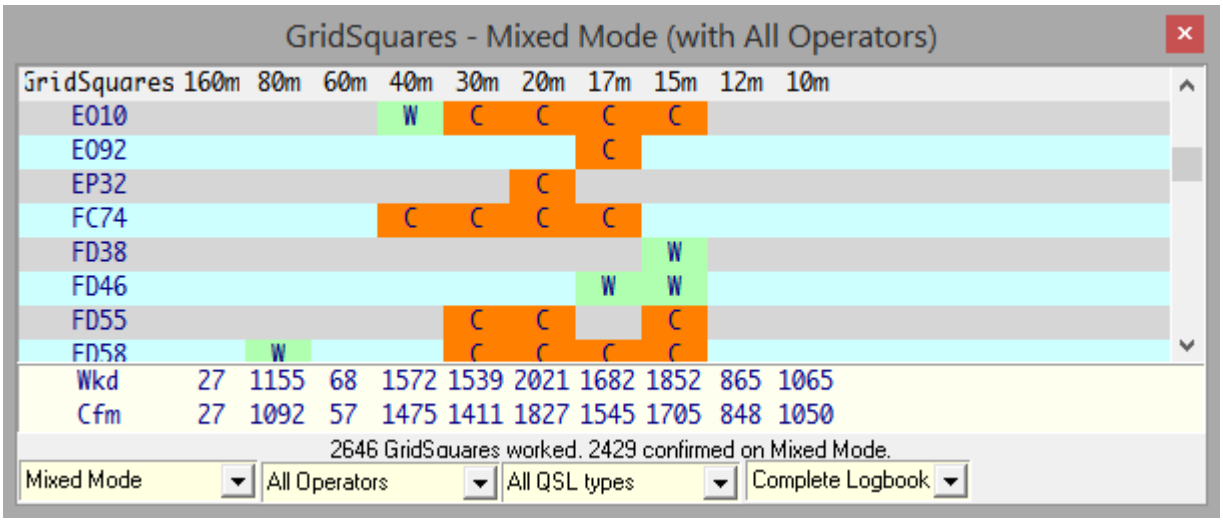
The following simple awards are supported by default ▶



- **10x10**: the Ten-Ten International Net Inc. promotes activity on 10m specifically through its award scheme. Members exchange their 10-10 membership numbers in casual QSOs, 10-10 contests and 10-10 nets on 10m. Overall interest in the club and its awards waxes and wanes with the 11-year solar cycle, for obvious reasons.

> **Hinson tip**: whereas other awards tables show statistics for several bands, only 10m contacts are valid for the 10-10 awards and hence only the 10m column is populated, even if there are 10-10 numbers recorded against QSOs logged on other bands. The clue is in the name.

- **CQWPX**: the **W**orld **P**refi**X** awards involved contacting stations with different prefixes – not just the main country designator but the number part as well *e.g.* W1, WA2, AA3, AE4, N5 and so on all counted separately. Special event and commemorative callsigns with rare prefixes such as W100AW, OE21FTDMC and 8N4OLP were attractive for the WPX awards, as well as the associated events being of interest.
- **Grid squares**: whereas the CQ DX Field award involved working the large 2-letter 'fields', some of us also enjoy chasing the smaller 2-letter-plus-2-number 'squares' within the fields as well, in our quest to *work the world*.



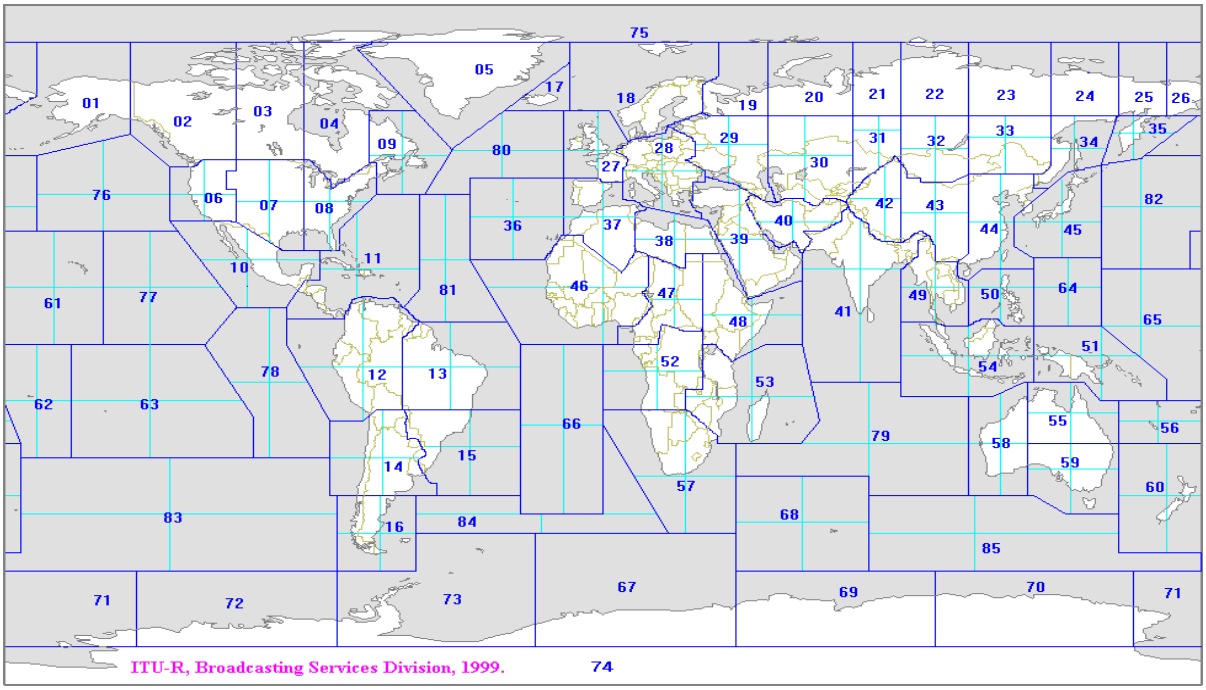
Notice that my GridSquares report ▲ does not systematically list every single grid square, but only the ones I have **W**orked so far, some of which have also been **C**onfirmed.

"L32 is one of the most reliable and customizable software I ever used. Trust me, I'm a sysadmin, experienced lots of $$$ software requiring a daily restart to have them working. Wish many programmers were like Bob :-) Ciao"

Luca IK5PWC

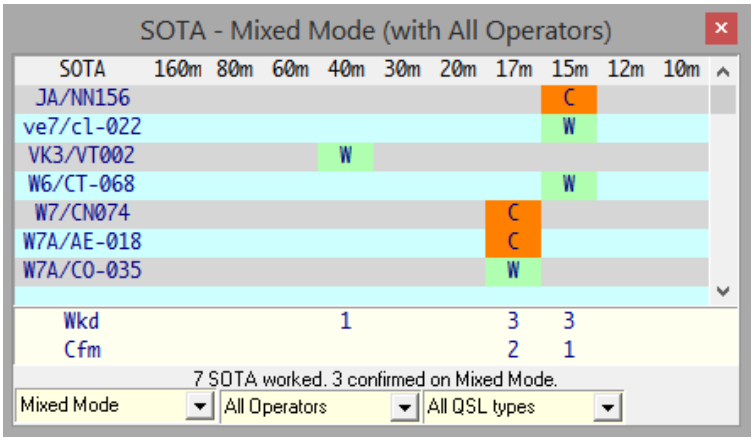
Copyright © 2026 Robert C. Furzer K4CY

- **ITU Zones**: the ITU's CIRAF zones designate 75 areas for professional broadcast coverage ▼



In amateur use, our "ITU zones" are *similar* to the CIRAF zones but not precisely the same. With limited consensus on the details and discrepancies between different amateur radio award schemes, contests *etc*., nearby stations along some boundaries may interpret the rules and so determine their ITU zones differently. Logger32 can identify gross zone errors but cannot resolve minor discrepancies: if a station claims to be in a particular ITU zone (*e.g.* through an LoTW confirmation received), their assertion is generally accepted and the zone may be logged accordingly. If you vehemently disagree, by all means change the logged zone – simply edit the zone field for the QSO in your logbook.

- **SOTA**: the Summits On The Air awards involve contacting intrepid operators clinging precariously to various high peaks and hilltops around the world, grouped and numbered according to the DXCC entities and (for the larger countries) numbered internal regions/call areas such as W7 ▶



SOTA is designated a 'simple award' since newly-identified summits are occasionally added to the reference list.

Copyright © 2026 Robert C. Furzer K4CY

- **All Africa**: the All Africa Award from the South African Radio League ▶ requires proven contacts with all six ZS call areas (ZS1 to ZS6) plus at least 25 of the remaining 50 African DXCC countries. Endorsements for single bands or modes are available and so, as with Logger32's other award reporting functions, there are options to select individual operating modes, operators (*i.e.* the callsign/s we used to make and log contacts) and QSL types at the bottom of the award report screen, while the caption along the top shows the filters currently in effect ▼



All Africa Award - Mixed Mode (with All Operators)

Countries and ZS regions	160m	80m	60m	40m	30m	20m	17m	15m	12m	10m	6m
3C - Equatorial Guinea								C			
3DA - Kingdom of eSwatini					C	C	C	C	C		
3V - Tunisia		W		C	C	C	C	C			
3X - Guinea		C		C	C	C	C	C			
5A - Libya				C	W	W	W				
5H - Tanzania				C	C	C	C	W			
5N - Nigeria				C	C	C	C	C	W		
5T - Mauritania		C		C	C	C	C	C	W	C	
5U - Niger				C		C					
5V - Togo				C		C	W	C			
5X - Uganda				C	C	C	C	C		C	
5Z - Kenya					C	C	C	C	C	C	
6W - Senegal		W		C	C	C	C	C	W	C	
Worked		12		40	51	60	55	44	31	27	
Confirmed		7		36	48	53	45	38	23	20	

50 African countries and 6 ZS regions worked. 56 confirmed on Mixed Mode.

Mixed Mode | All Operators | QSL & LoTW QSLs

The report's filter settings are automatically saved in *C:\Logger32\Logger32.ini* whenever they are changed for any simple award, and automatically retrieved from the configuration file when we next open a simple award report, so we magically pick-up where we left-off.

The body of the report shows a breakdown of our African QSOs by band, with C meaning **C**onfirmed by QSL cards or LoTW (as per the AAA rules) or W for **W**orked but not yet confirmed, with the per-band and grand total counts displayed at the bottom of the report.

Use the right-side slider to scroll through the body of the report or, to display more or less of the table, click and drag the top or bottom edge of the window vertically.

- **Others**: you can add your choice of simple awards using Logger32's award maintenance functions. Read on for details.

18.1.2 Primary administrative subdivision awards

These are awards based on contacting the states, provinces and similar major areas within various countries.

Many countries have historically been divided into regions and some are further subdivided. For example, USA has 48 contiguous (mainland) states with several counties in each state. The ADIF standard handles this through a combination of just two fields:

- **Primary administrative subdivisions:** these are substantial regions such as US states and Canadian provinces; and (where applicable)

Copyright © 2026 Robert C. Furzer K4CY

- **Secondary administrative subdivisions**: smaller areas *e.g.* US counties and Japanese guns ▼

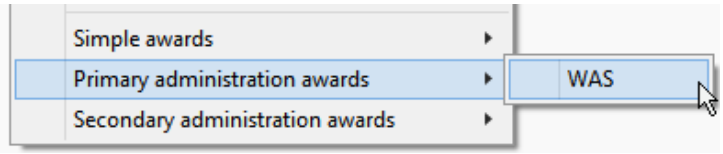
Award scheme	Example administrative subdivisions	
	Primary	Secondary
DOK	E	23 for Hamburg Trave
JCC	01	03 for Hokkaido Otaru
JCG	10	004 for Tokyo Oshima
USACA	CA	Alpine county
WAJA	12	Chiba prefecture
WAS	CA	[None]

Various award schemes specify administrative areas in which QSOs are valid for the awards, several of which are enumerated (listed explicitly) in the ADIF standard *e.g.* ▼

Enumeration for Country Code 54 (European Russia)				
Code	Primary Administrative Subdivision	Oblast #	CQ Zone	ITU Zone
SP	City of St. Petersburg	169	16	29
LO	Leningradskaya oblast	136	16	29
KL	Republic of Karelia	88	16	19
AR	Arkhangelsk (Arkhangelskaya oblast)	113	16	19
NO	Nenetsky Autonomous Okrug	114	16	20
VO	Vologda (Vologodskaya oblast)	120	16	29
NV	Novgorodskaya oblast	144	16	29
PS	Pskov (Pskovskaya oblast)	149	16	29
MU	Murmansk (Murmanskaya oblast)	143	16	19
MA	City of Moscow	170	16	29
MO	Moscowskaya oblast	142	16	29

Hinson tip: to ensure ADIF-compliance, Logger32 supports the subdivisions according to the ADIF standard. Only the specified values are permitted by default.

WAS (Worked All States) ▼ is the only primary administrative award supported by default but don't fret, you can add further awards based on contacting the equivalent primary administrative subdivisions in other countries. There's a worked example below, adding a **W**orked **A**ll **C**hina **A**ward for contacting the Chinese provinces.



18.1.3 Secondary administrative subdivision awards

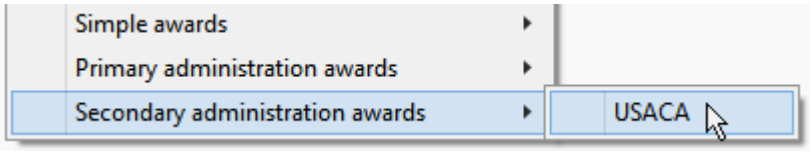
These awards are based on more fine-grained subdivisions in various countries, such as US counties and Russian districts (RDAs).

Copyright © 2026 Robert C. Furzer K4CY

The ADIF standard doesn't enumerate Secondary administrative subdivisions explicitly, instead deferring to lists and maps published on the Web by various parties – mostly the award sponsors *e.g.* ▼

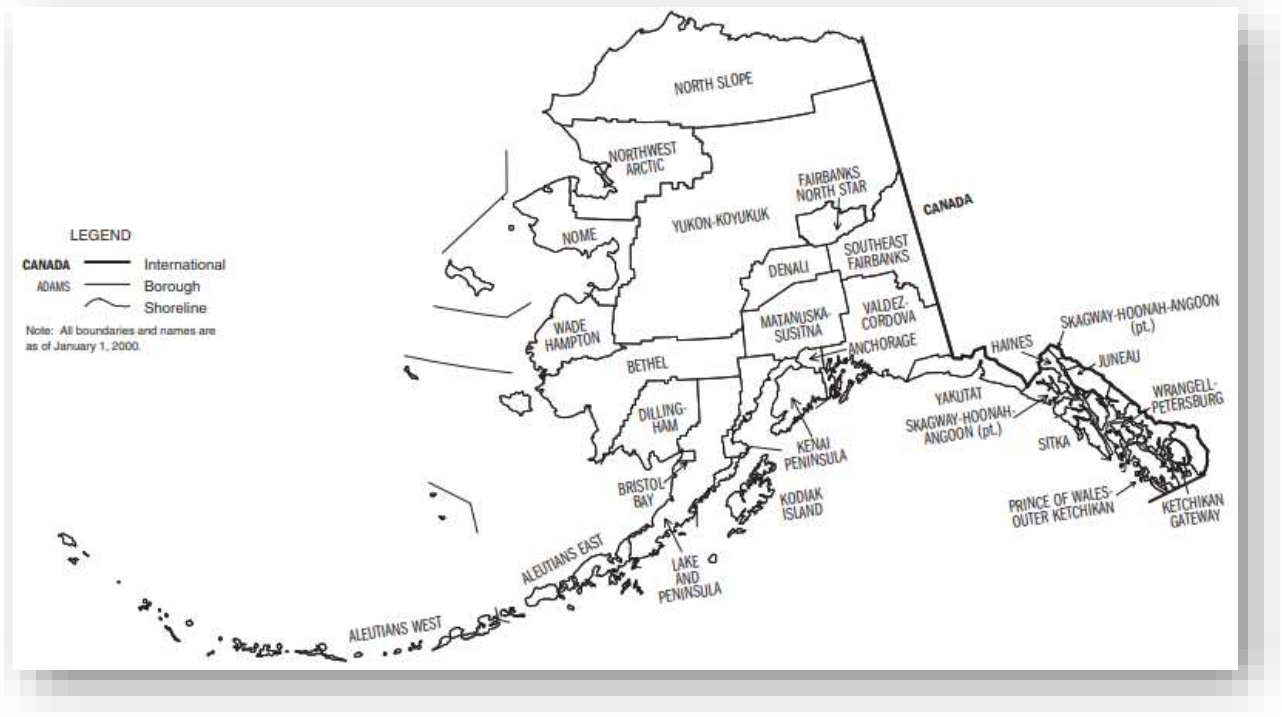
Secondary Subdivision	Country Code	DXCC Entity	Number of secondary subdivisions	Award	Subdivision List	Award Sponsor	Sponsor Defined Code Format	Examples
U.S. Counties	6	Alaska		USA-CA	Alaskan Counties	CQ Magazine [▼]	<Two-letter state abbr> COMMA <county name>	MA,Middlesex → Massachusetts state, Middlesex county
	110	Hawaii	5		CountyHunter [▼]			
	291	United States	3068		FIPS 6-4 [▼]			
Russian Districts	15	Asiatic Russia	2644 (+ 180 deleted)	RDA	RDA List	Tamboy award group ("TAG")	<Two-letter Oblast abbr> DASH <Two-digit district code>	AB-01 → Aginsky Buryatsky Autonomous Okrug,Aginsky Area
	54	European Russia						
	61	Franz Josef Land						
	126	Kaliningrad						
	151	Malyi Vysotski Is						

USACA (the **USA Counties Award**) ▼ is the only secondary award supported in Logger32 by default



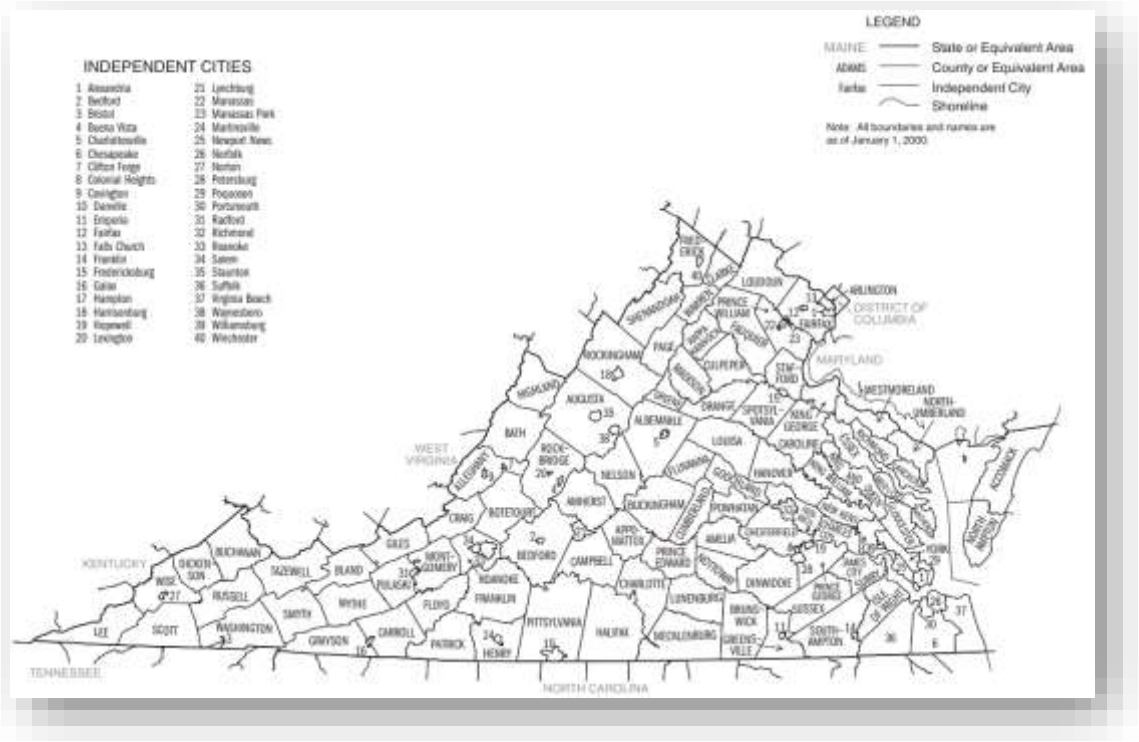
but again, don't fret: we can define awards based on contacting similar Secondary administrative subdivisions in other countries, if we like.

We *could,* for instance, update Logger32's list of USACA counties to include Alaskan boroughs ▼



Copyright © 2026 Robert C. Furzer K4CY

…. and the independent cities of Virginia ▼



… but they are *not* accepted for the award in fact. Alternatively, use Logger32's CNTY-equivalence function to map the Alaskan boroughs and [most] independent cities confirmed through LoTW to the corresponding counties accepted for the USACA award.

18.2 Award tracking and reporting

18.2.1 Confirmation

Whereas a few awards rely purely on unilateral claims or assertions that applicants have made the required contacts, most require some form of confirmation from the QSO counterparties – often with details such as their callsign (of course!), location (DXCC entity, island name, SOTA reference *etc*.), band/frequency, mode and (for QRP awards) power output.

Five QSO states are used in Logger32, according to whether you have contacted a given entity and if so its confirmation status for the awards you are tracking:

1. **Not yet worked**: this will be an **A**ll **T**ime **N**ew **O**ne *when* you make and log a QSO.

> **Hinson tip**: notice I wrote *when*, not *if*. I'm a patient and optimistic DXer. Trust me, CE0X and EZ *will* be contactable some day. A scientist or engineer working on Kure or Johnston islands *will* get permission to make a few QSOs in her downtime. A DXpedition *will* eventually make it through the roaring forties, climbing the icy cliff faces to activate Bouvet Island once more. Keep the faith! Stay positive! Get ready for ATNO day!

2. **Worked**: at least one QSO with an entity that would be accepted for an award has been logged. Congratulations, you've cleared the first hurdle …

Copyright © 2026 Robert C. Furzer K4CY

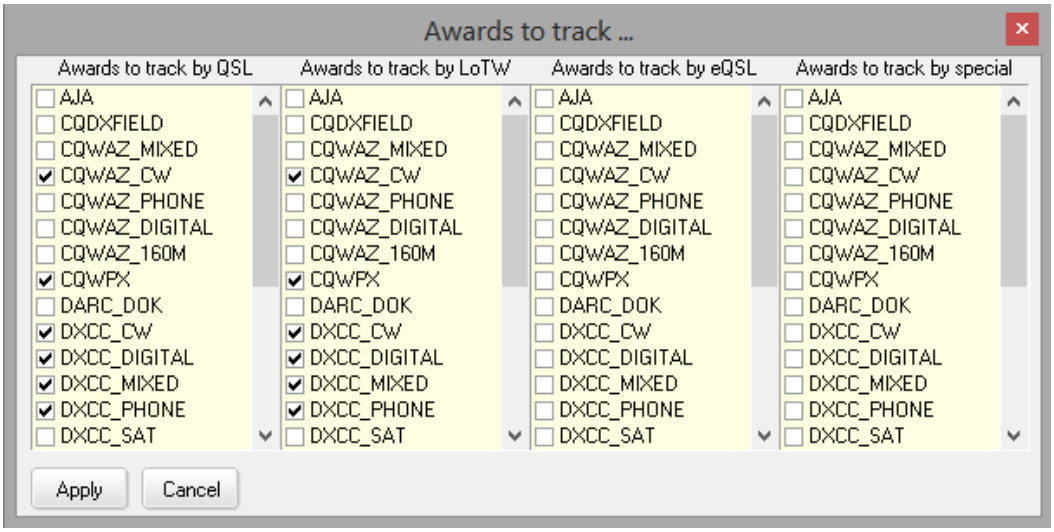
3. **Confirmed**: one or more logged QSOs with the entity have been confirmed to the assurance level necessary to satisfy the award sponsor. Confirmations come in four flavors:
   - **QSL cards** – postcards or letters from the contacted stations, which usually need to be validated by the sponsor or their authorized card-checkers in order to be accepted as proof.
   - **LoTW** - where both parties have digitally signed and uploaded QSO information with matching details such as callsigns, date, time, band and mode, to LoTW.
   - **eQSL** – a QSO-matching system like LoTW but with lower assurance (more readily spoofed, less trustworthy).
   - **Special** – this category caters for other forms of confirmation accepted by a few awards. The IOTA award, for instance, allows valid QSOs in the IOTA contest log of an IOTA island station (following adjudication) to be claimed by the contacted stations for their IOTA awards.

   Various types of confirmation may or may not be allowed and combined according to the rules of a specific award *e.g.* the DXCC award scheme accepts validated original QSL cards *and* LoTW confirmations but *not* eQSLs or photocopied/scanned/emailed cards.
4. **Submitted**: appropriate proof confirming the logged QSO (*e.g.* a QSL card or LoTW matched QSO record) has been sent in for adjudication on behalf of the award sponsor. Once it passes the adjudication process, it becomes …
5. **Granted**: the QSO qualifies for and has been credited towards the award. As far as that award is concerned, you have crossed the finishing line … but remember there are hundreds more awards on offer! Individual QSOs generally qualify for multiple awards.

18.2.2 Select which awards to track

You are unlikely to be interested in *all* the awards out there, especially given the effort and processing required to chase, track and apply for numerous awards[302]. If Logger32 has to track a large number, you may suffer noticeable delays when logging QSOs, updating statistics *etc*. as well as being constantly distracted by the highlighting and alarms for new ones.

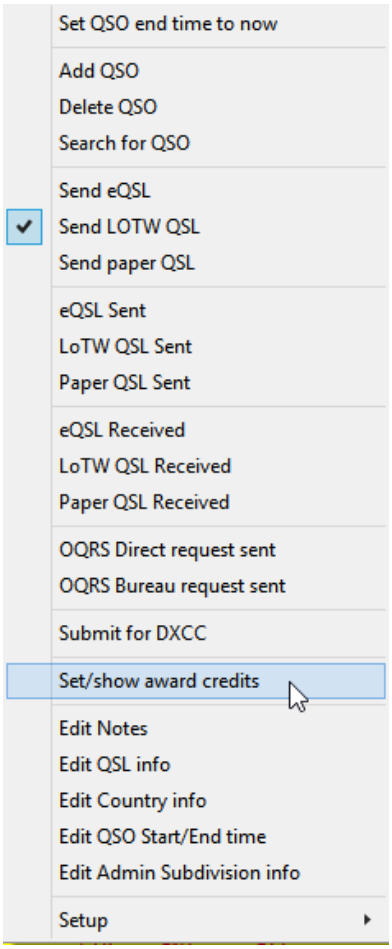
You can elect which awards to track using **Tools ⇨ Awards setup ⇨ Awards to Track ▶**



[302] Have fun, sure, but don't be too greedy! Focus, grasshopper, focus.

Copyright © 2026 Robert C. Furzer K4CY

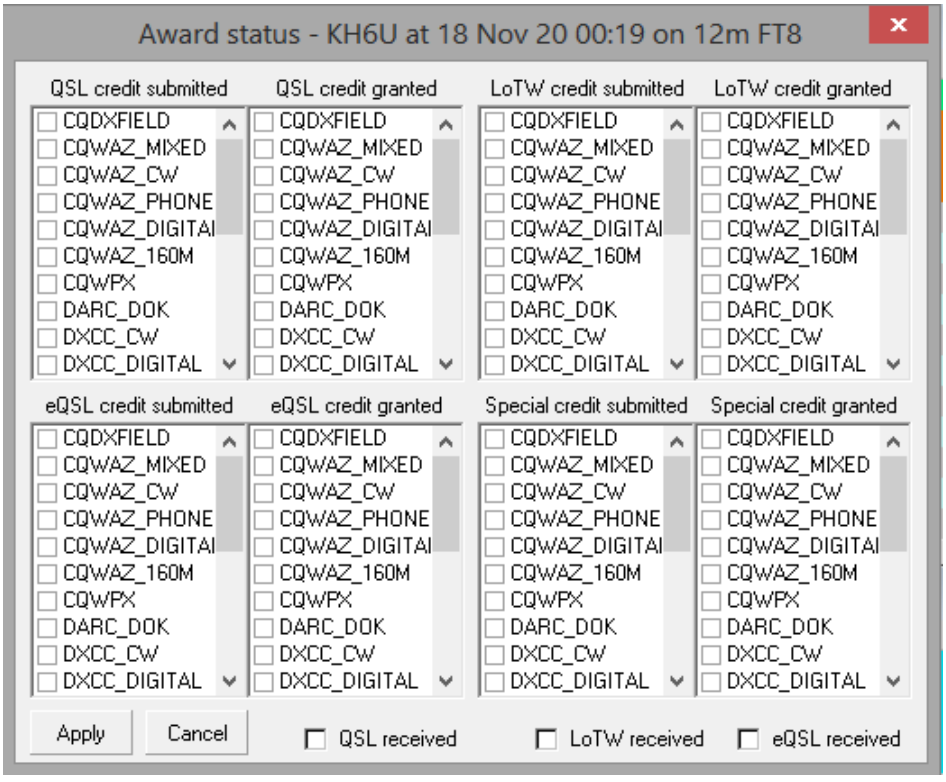
Click to tick the awards under one or more of the columns, depending on what forms of confirmation are required by the award rules.



Right-click a QSO in the logbook for this menu ▶

18.2.3 Crediting QSOs to awards

Click <**Set/show award credits**> to display the award status for the selected QSO ▶



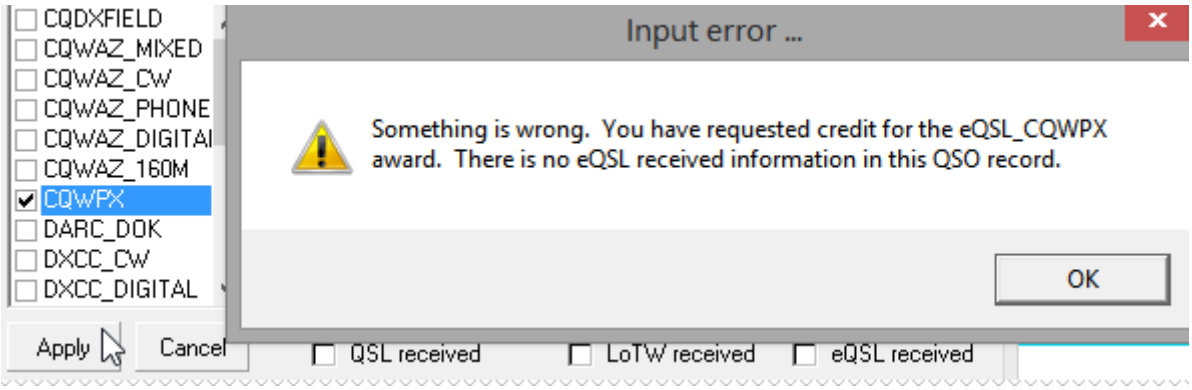
The award status form gives us the opportunity to specify if, for the selected QSO, any of four types of confirmations have been submitted towards any of the awards listed, and whether any have been granted.

Notice along the bottom of the form ▶ we can tick to record receipt of a QSL card, or LoTW or eQSL confirmations, for the QSO. This is one of a few places in Logger32 where we can update our QSL records.

Copyright © 2026 Robert C. Furzer K4CY

Click to tick the relevant options then click <**Apply**>.

Basic error-checking is performed at this stage. For example, an error message is displayed if we try to submit or credit a QSO without any form of confirmation (except 'special') ▼

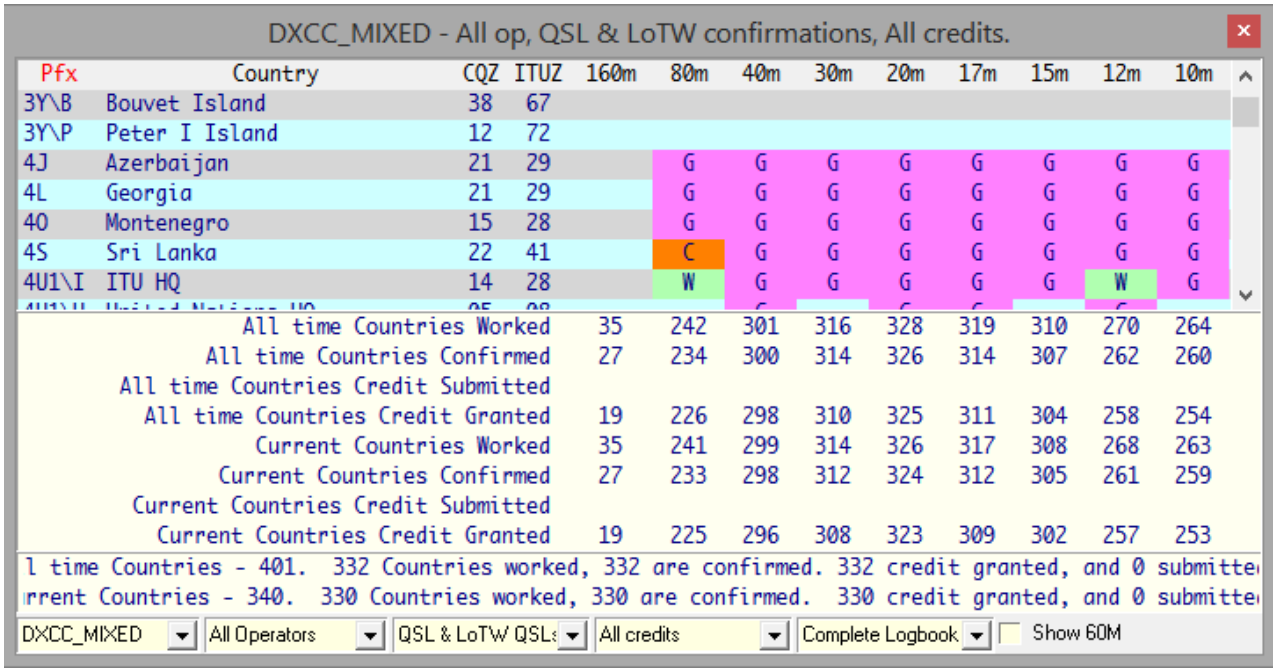


Logger32's file import/export functions handle the basic ADIF-specified GRANTED and SUBMITTED fields no problem, although standard ADIF does not specify the type of confirmation (*i.e*. QSL cards, LoTW and eQSL confirmations, and Special confirmations such as QSOs found in an IOTA station's official log). Therefore, Logger32 uses additional APP_LOGGER32_GRANTED and APP_LOGGER32_SUBMITTED fields in ADIF files.

Hinson tip: rather than simply ignoring the standard fields, importing an ADIF log from logging software other than Logger32 drops any GRANTED or SUBMITTED contacts into the Special confirmation category. We may then update the QSL status through Logger32.

18.2.4 Awards tables

Here is part of a typical awards table for the ARRL DXCC Mixed award ▼



Copyright © 2026 Robert C. Furzer K4CY

Each band/country (or whatever) slot in the body of the table can show a color-coded[303] **W**orked or **C**onfirmed marker. For some specific awards (*e.g.* DXCC), **S**ubmitted or **G**ranted markers can also be shown using information downloaded from the relevant awards websites.

Clicking any non-empty cell on an award table opens the generic QSOs pane showing the corresponding QSO/s.

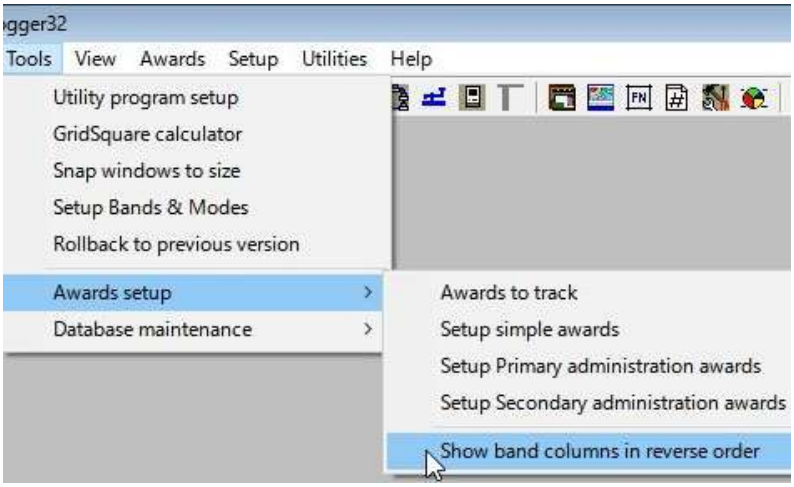
The totals shown in the pale yellow summary section at the bottom of the award tables include the band subtotals[304].

Hinson tip: notice once again the selector boxes at the very bottom of the report screen. Options/filters such as single-mode awards and various forms of confirmation can be selected here, updating the statistics accordingly. Any selections here are saved and applied automatically when you close and next open this report, so **if you find your awards table is surprisingly empty and the totals look wrong, check at the bottom for any inappropriate selections.** For example, the award reports can cover the <**Complete Logbook**> (meaning all time) or just specific years. You may have been lucky enough to work, say, P5 North Korea … but probably not this year, so if you have selected the current year option, your invaluable P5 QSO will not show in the report. Change the selector to <**Complete Logbook**> and it will magically reappear. *Phew!*

Hinson tip: for mode-specific awards, the relevant modes must all be listed at least once in your Bands & Modes table in order for Logger32 to accumulate the statistics. If you are aghast to discover that logged QSOs on a legacy mode you no longer use are not shown as expected for an award, double-check that the mode is still listed somewhere in Bands & Modes.

Some awards tables are sortable by specific (indexed) columns, in which case the column header text turns red *e.g.* Pfx in the DXCC report above. To sort by a different column, simply click the column heading. For convenience, the chosen sort order is saved when you close the report, and re-applied automatically the next time you open it.

If you wish, the left to right sequence of the band columns can be reversed using **Tools ⇨ Awards setup ⇨ Show band columns in reverse order ▶**

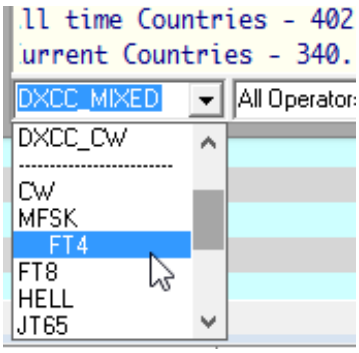


Statistics can be broken down by mode for some awards. Configure the modes and submodes in your Bands & Modes table.

[303] To change the background colors, read about highlighting.

[304] For the DXCC awards, the all-time and current totals are shown, respectively counting or ignoring DXCC entities that have been deleted from the DXCC list *after* we contacted them. Contacts with deleted entities count for most DXCC awards but not the DXCC Challenge.

Copyright © 2026 Robert C. Furzer K4CY

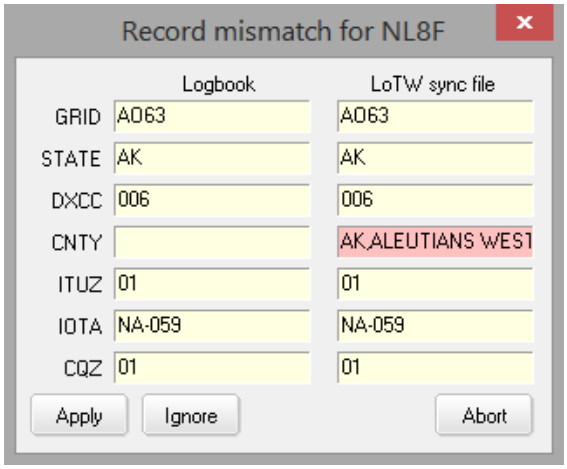


◀ If you use *sub*modes (such as FT4, PSK31 and PSK63), additional options (such as MFSK and PSK) appear on Logger32's mode drop-down list at the bottom left of the report.

The "PSK" option, for instance, aggregates PSK31 *and* PSK63 QSOs … or your statistics for those submodes can be reported individually: your choice.

Hinson tip: since the DXCC rules specify that only CW contacts made *since* 1974 are eligible for the DXCC CW single-mode award, any *prior* CW contacts are not shown in the DXCC CW award table and statistics, even if they have been confirmed. They still count for the mixed-mode DXCC though.

18.3 Administering and customizing awards



As initially installed and configured, Logger32 does not recognize the Alaskan boroughs, generating errors during the LoTW synchronization process for Alaskan stations who confirm their boroughs *e.g.* ▶

"AK, Aleutians West" is one of the Alaskan boroughs ('Secondary Administrative Subdivisions') defined thus in a table in the current ADIF specification ▼

Enumeration for U.S. Counties DXCC Entity Code 6 (Alaska)			
Code	Secondary Administrative Subdivision	Alaska Judicial District	Deleted
AK,Aleutians East	Aleutians East	Alaska Third Judicial District	
AK,Aleutians Islands	Aleutians Islands	Alaska Third Judicial District	Deleted
AK,Aleutians West	Aleutians West	Alaska Third Judicial District	
AK,Anchorage	Anchorage	Alaska Third Judicial District	

Similarly, the independent cities of Virginia qualify for USA Counties Award but are not recognized by Logger32 as valid Secondary administrative subdivisions, by default.

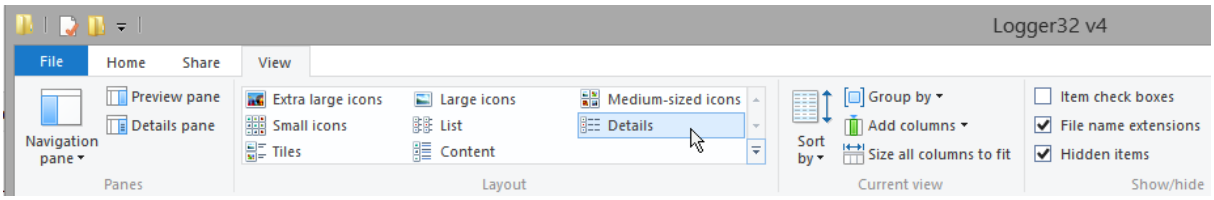
So, how do we go about updating Logger32 accordingly? Using this as a worked example, follow the steps below …

18.3.1 Manually updating Secondary administrative subdivisions

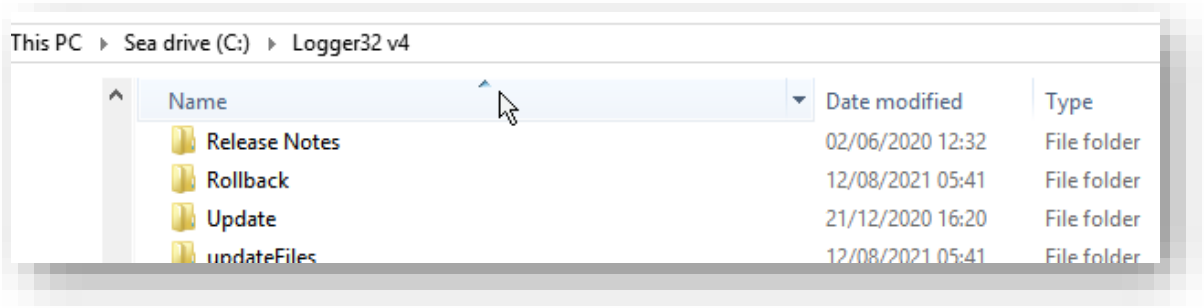
1. Just in case something goes horribly wrong, first make an insurance copy of the secondary administrative subdivision database from *C:\Logger32* so you can easily revert to the current configuration. Here are the instructions in excruciating detail:
   - Close Logger32, so that none of its database files are held open.
   - Open File **E**xplorer using <**Win**+**E**> (while pressing the key printed with the Windows logo, tap the **E** key).

Copyright © 2026 Robert C. Furzer K4CY

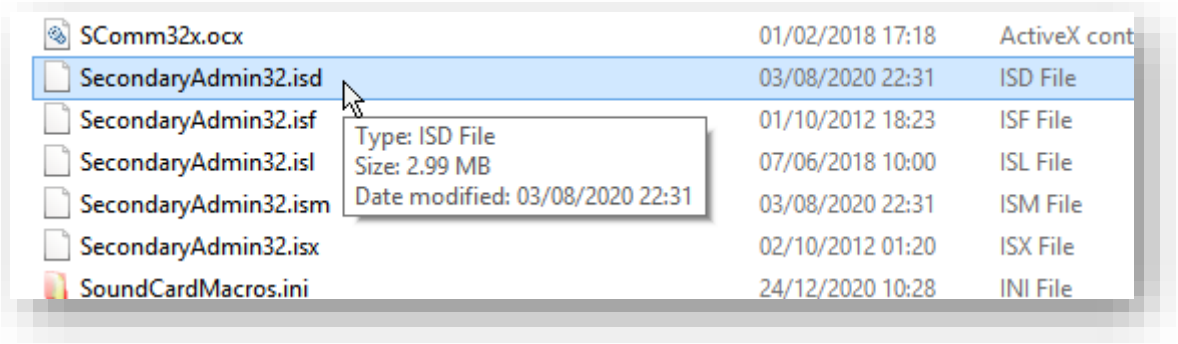
- Click to select the C: drive on the left pane to list its folders and files.
- Double-click to open the *Logger32* folder from the list on the right pane.
- Under File **E**xplorer's **View** menu, select **Details** to list the files in detail ▼



- Click the header area for the **Name** column once or twice[305] to sort the folders and files alphabetically by their names ▼



- Scroll down the file list and click to select the first of the *SecondaryAdmin32* database files holding the secondary administrative subdivision data (5 files on my system) ▼



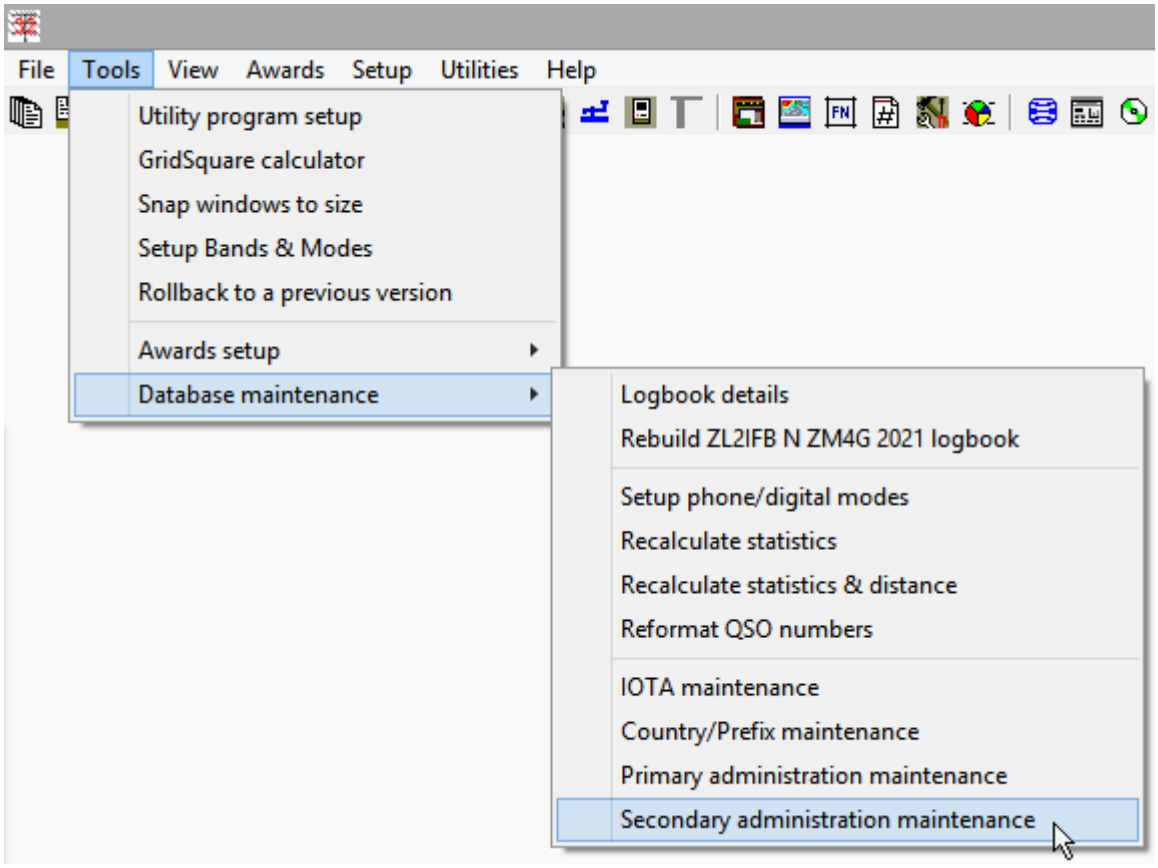
- Shift+click the last of the *SecondaryAdmin32* files to select all of them.
- If you *know* you will, or think you might possibly want to edit the primary administrative subdivisions, control-click those database files as well to add them to the selection.
- Press <**Ctrl**+**C**> to copy the file details.
- Either create a new folder to save the backups or navigate to an existing folder.
- Press <**Ctrl**+**V**> to paste the file details and so save copies of the files to the backup folder.
- Exit/close File **E**xplorer by clicking the X in the top right corner of its window.

2. Re-start Logger32, safe in the knowledge that you have those backups.

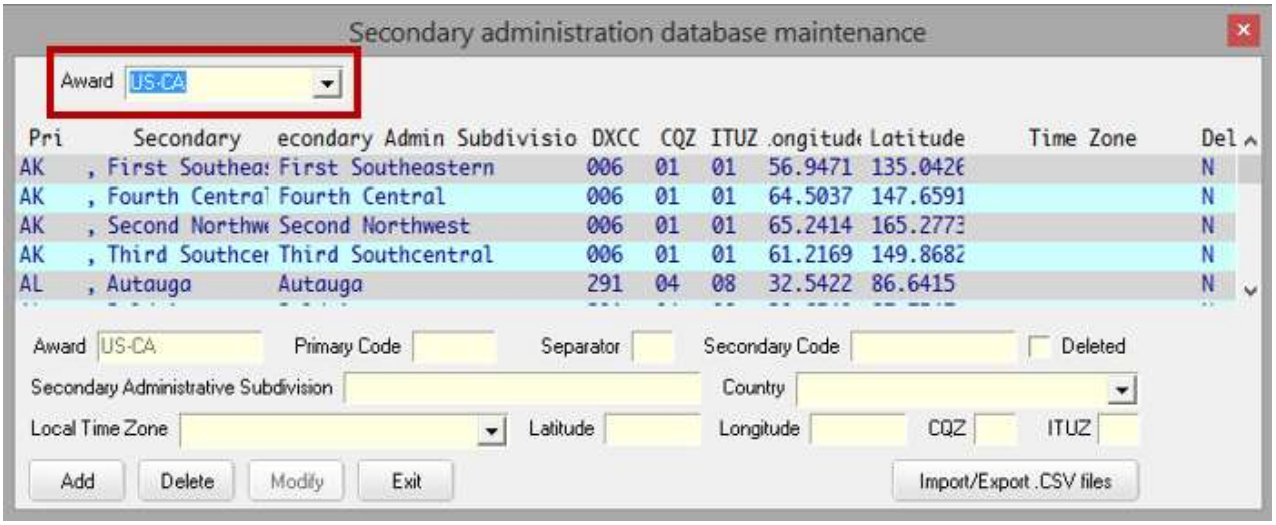
[305] This is a toggle: successive clicks change between ascending and descending order, inverting the tiny arrowhead in the sorted column's header.

Copyright © 2026 Robert C. Furzer K4CY

3. To update the Secondary administrative subdivisions, open **Tools** ⇨ **Database maintenance** ⇨ **Secondary administration maintenance** ▼



4. Click to pick "US-CA" from the drop-down list of awards in order to access the secondary administrative subdivisions currently defined for the USACA award ▼



5. To add a new Secondary administrative subdivision (*e.g.* the Alaskan borough of Aleutians West):
   - <**Tab**> to or click on the <Primary Code> yellow data-entry field and type *AK* <**Tab**>. AK is the "code" for Alaska, which is the Primary administrative subdivision within which are the Secondary administrative subdivisions we are going to update.

Copyright © 2026 Robert C. Furzer K4CY

- As shown in the ADIF specification "Code" column, the separator between Primary and Secondary administrative subdivisions is a comma (*e.g.* "AK, Aleutians West"), so type a comma, then <**Tab**> to the next field.
- Since the ADIF specification for Alaska does not list any Secondary Codes, type in the name of the Secondary administrative subdivision in full *i.e*. *Aleutians West* <**Tab**>.
- <**Tab**> again to skip the Deleted field[306].
- In Secondary Administrative Subdivision, type *Aleutians West* <**Tab**>.
- For the country, either select *Alaska* from the drop-down list or type *Alaska* (being careful not to mis-type it) and <**Tab**>.
- You can either look-up and complete the Local Time Zone for Aleutians West *i.e*. the normal offset from UTC in hours – positive values for countries to the West of the Greenwich meridian, negative values for countries to the East, ignoring daylight savings time – or leave it blank[307]. <**Tab**> to move on.
- Enter the latitude of the approximate geographic center of the Aleutians West region in degrees and decimal degrees – positive values for North of the equator or negative for South[308] *e.g.* 49 <**Tab**>.
- Likewise enter the longitude of the approximate geographic center of the Aleutians West region in degrees and decimal degrees – positive values for West of the Greenwich meridian or negative for East *e.g.* 170 <**Tab**>.
- Enter the CQ zone for Aleutians West *i.e*. 1 <**Tab**>.
- Enter the ITU zone for Aleutians East *i.e*. 1 <**Tab**>.
- Check for any typos, correct the record if necessary, then *click the <**Add**> button* to commit (save) the new Aleutians West entry to the Secondary administrative database.
- Move on to enter the next Secondary administrative subdivision – including the independent cities of Virginia – in the same way, until they are all done.

6. To delete a Secondary administrative subdivision completely, first click to select the relevant row from the table, then click the <**Delete**> button at the bottom of the form.

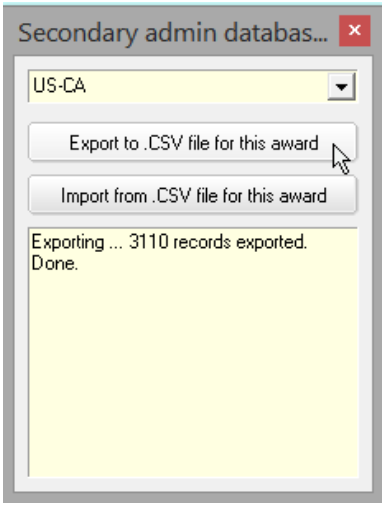
[306] Administrative subdivisions that have been removed from the list and are no longer valid for future QSOs can be marked *Deleted* without affecting any applicable QSOs logged previously.

[307] Logger32 can use the time zone value to calculate and display the local time for a DX station you are working. If no value is entered here, Logger32 uses the Primary administrative subdivision's time zone value (if defined), otherwise the time zone for the DXCC entity.

[308] The latitude and longitude values here are used in a similar way to the time zone, for greater precision when calculating bearings and distances to the applicable stations. If not stated, Logger32 uses the Primary administrative subdivision or DXCC entity values for coarser estimates. Logger32 uses a DX station's grid square by preference, if that is known.

Copyright © 2026 Robert C. Furzer K4CY

7. Ideally while the editing form is still open, click <**Import/Export .CSV files**> to save the updated information to disk in a form that can be shared with other Logger32 users:
   - Check and if necessary select the relevant award *i.e.* "US-CA" in the top drop-down selector box ▶
   - Click <**Export to .CSV file for this award**>.
   - Navigate to a suitable folder, check/update the file name and click <**Open**> to generate and save the *.CSV* file.
   - Click the corner X to close the import/export pane.
   - Don't forget to check and preferably share the *.CSV* file with others *e.g.* by uploading it to the Files area of the Logger32 reflector and sending a note to the reflector about it, perhaps with a hint to read the next section of this User Manual.



> **Hinson tip:** a pragmatic way to resolve various ambiguities in the official ADIF specification for some Secondary administrative subdivisions is to refer to the current *config.xml* file used by TQSL. For instance, the US Census Bureau map cited as a reference for the USA Counties Award shows "E. Baton Rouge", whereas the county name is enumerated in full, in capitals, as "EAST BATON ROUGE" in TQSL's *.xml* file, hence that is how it is represented in any LoTW confirmations that you receive (assuming users are using the current TQSL configuration, anyway!).
>
> **Hinson tip:** the Primary and Secondary administrative subdivision maintenance functions in Logger32 are awkward to navigate and update, particularly for multiple updates. I find it much easier to export, edit, save and import the *.CSV* files using a text editor or Excel.
>
> **Hinson tip**: if you are using Excel to check and edit the *.CSV* files, the data can be formatted with leading zeroes where relevant: click the column header to highlight the whole column; right-click and select <**Format Cells**>; select the <**Custom**> format; set the <**Type**> to 00 (for two digits) or 000 (for three digits) and click <**OK**>. If you are using a plain text editor such as Notepad, simply type in the requisite number of leading zeroes.
>
> **Hinson tip**: if it catches the beady eye of Bob and the Logger32 beta crew, the *.CSV* may be uploaded to the Logger32.net website and/or included in future Logger32 auto-updates. On behalf of Logger32 users, thank you for generously maintaining and sharing the *.CSV* file.

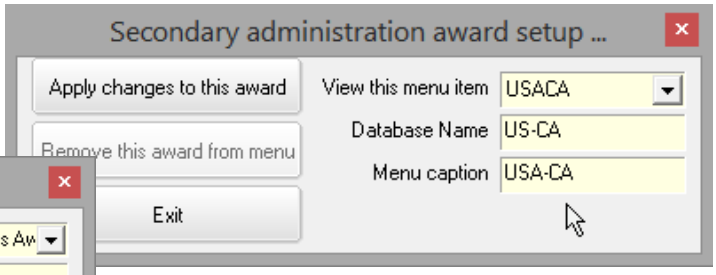
8. If you are all done, click <**Exit**> to end the Secondary administrative subdivision maintenance … or move on to maintain the next award's details in the same manner.

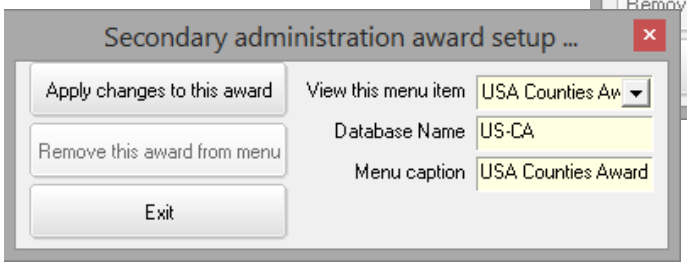
18.3.2 Renaming an award

The USA Counties Award is variously known as "US-CA" and "USACA" in Logger32. If little inconsistencies like this give you sleepless nights, you can correct the award caption through **Tools ⇨ Award setup ⇨ Setup Secondary administrative subdivision awards**.

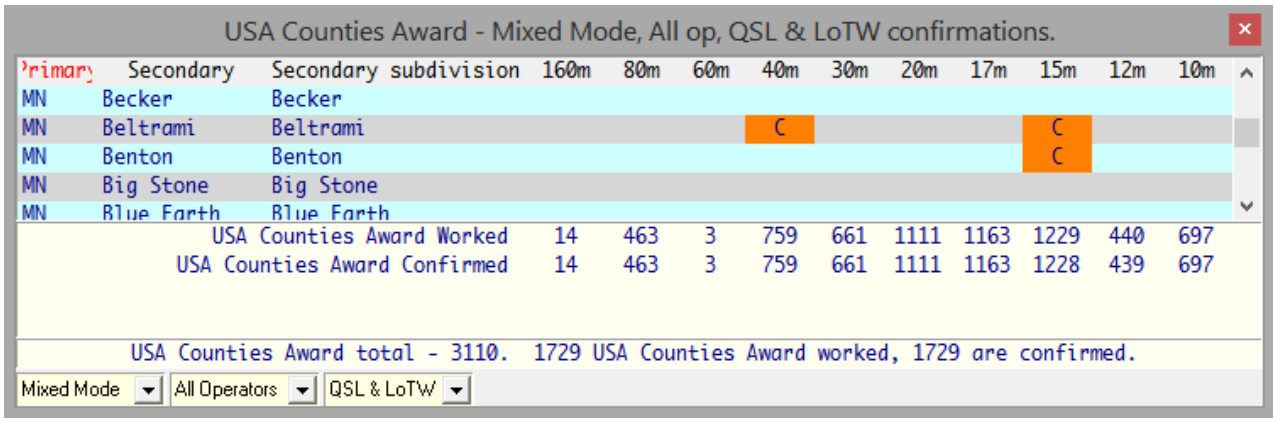
Copyright © 2026 Robert C. Furzer K4CY

Select the award from the <**View this menu item**> drop-down then edit the **Menu caption** text[309] ▶





◀ Click <**Apply changes to this award**> to see the edited caption appear in the <**View this menu item**> and the award table ▼



Sleep well!

18.3.3 Automatically updating Secondary administrative subdivisions

If some other kind soul has shared a new/updated *.CSV* file for an award that interests you, complete this process to update your Logger32 system accordingly:

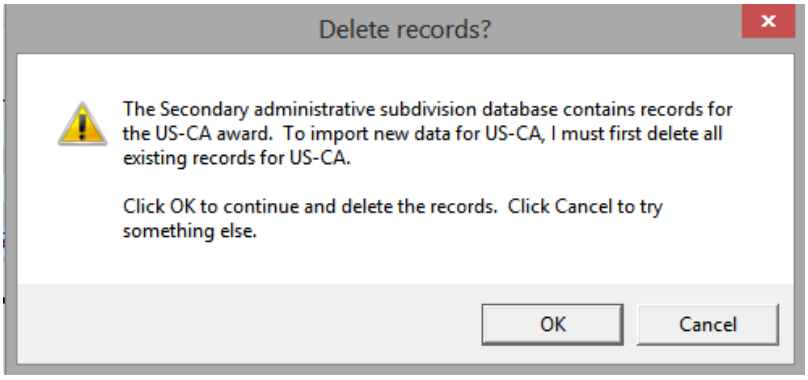
1. Download the *.CSV* file to a suitable folder on your PC – perhaps a new folder under *C:\Logger32* just for administrative subdivision *.CSV*s *e.g. C:\Logger32\Admin subdiv CSVs*
2. Preferably, open the *.CSV* file in a spreadsheet program[310] to check it through and decide whether to import it into your system. If it appears incomplete, out of date or otherwise poor quality, give it a miss.
3. If you decide to go ahead with the update, **make a backup** of your primary and/or secondary administrative subdivision database files by following steps 1, 2 and 3 in the previous section just in case things don't go to plan. Skip this step at your peril.
4. Click **Tools ⇨ Database maintenance ⇨ Secondary administration maintenance**.
5. Click <**Import/Export .CSV files**> to open the import/export function.
6. Select the relevant award from the drop-down list of awards.

[309] Do not be tempted to 'correct' the <**Database Name**> from US-CA as that is used internally by Logger32 to manage the award, associating the award with its secondary administrative subdivisions. If you change the <**Database Name**>, the award table loses its data. Put the original name back to recover.

[310] You *can* open and read *.CSV* files in a plain text editor such as Notepad … but the format is not ideal for the Mark I Eyeball. Nevertheless, it's worth a look. Gross errors will hopefully stand out.

Copyright © 2026 Robert C. Furzer K4CY

7. Click <**Import from .CSV file for this award**>.

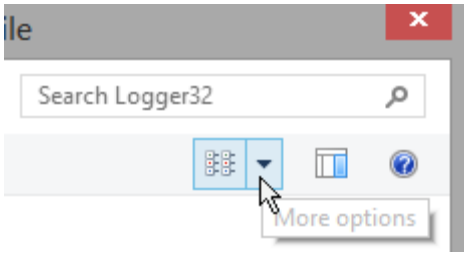


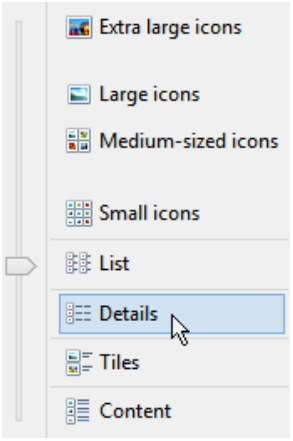
Logger32 immediately warns you that it will delete any existing stats for this award ▶

Read and consider the warning. It is not as scary as it may seem as it's no big deal to delete and recreate your award statistics. If you have run the recalculate statistics function before, Logger32 will have deleted and regenerated its statistics for *all* the awards. So, click <**OK**>.

8. Navigate to *C:\Logger32\Admin subdiv CSVs* and click to select the newly-downloaded *.CSV* for the award, then click <**Open**> to start importing the selected file.

Hinson tip: in File Explorer, you can search for files in the current folder and sub-folders using the search box at the upper right of the window but it's often easier to select the <**Details**> view[311] from the pull-down options ▶

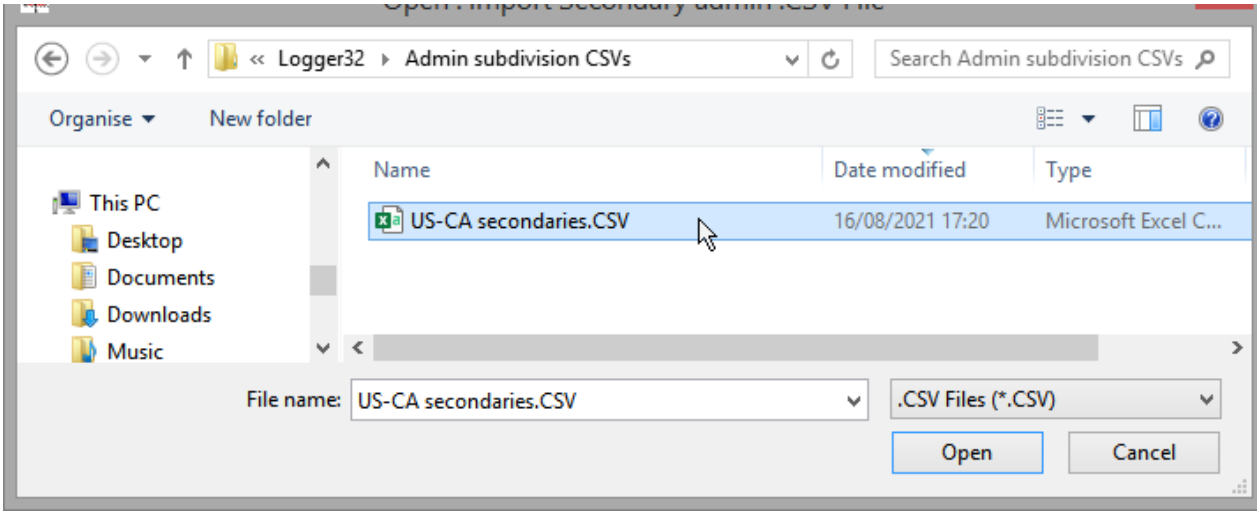




◀ Click and drag the pointer to preview the different views and let go to pick the previewed one, or simply click <**Details**> to go straight there.

On the details folder view, click the <**Name**> column header to sort the files by name and look for the *.CSV* you downloaded and saved, or click to sort on the <**Date**> column header, given that the downloaded *.CSV* file is probably the most recent file in the folder.

Find the file, click to select it, then click the <**Open**> button to, errrr, open it ▼



[311] Windows generally recalls the view you used last in each folder. For folders you haven't viewed yet, it picks a view according to the types of file they contain *e.g.* icons for folders mostly containing images.

Copyright © 2026 Robert C. Furzer K4CY

9. The import process takes a moment to ▶
   - Delete the current statistics for the selected award;
   - Import the award criteria from the *.CSV* file into the database;
   - Regenerate the statistics for the selected award by finding applicable QSOs in the open logbook, checking their details (*e.g.* which – if any - secondary admin subdivisions are identified) and updating the stats accordingly (*e.g.* whether the QSOs have been confirmed by LoTW, QSL cards *etc*.).

   The progress bar progressively shows the process progressing from partial to completion.



10. [*Optional*] Recalculate your statistics: scanning your log and regenerating the worked/confirmed statistics ensures that the records are complete and up to date … but it can take a wee while to process a large log on a slow PC, so you may want to defer this until a better time when there is no DX to work – perhaps overnight (if you are not an LF DXer) or around noon (if you are).
11. Click the corner X to close the import/export function and that's it, all done. Carry on chasing the DX and hopefully earning those awards!

18.3.4 Tracking additional awards

In addition to the awards already included in Logger32, there are *hundreds* more – *thousands* in fact if you count variants such as single band or mode awards, newcomer/novice awards *etc*. plus dormant/expired/legacy awards that are no longer actively promoted and awarded.

The Logger32 support team cannot reasonably cope with such variety and the pace of change in award schemes … so we don't try. Instead, Logger32 gives us the facilities to select, define and track our progress towards whichever operating awards catch our imagination[312].

18.3.5 Setting up a Primary administrative subdivision award (worked example)

Although as far as I know there is no "**W**orked **A**ll **C**hina **A**ward" based on contacting the Primary administrative subdivisions (provinces *etc*.) of mainland China, let's *imagine* there is one and work systematically through the entire process of setting up WACA in Logger32.

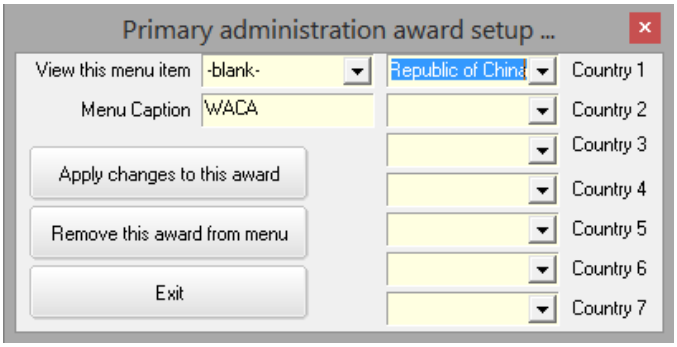
Hinson tip: this *is* a complicated process. Most Logger32 users will never need to define new awards … but some will, and hopefully they will find this worked example with explicit instructions valuable. I suggest *reading* the whole thing first so you know what you are about to do, then following the instructions line-by-line to do it. You might like to display this section of the manual on a different screen or laptop beside your Logger32 system, or print out the next 4 pages.

[312] Logger32 does not support awards involving *several* but not *all* DXCC entities, such as **W**orked **A**ll **E**urope and **C**ommonwealth **C**entury **C**lub. The maximum number of DXCC entities that Logger32 will accept for a Primary administration award is just seven, due presumably to the underlying database architecture and software design decisions. Perhaps it's Bob's lucky number.

Copyright © 2026 Robert C. Furzer K4CY

Step 1: define the award

- Open **Tools ⇨ Awards setup ⇨ Setup Primary administration awards**.
- Click to select one of the *-blank-* lines in the drop-down list next to <**View this menu item**> ▶
- Click to select the *Peoples Republic of China* line in the drop-down list for <**Country 1**>.
- Type a short name for the award into <**Menu Caption**>. Pick a unique name – not WAC since that is the **W**orked **A**ll **C**ontinents award. WACA should work.



- Click <**Apply changes to this award**> to save the new award definition. The *-blank-* line instantly changes to show the award name. Don't skip this step or you'll have to open and complete the form again.
- Having applied the changes, click <**Exit**> or click the corner X to close the form.

Step 2: generate a *.CSV* file for the Primary administrative subdivisions

For this worked example, we presume there is no *.CSV* file available for the Chinese provinces, so you'll have to generate it yourself.

There is currently no enumeration of the Primary administrative subdivisions for China in the ADIF standard[313]. However, Google tells us that ISO 3166-2 lists two-letter codes and the anglicized names for 23 Chinese provinces, five autonomous regions, two special administrative regions and four municipalities ▼

Code	Subdivision name[note 1] (National 1958 = ISO 7098:2015 = UN III/8 1977)	Subdivision name (zh) [note 2]	Subdivision category
CN-AH	Anhui Sheng	安徽省 (Ānhuī Shěng)	province
CN-BJ	Beijing Shi	北京市 (Běijīng Shì)	municipality
CN-CQ	Chongqing Shi	重庆市 (Chóngqìng Shì)	municipality
CN-FJ	Fujian Sheng	福建省 (Fújiàn Shěng)	province

Since both special administrative regions (Hong Kong and Macau) plus the province of Taiwan Sheng have their own callsign prefixes and DXCC entity numbers, we will exclude them from the [fictional] **W**orked **A**ll **C**hina **A**ward. As it happens, the remaining Chinese Primary administrative subdivisions are conveniently specified in TQSL's configuration file *Config.xml,* along with their ITU and CQ zone numbers ▼

```
<field Id="CN_PROVINCE" flags="0" intype="2" label="Province" len="2" type="C">
  <enums>
    <enum value="AH" zonemap="44:24">Anhui</enum>
    <enum value="BJ" zonemap="44:24">Beijing</enum>
    <enum value="CQ" zonemap="43:24">Chongqing</enum>
    <enum value="FJ" zonemap="44:24">Fujian</enum>
    <enum value="GD" zonemap="44:24">Guangdong</enum>
```

[313] Not yet anyway. It has been proposed to add these Chinese provinces *etc*. to the ADIF standard.

Copyright © 2026 Robert C. Furzer K4CY

Retype or copy and edit those 31 lines from *config.xml* into a spreadsheet with 31 data rows plus a header row for the requisite columns as shown here ▼

	A	B	C	D	E	F	G	H	I	J
1	Code	Oblast	Description	CQZ	ITUZ	Prefix	TimeZone	Latitude	Longitude	Deleted
2	AH		Anhui Sheng	24	44					N
3	BJ		Beijing Shi	24	44					N
4	CQ		Chongqing Shi	24	43					N
5	FJ		Fujian Sheng	24	44					N
6	GD		Guangdong Sheng	24	44					N
7	GS		Gansu Sheng	23	43					N
8	GX		Guangxi Zhuangzu Zizhiqu	24	44					N

Hinson tip: some regions of China (*e.g.* GX) span more than one ITU zone, but only one zone can be listed: for these, look at the maps to decide which ITU zone to record as the main/default zone for the region according to the biggest area or the presence of a major city where most amateurs are likely to live. Once a QSO is logged, you can edit the zone if the default is wrong.

Save the spreadsheet to a convenient folder on your hard drive in **C**omma **S**eparated **V**ariable (*.CSV*) format, with a suitable name *e.g. C:\Logger32\CSVs\WACA Chinese primaries.CSV*

Step 3: import the *.CSV* file for the award



- Open **Tools ⇨ Database maintenance ⇨ Primary administration maintenance**.
- Click the <**Export/Import .CSV files**> button to open a form ▶
- Click to select the *Peoples Republic of China* under the drop-down list at the top[314].
- Click <**Import from .CSV file for this Country**>
- A scary warning message may appear at this stage ▼



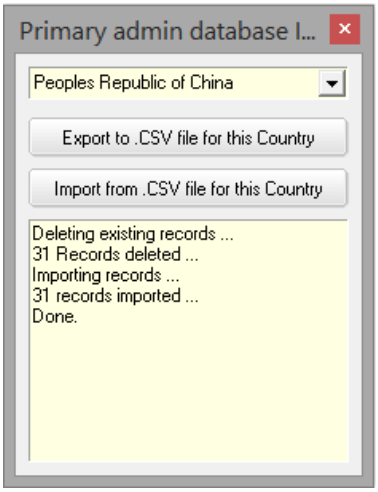
Read the message and if you are brave, click <**OK**> to proceed.

- Locate and click to select *C:\Logger32\CSVs\WACA Chinese primaries.CSV* created in step 2, then click <**Open**> to import the file into Logger32.

[314] If *Peoples Republic of China* does not appear in your list, insert "318 Peoples Republic of China" into *C:\Logger32\ADIFCountriesWithPrimarySubdivisions.txt* between "291 United States of America" and "339 Japan" (all without the speech marks). All the entries in that file **must** be in strict numerical order, otherwise puppies will die.

Copyright © 2026 Robert C. Furzer K4CY

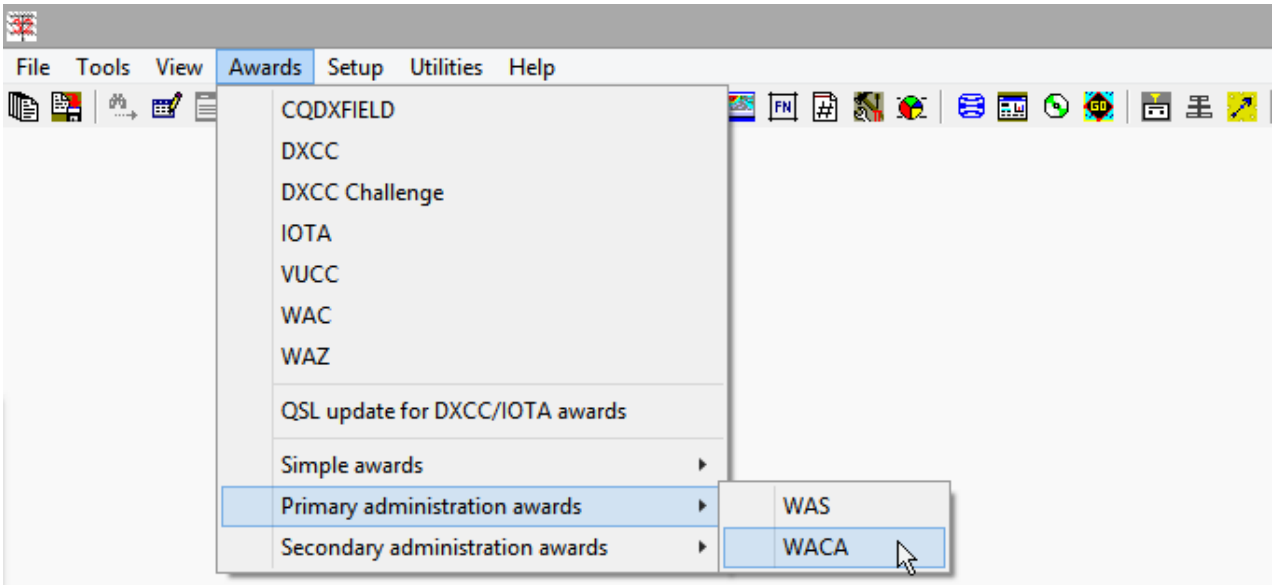
If you have previously configured Primary administrative subdivisions for China, Logger32 will delete the existing records, import the new ones, warn you about any gross errors and conclude with a curt but strangely reassuring "Done" ▶



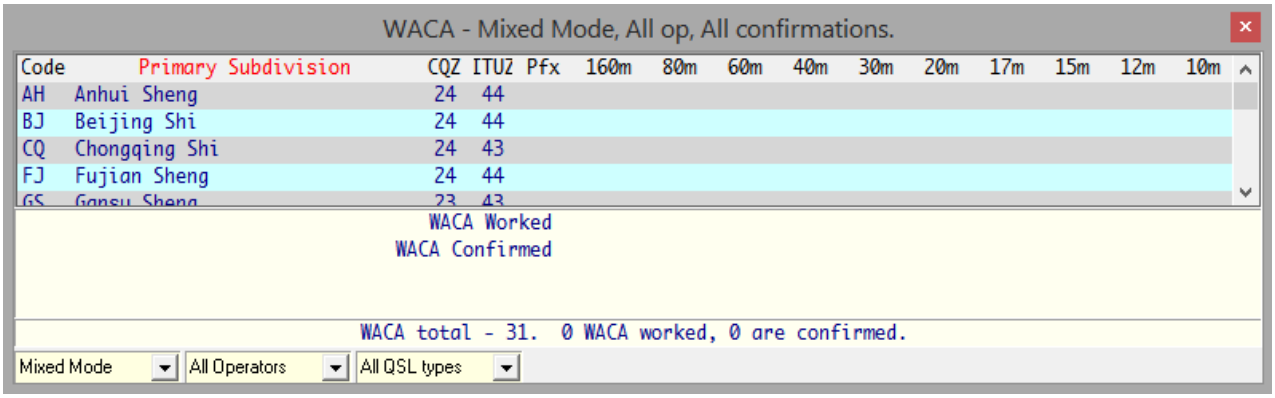
- Click the top-right corner X to close the form.

Step 4 – check the award table

You should now find the shiny new WACA award listed under **Awards ⇨ Primary administration awards** ▼



Clicking it opens the WACA table reporting your status towards this award ▼

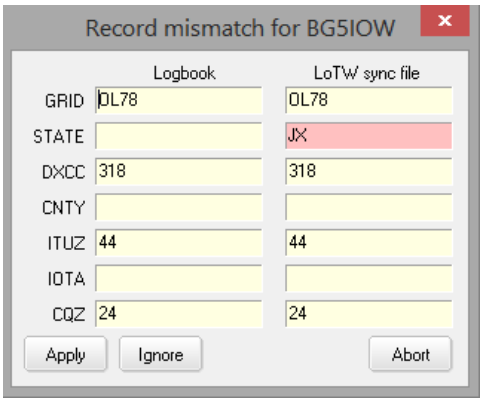


Don't panic! Presently, none of the cells in the WACA table show qualifying QSOs, and the yellow summary/totals area is mostly blank … because you have only just defined the Primary administrative subdivisions, whereas none were recorded previously.

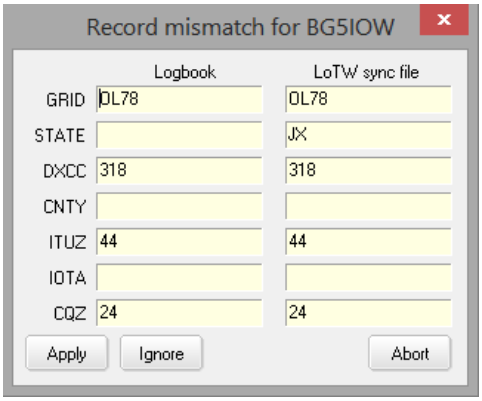
Copyright © 2026 Robert C. Furzer K4CY

Next, download and synchronize a batch of LoTW confirmations, including some Chinese ones with the newly-defined provinces.

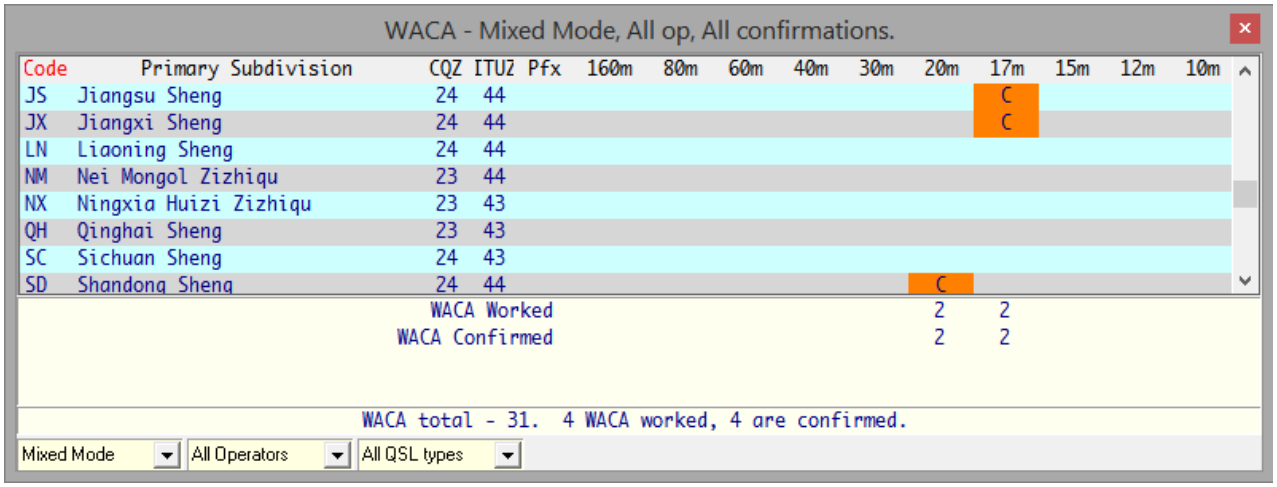
Previously (pre-WACA), Chinese LoTW confirmations with provinces (STATEs) generated pink mismatch errors in the manual LoTW synchronization process ▶



We used to either <**Ignore**> the mismatched records or erase the provinces in order to <**Apply**> these confirmations to the log.



◀ Now, having added WACA to Logger32, we can simply <**Apply**> the LoTW confirmations, and they duly appear as colored **C** blobs on the WACA report ▼



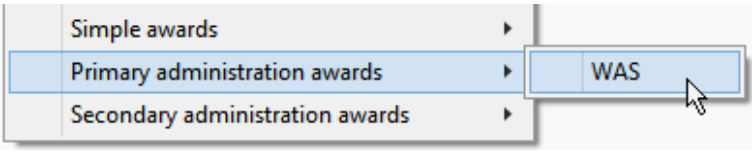
Success! It works!

18.4 Configuring the awards

18.4.1 The database files

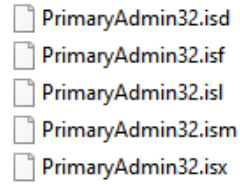
Two databases are provided with Logger32, as a set of nine files installed in *C:\Logger32*:

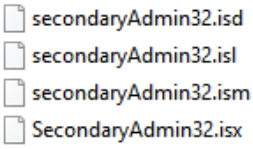
- The **PrimaryAdmin32** database containing details of the "primary administrative subdivisions" consists of five files named *PrimaryAdmin32.isd, .isf, .isl, .ism* and *.isx* By default (when Logger32 is initially installed), this database only contains the primary administrative subdivisions defined in the ADIF specification, such as the Canadian provinces, plus American states that qualify for ARRL's **W**orked **A**ll **S**tates award ▶

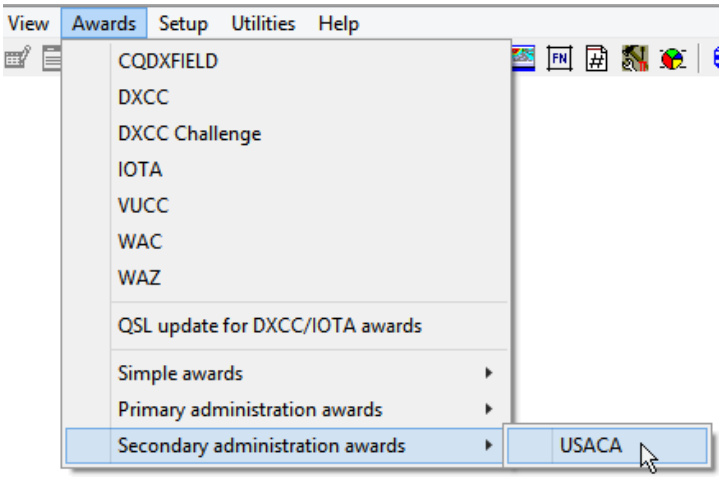


Copyright © 2026 Robert C. Furzer K4CY

- The **SecondaryAdmin32** database consists of four more files named *secondaryAdmin32.isd, .isl* and *.ism*, plus *SecondaryAdmin32.isx* ▶







◀ By default, Logger32's secondary administrative subdivisions database only specifies US counties valid for the **USA C**ounties **A**ward.

If you define additional secondary awards such as JCC, JCG, DOK, NATS, ECA and WAB[315], the award specifications and statistics are added to this database.

CAUTION!

Editing/maintaining the administrative databases and awards?

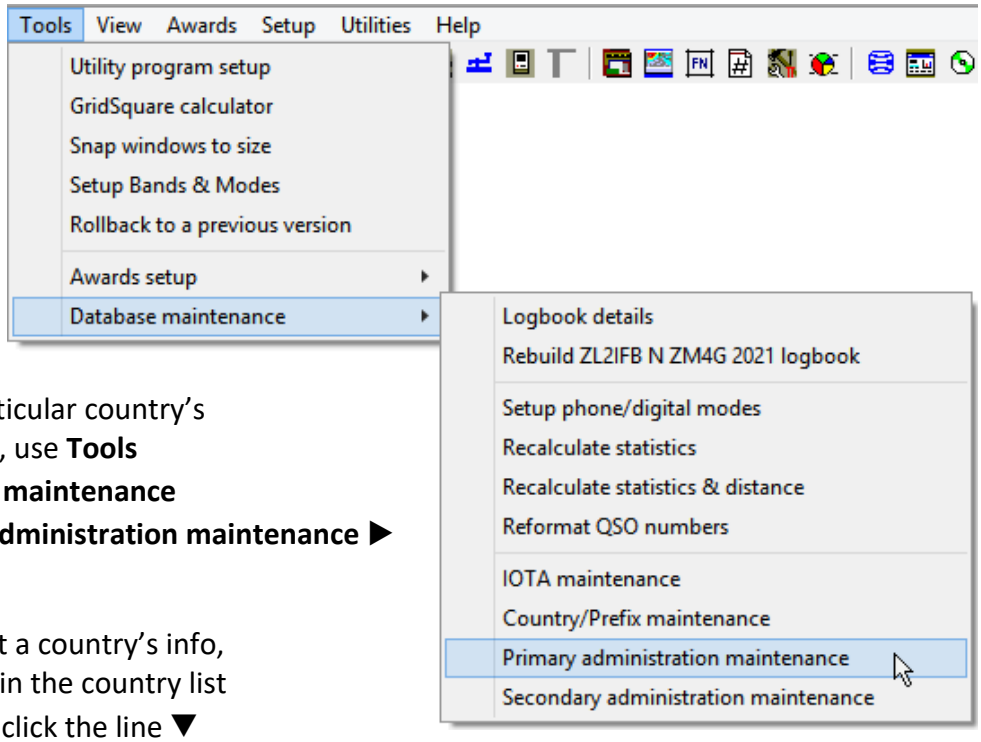
Backup before *and* after!

1. Make a backup copy of the database files *before* editing or maintaining the primary and secondary admin databases. Since you are entering risky territory, you may need to revert to how it is now, either abandoning the changes or starting afresh. Backups stored safely (*e.g.* on a USB memory stick) are your insurance policy, your get-out-of-jail-free card in case of user errors, malware infections, disk failures, cosmic rays and other incidents that corrupt or destroy the databases.
2. The Logger32 installer delivers minimalist primary and secondary databases. Any customization changes that you make to your databases (*e.g.* adding primary or secondary awards, updating provinces) may be overwritten if you reinstall Logger32 with its minimalist databases *unless* you export your custom awards and/or country information (both primary and secondary) into *.CSV* files and save those with your database backups, *after* making the customizations. That way, if Logger32 is reinstalled (even if Logger32's administration databases are restructured at some point), you should still be able import your saved *.CSV* files.
3. It's worth the effort. Trust me, I'm a doctor, a graduate of the School of Hard Knocks.

[315] The **W**orked **A**ll **B**ritain award uses Ordnance Survey's 10km squares that don't correspond to the Maidenhead locators or EU regulations. I blame Brexit.

Copyright © 2026 Robert C. Furzer K4CY

18.4.2 Primary administration (state) data

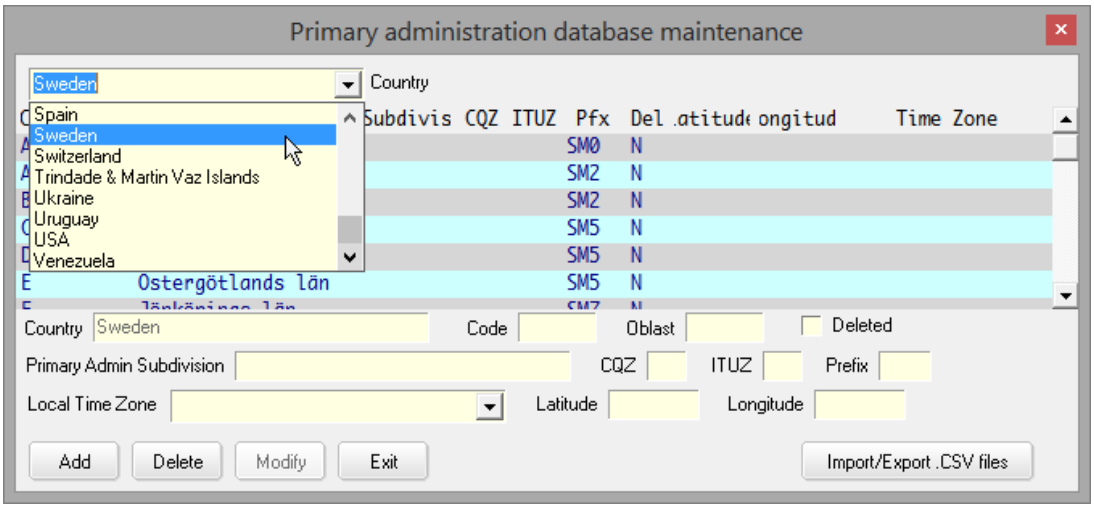


To edit a particular country's primary data, use **Tools**

➪ **Database maintenance**

➪ **Primary administration maintenance** ▶

To edit a country's info, find it in the country list and click the line ▼



A form showing the current data for that country appears, ready for editing. All fields can be edited *except* for the country itself (a key field): select a line and click <**Modify**> to proceed.

Hinson tip: to define a new DXCC entity (country), you can click the blank line at the top of the pull-down country list and enter the new country name, as per the official ARRL DXCC list. On completion, click <**Add**> to save it. However, it is easier and safer to wait for the relevant Club Log update to come through, generally a little before the new entity becomes valid. Be patient, grasshopper.

Copyright © 2026 Robert C. Furzer K4CY



If numerous changes are to be made to a DXCC entity, it may be easier to export the data for that entity to a *.CSV* file.

◀ Select the relevant country from the list then click <**Export to .CSV file for this Country**> and specify a folder and file name.

The *.CSV* file can then be opened and edited in, say, Excel or Notepad, and saved as a *.CSV* file when done.

<**Import from CSV file for this Country**> lets you re-load the edited data back into Logger32's database but first **double check that the correct country is selected to avoid messing up your database.** You do have good backups, right?

Hinson tip: this mechanism lets us share the burden of updates. The *.CSV* files are useful for hams chasing the same awards. So long as some kind soul in the community diligently and accurately maintains the awards and shares the updated *.CSV* files, other Logger32 users can simply import them to stay up-to-date. For instance, if the Swedish government makes administrative changes, someone who knows the details can export, edit and import the *.CSV* file for Sweden to update themselves *and* share the updated *.CSV* file with other Logger32 users (*e.g.* via the Logger32 reflector's Files area) for them to import too.

18.4.3 Administrative subdivision databases

Two internal databases contain the primary and secondary administration details required for various awards in two sets of four files called *PrimaryAdmin32.xxx* and *Secondary Admin32.xxx* where *xxx* is *ISD*, *ISF*, *ISM* and *ISL*. Both databases can be maintained through functions provided in Logger32.

The primary and secondary databases can be exported and if necessary re-loaded from *.CSV* files. **C**omma **S**eparated **V**ariable files are useful:

- As backups.
- To prepare the administrative subdivision criteria used by awards.
- To check and maintain the criteria *e.g.* when administrative areas are renamed or the boundaries are changed.
- To share award criteria with other Logger32 users.

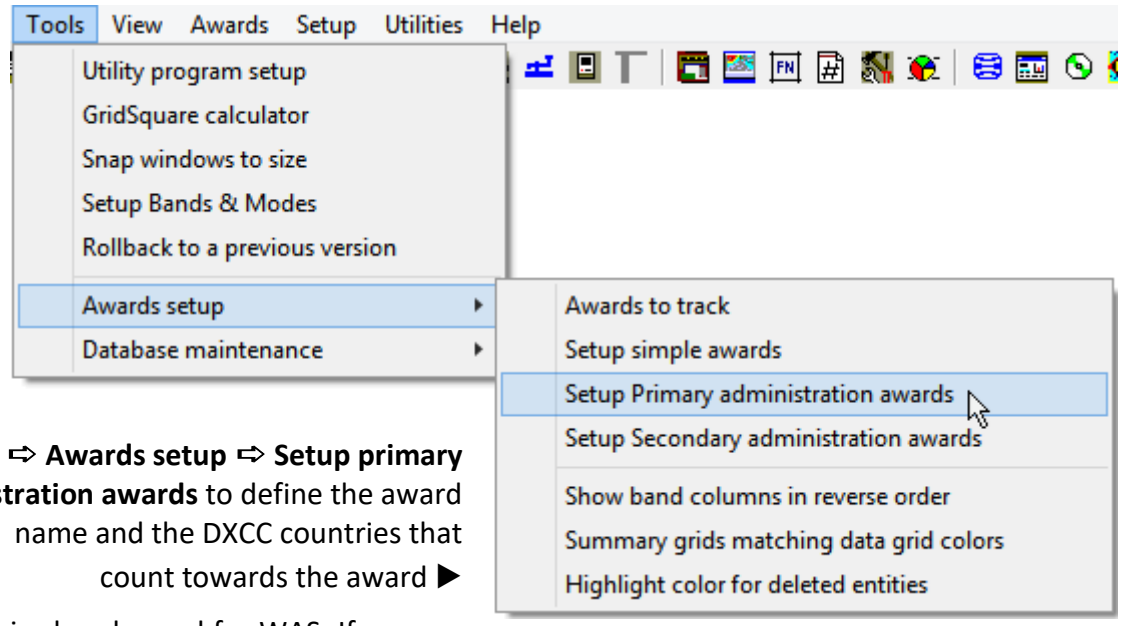
Hinson tip: **backup a database that you are going to edit, *before* you start editing it** *i.e.* make copies of those files for each database. You are entering dangerous territory … and if it all goes horribly wrong, you may need to backout. Trust me, restoring a few files from a recent backup is *much* easier and *much* more likely to succeed than trying to unscramble the mess you may have inadvertently caused.

Hinson tip: it may also be worth backing up any database that you have edited, *after* you have finished editing them *i.e.* make post-edit copies of those files for each database. That way, if a Logger32 update, re-install or some other problem damages, deleted or overwrites them, you can simply restore the backups. You can also recover using the *.CSV* files, so save them safely too.

Copyright © 2026 Robert C. Furzer K4CY

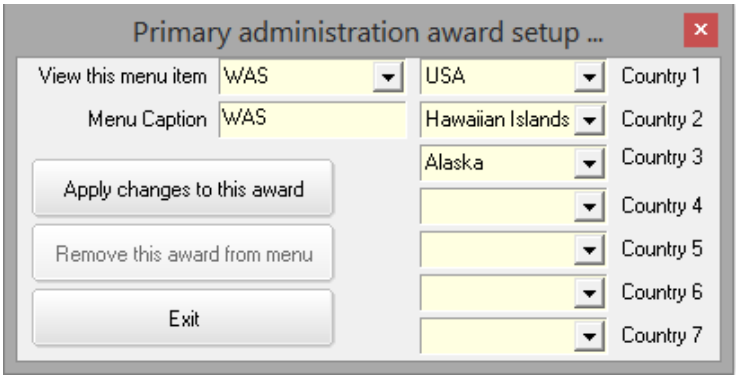
18.4.4 Primary administration awards

Primary administration awards use only information from the Primary administration database. ARRL's **W**orked **A**ll **S**tates award, for example, uses data on three DXCC entities (USA, Alaska and Hawaii). In order to set up a Primary administration award, you need the award name and details of the administrative areas to use, generally as specified in the published rules for the award.



Click **Tools ⇨ Awards setup ⇨ Setup primary administration awards** to define the award name and the DXCC countries that count towards the award ▶

Award № 1 is already used for WAS. If you are preparing a new award, pick a number that has no associated name. In any event, be *very* careful not to overwrite an existing award.



◀ Use the pull-down lists to select the countries that apply to this award.

Finally, click <**Apply changes to this award**> to make the updated/new award table available as an award.

18.4.5 Secondary administration awards

Awards for working secondary administrative areas (such as counties or guns) *must* each be associated with a specific primary administration. DOK, JCC and JCG are examples that are already supported by Logger32. Awards relevant to areas within your own country could also be configured.

In order to make this work:

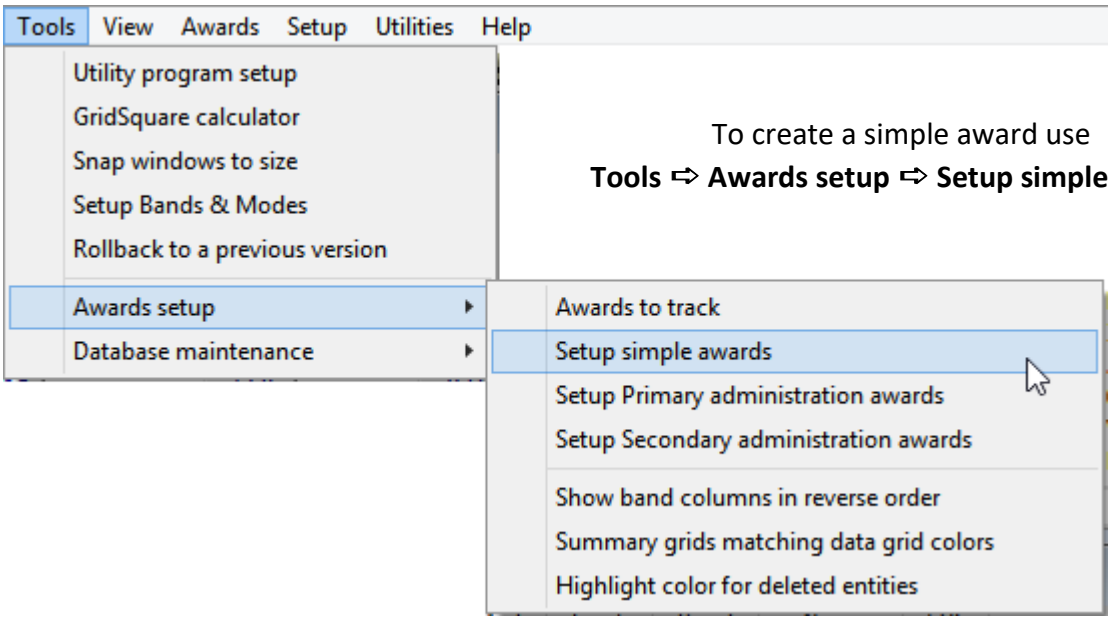
1. Ensure Primary administration data are present. Not all DXCC entities have primary data.
2. Construct a new secondary data list.
3. Load the new secondary data list into the Secondary administration database.
4. Make the new award available in the Secondary administration awards menu.

Copyright © 2026 Robert C. Furzer K4CY

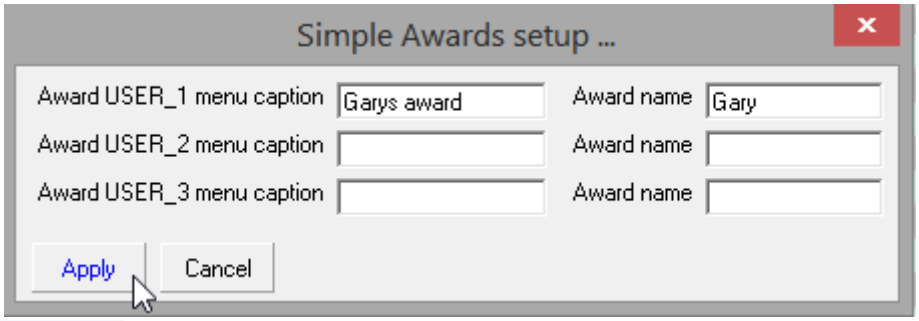
18.4.6 Simple awards

Simple awards have no finite reference checklist as such: qualifying QSOs in the log are used to collate a list of all the collected items (*e.g.* club members' membership numbers). Data are collected from one of the three "User" fields and continue to grow as new entries are worked.

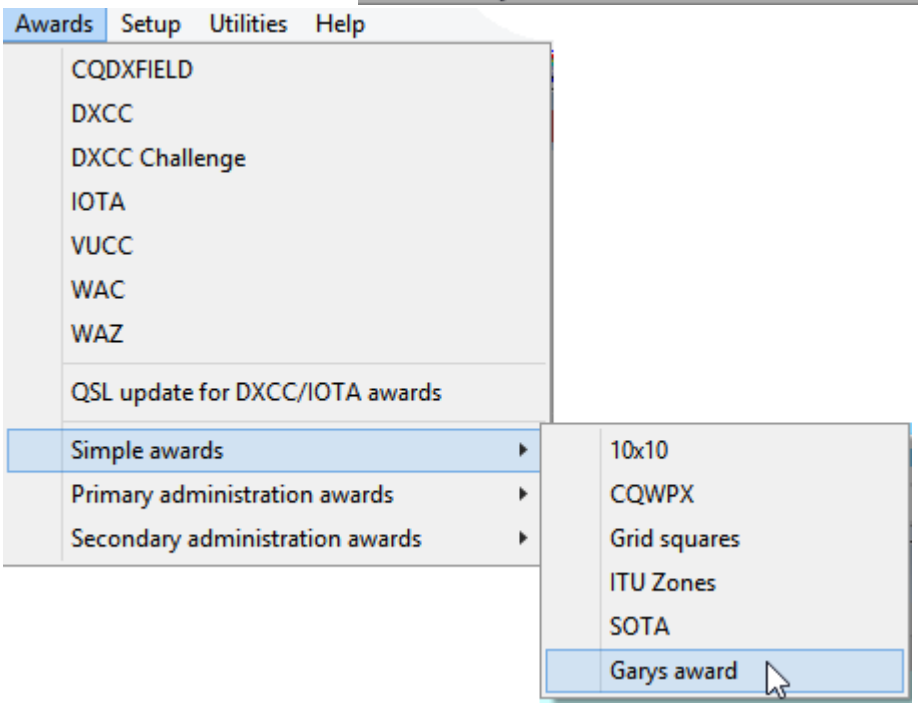
You can collect almost anything you want *e.g.* special DOKs, Castles|Summits|Parks|Volunteers On The Air, lighthouses, WorldWide Fund for Nature sites *etc*.



To create a simple award use
Tools ⇨ Awards setup ⇨ Setup simple awards



Supply a name for your award and a caption (label), then click <**Apply**> ▶

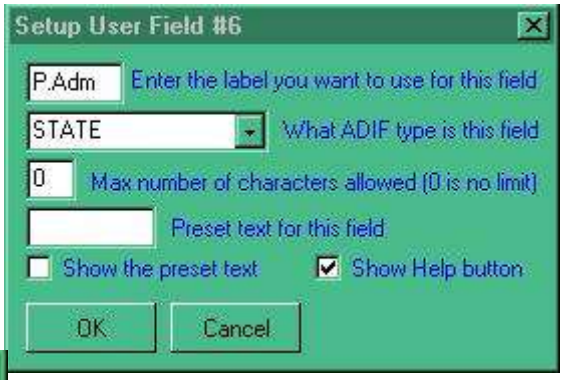


◀ The award's name is listed under **Awards ⇨ Simple awards** and appears on the award table caption.

Copyright © 2026 Robert C. Furzer K4CY

18.4.7 Entering award data via the log entry pane

When entering award data into the log, type the primary (STATE) and secondary (CNTY) data into the log entry pane (as for US State and County).

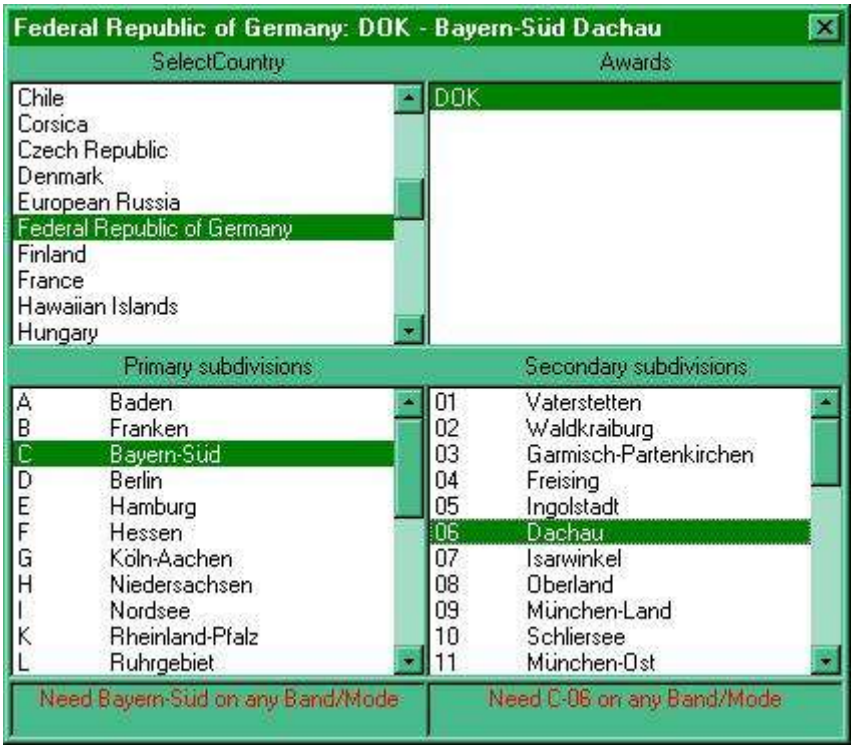


In the example below, the "P.Adm" and "S.Adm" input fields are in fact the STATE and CNTY fields, renamed using the user field setup form ▶



◀ If you are unsure of the correct designation for STATE and CNTY, click the <**?**> buttons …

… and select the primary and secondary subdivisions as required using the right mouse button to update the log entry pane, ready to log the QSO.

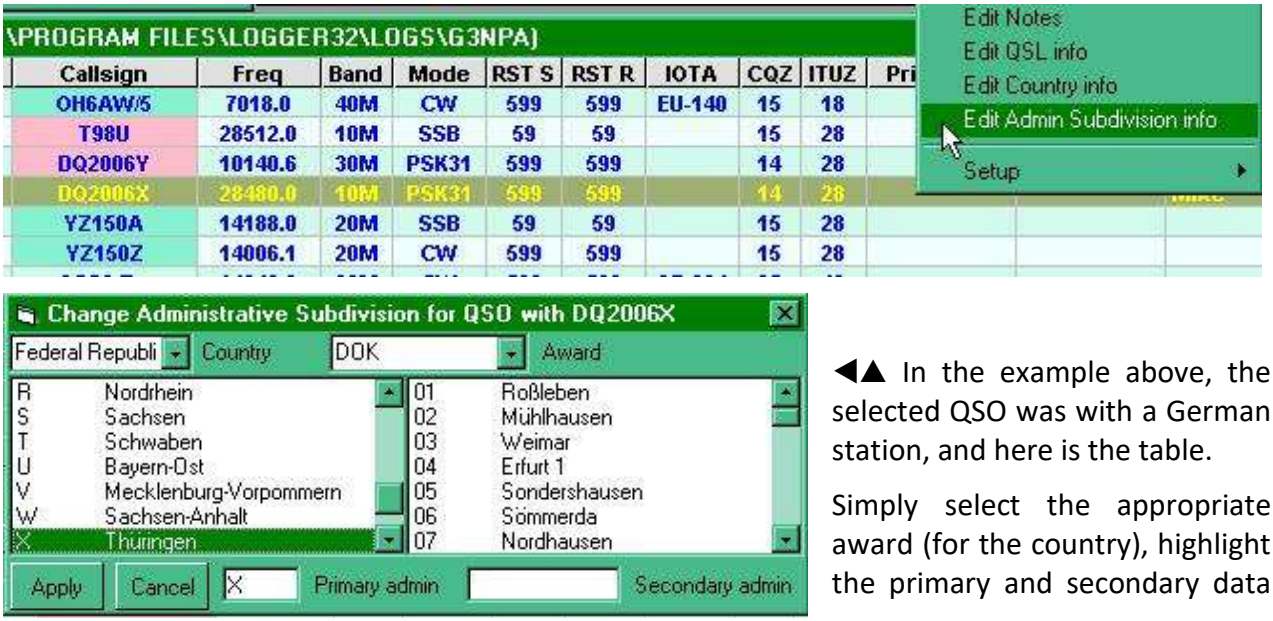


Logger32 displays further hints at the bottom ▶

18.4.8 Entering/editing award data in the logbook

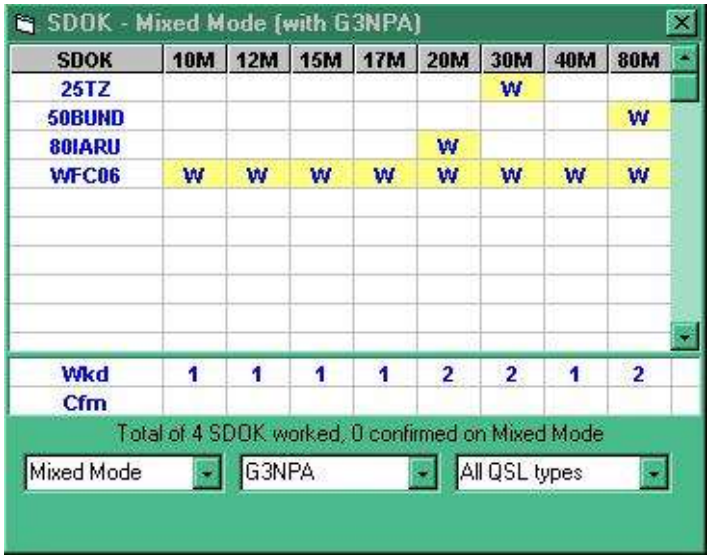
It is not possible to edit the primary/secondary administrative division columns of the logbook directly because they are key fields. Instead, right-click the primary or secondary admin column to open a menu. Click <**Edit Admin Subdivision info**> then select from the complete set of primary and secondary divisions available.

Copyright © 2026 Robert C. Furzer K4CY



◀▲ In the example above, the selected QSO was with a German station, and here is the table.

Simply select the appropriate award (for the country), highlight the primary and secondary data (or type in the code into the appropriate panes), then click <**Apply**> to transfer the data to the log.

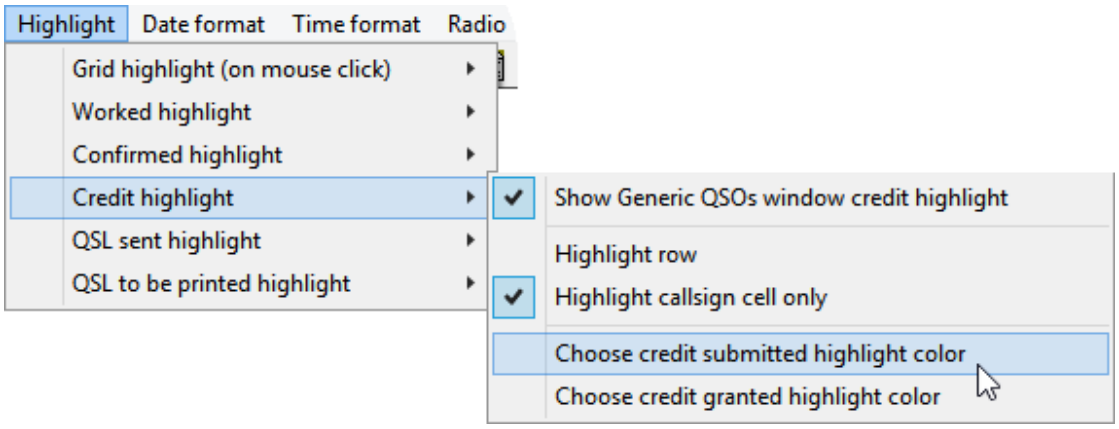


On occasions the award pane may not fill automatically, either because there is more than one award using the primary data (*e.g.* the JCC and JCG awards in Japan) and Logger32 cannot determine which to use, or no primary data has been set up for the particular country.

If the primary code does not exist, Logger32 displays an error message. If the code being entered is valid, check the primary and secondary databases for the country and/or award.

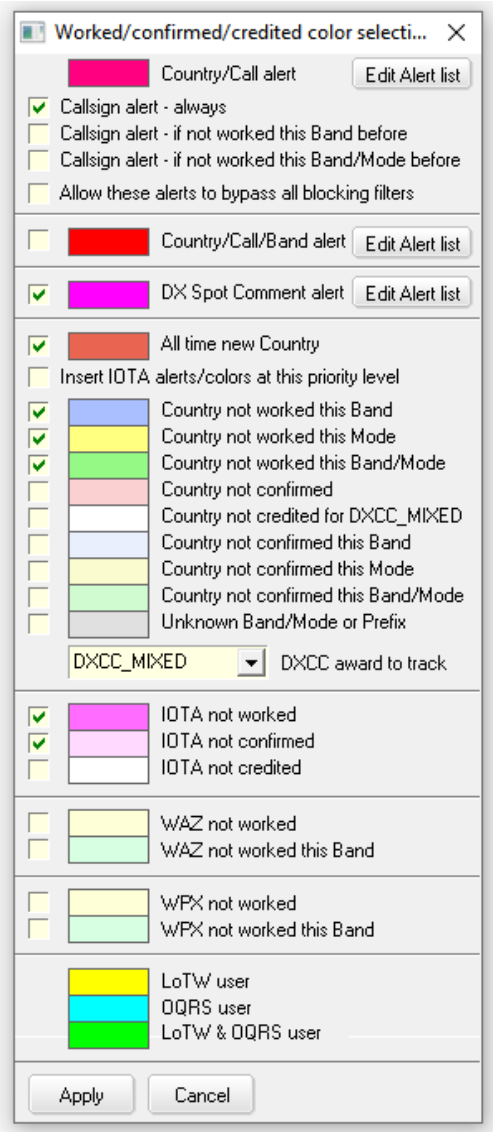
18.4.9 Award tabs

Configure the colors used to highlight **S**ubmitted and **G**ranted credits in the awards using **Setup ⇨ Highlight ⇨ Credit highlight ⇨ Choose credit submitted|granted highlight color ▼**



Copyright © 2026 Robert C. Furzer K4CY

18.4.10 DX spot highlight colors



To open this menu,
right-click the DX Spots pane,
then click **Setup ⇨ Appearance ⇨ DX Spot highlight colors**.

The award named in
<**Country not credited for** [award]> ▶

… is determined by
the selection under
<**DXCC award to track**> ▶

In this example, highlighting configured in the middle section of the form concerns the DXCC_DIGITAL award.

The award selection (DXCC, DXCC-MIXED *etc*.) has no priority. Whatever is selected is used: credits (no matter how many, or what specific DXCC entity they may be for) are ignored *unless* they have been applied to the particular DXCC award you have selected.

18.5 DXCC awards

18.5.1 DXCC entity (country) validation of your log using Club Log

You can validate the DXCC entities assigned to logged QSOs in your logbook using Club Log's excellent, comprehensive and up-to-date information. Instructions below.

Taking account of the date ranges for known DXpeditions, rare entity activations, special events and other prefix/callsign exceptions, the process systematically determines the most likely DXCC entities for all your QSOs *at the time they were made*.

Regularly updating the DXCC exceptions from Club Log will help insure new QSOs are logged with the appropriate DXCC entities. See database maintenance for more on this.

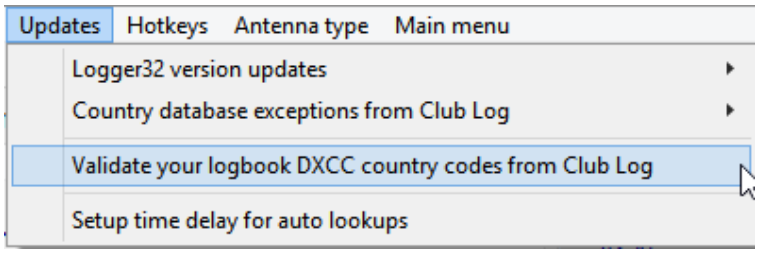
18.5.2 Validate QSOs already logged

Warning: the validation process is likely to modify your logbook.
Backup your logbook *before* landing yourself in deep trouble.

Copyright © 2026 Robert C. Furzer K4CY

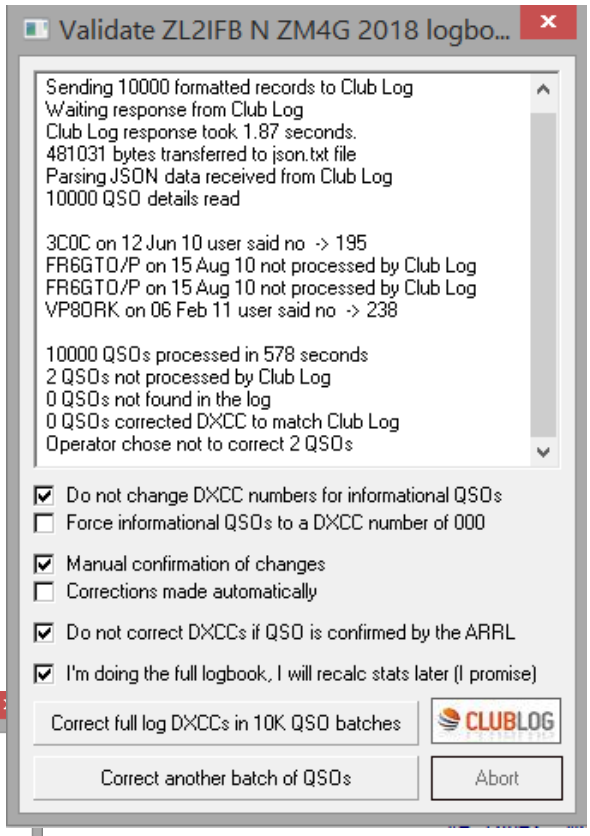
Hinson tip: you can either let it rip and update detected DXCC anomalies in your logbook automatically, or review each suggested log change manually before deciding whether to accept or reject it. Either way, the process is capable of validating ~10,000 QSOs in less than a minute. By hand, it would take *hours*!

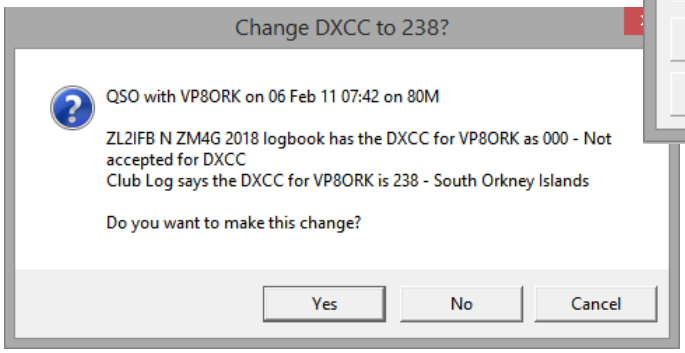
Click **Setup ⇨ Updates ⇨ Validate your logbook DXCC country codes from Club Log** to kick-start the process ▶



Then tick the relevant options on the form that opens:

- **Do not change DXCC numbers for informational QSOs**: having deliberately marked certain log entries as 'informational', you have already determined that they are not valid QSOs, so there is no need to update the DXCC entities or other details.
- **Force informational QSOs to a DXCC number of 000**: a DXCC entity number of 000 indicates that a QSO is 'Not accepted for DXCC' – for example a pirate operation, incomplete QSO, or operation entirely from a boat. DXCC=000 also excludes the QSO from your DXCC statistics, and means that any DXcluster spots or QSOs with the applicable DXCC entity may be flagged and alerted as 'new ones' unless you already have other valid QSOs in the log.
- **Manual confirmation of changes**: lets you review and decide what (if anything) to do with each of the anomalies as they are found ▼





• **Corrections made automatically**: tick this instead if you made that backup *and* you are happy for Logger32 and Club Log to make such important decisions for you[316].

[316] That's the *Just flamin' get on with it!* option … and to be honest, it's pretty safe. The few identified 'errors' I choose not to correct in my 125k log are QSOs that were dubious, incomplete or unconfirmed, that I manually set to DXCC entity code 000 but retained just in case they are ever resolved. I could append "=" to the callsigns to mark them as 'informational QSOs' so as not to be reminded each time.

Copyright © 2026 Robert C. Furzer K4CY

- **Do not correct DXCCs if QSO is confirmed by the ARRL** handles QSOs that have already been validated and perhaps even claimed and granted for award purposes on LoTW. Even if you or Club Log believes these QSOs were in fact invalid for some reason (*e.g*. incomplete), having been validated by LoTW, there is not much point in now 'correcting' your log.
- **I'm doing the full logbook, I will recalc statistics later (I promise)**: this option lets the validation process run faster by *not* automatically recalculating your statistics if any log changes are made, so you have to remember to do afterwards, or else your award statistics may not reflect the corrected log. Leaving this option *un*ticked results in your statistics being updated automatically during the process: a little slower but easier and safer. Your choice.

The function processes batches of up to 10,000 QSOs at a time. Start by clicking <**Correct full log DXCCs in 10K QSO batches**> to start validating the first batch. If your logbook has more than 10k QSOs, click <**Correct another batch of QSOs**> after each batch completes.

Hinson tip: you don't need to validate your entire log in one sitting. Logger32 'remembers' which batch or batches it has validated so far, and automatically validates the *next* batch whenever the function is next run, so after the first run, all you need to do is find some time to open it and click <**Correct another batch of QSOs**> whenever it suits you – maybe every few months - to continue validating your log. You can always start re-validating your entire log if you wish by clicking <**Correct full log DXCCs in 10k batches**> instead, which resets Logger32's 'last validated QSO' counter – something you might find worthwhile every year or three, taking advantage of any additional validation information from Club Log about QSOs that were previously validated (very rare in practice).

Don't forget to recalculate your statistics if you promised to do so earlier!

18.5.3 QSL update for DXCC/IOTA awards

Use **Awards ⇨ QSL update for DXCC/IOTA awards** to do batch updates of award status flags, converting QSOs from <**Submitted**> to <**Granted**> ▶

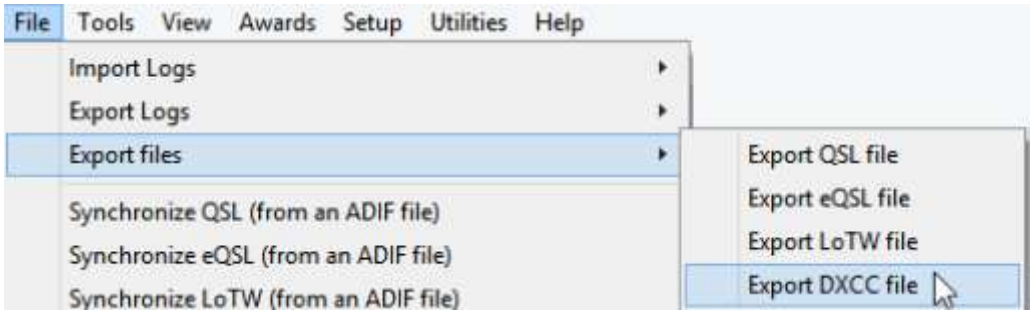


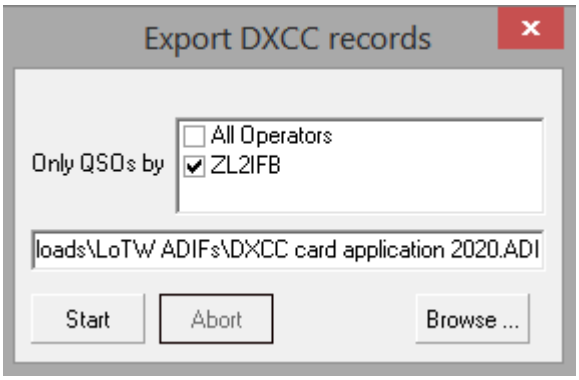
18.5.4 Applying for DXCC with QSL cards

In addition to any LoTW confirmations, you can submit QSL cards towards your DXCC award using ARRL's Online DXCC System. While you *can* enter the QSO details from your QSL cards manually, one at a time, that is error-prone and tedious.

Copyright © 2026 Robert C. Furzer K4CY

Handily, Logger32 is clever enough to generate an ADIF file listing QSOs with 'new ones' (*e.g.* your very first QSO with a DXCC entity, or the first one on a given band and mode slot) which have been confirmed on QSL cards but *not* on LoTW. To create the file listing your QSL cards to be submitted for DXCC, use **File ⇨ Export files ⇨ Export DXCC file ▼**

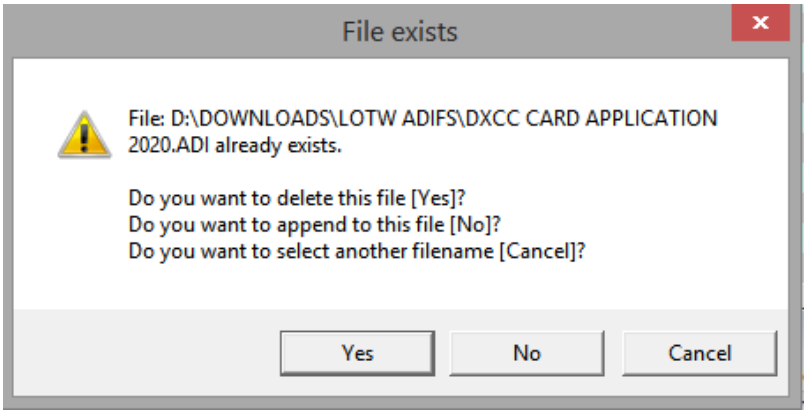


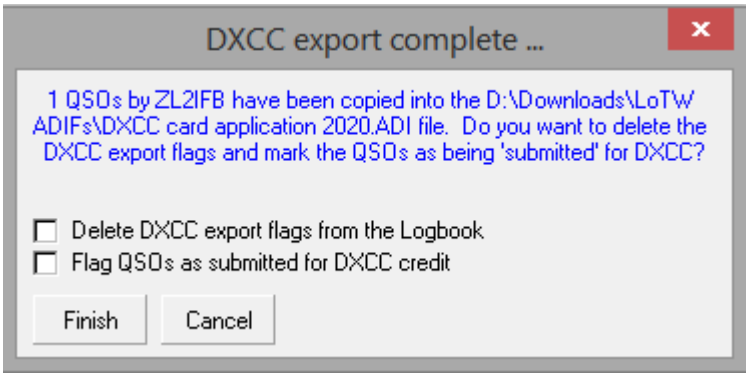


◀ Select the operator/s (callsigns) whose QSOs are to be submitted for DXCC, and specify a folder and filename for the *.adi* output file (if you're unsure, use <**Browse …**> to look for a suitable folder).

Then click <**Start**> to get going.

If the output file already exists, Logger32 politely asks whether to delete and replace it (click <**Yes**>), or append to it (click <**No**>), or abort the export process (click <**Cancel**>) ▶





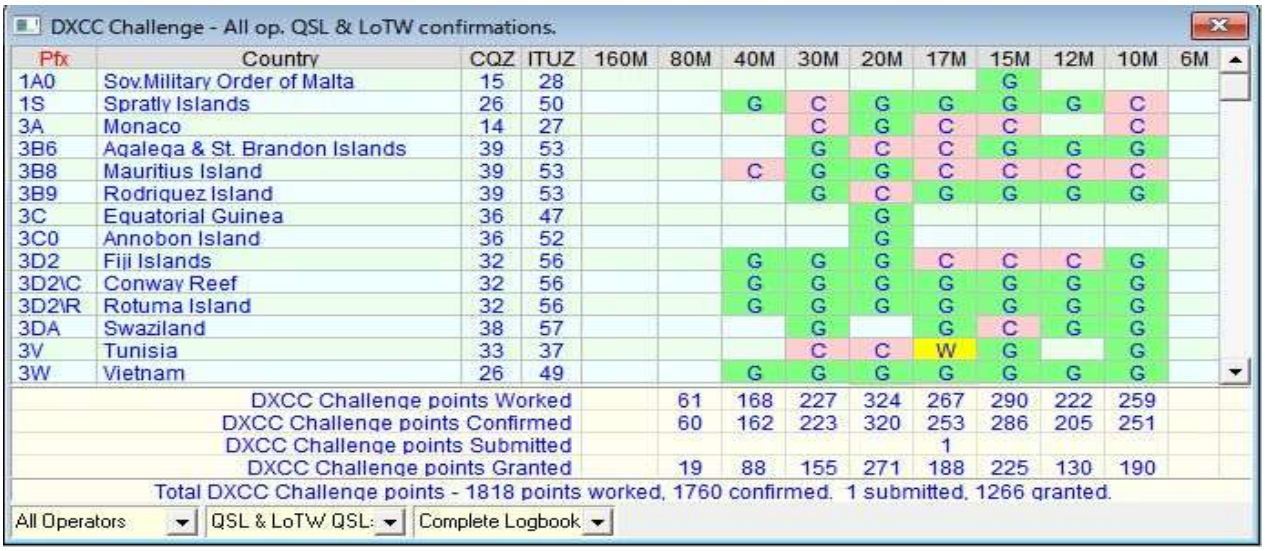
◀ After the file has been exported, you are asked whether to <**Delete DXCC export flags from Logbook**> and/or to <**Flag these QSOs as submitted for DXCC credit**>. At this point in the process, you haven't actually submitted[317] the file for DXCC, so you may prefer to wait until you have done that before updating the flags,

[317] "Confirmed", "Submitted" and "Credited" mean different things. "Confirmed" indicates that you have received confirmation in some form from the QSO partner. "Submitted" indicates that you have used the

Copyright © 2026 Robert C. Furzer K4CY

cancelling it for now: that way, if something goes wrong with the submission, you can simply re-run the DXCC export and try again, without having to locate the QSOs and flag them for export once again. Conversely, if the DXCC submission goes to plan, you can re-run the export but this time update the flags at the end.

The DXCC Challenge always displays all 10 bands actually covered by the award, regardless of your Bands & Modes table ▼


DXCC Challenge - All op. QSL & LoTW confirmations.

Pfx	Country	CQZ	ITUZ	160M	80M	40M	30M	20M	17M	15M	12M	10M	6M
1A0	Sov.Military Order of Malta	15	28							G			
1S	Spratly Islands	26	50			G	C	G	G	G	G	C	
3A	Monaco	14	27				C	G	C	C		C	
3B6	Agalega & St. Brandon Islands	39	53				G	C	C	G	G	G	
3B8	Mauritius Island	39	53			C	G	G	C	C	C	C	
3B9	Rodriquez Island	39	53				G	C	G	G	G	G	
3C	Equatorial Guinea	36	47					G					
3C0	Annobon Island	36	52					G					
3D2	Fiji Islands	32	56			G	G	G	C	C	C	G	
3D2\C	Conway Reef	32	56			G	G	G	G	G	G	G	
3D2\R	Rotuma Island	32	56			G	G	G	G	G	G	G	
3DA	Swaziland	38	57				G		G	C	G	G	
3V	Tunisia	33	37				C	C	W	G		G	
3W	Vietnam	26	49			G	G	G	G	G	G	G	
	DXCC Challenge points Worked				61	168	227	324	267	290	222	259	
	DXCC Challenge points Confirmed				60	162	223	320	253	286	205	251	
	DXCC Challenge points Submitted								1				
	DXCC Challenge points Granted				19	88	155	271	188	225	130	190	

Total DXCC Challenge points - 1818 points worked, 1760 confirmed, 1 submitted, 1266 granted.

All Operators | QSL & LoTW QSL: | Complete Logbook

For additional information, see the logbook and previous QSOs chapters.

18.5.5 Flagging and *exporting* QSL award files

Logger32 can export a *DXCC.ADI* file containing details of your QSOs that have been confirmed on QSL cards.

Hinson tip: QSOs in the file are sorted in date order. Be kind to the nice card-checkers: please sort, check and submit your QSL cards for verification in the same order as the form, especially if you have a pile of cards to be checked. Why make the checker grumpy by having to sort out your piles first?

confirmed QSO to apply for an award. “Credited” means the QSO has been checked by the award administrators and accepted for the award in question.

Copyright © 2026 Robert C. Furzer K4CY

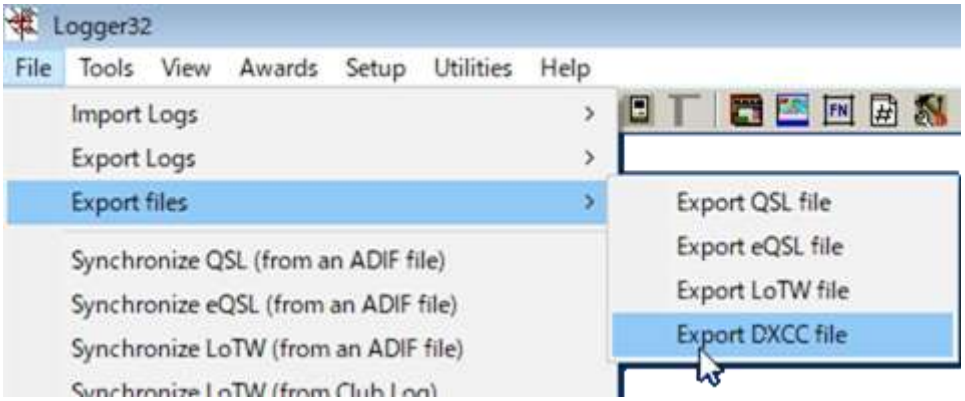
QSOs confirmed on QSL cards can be marked for inclusion in the *DXCC.ADI* file using either of two ways:

1. Type the callsign from the QSL card in the log entry pane or highlight a QSO in the logbook to display contacts in the previous QSOs pane. On the previous QSOs pane, right-click whichever QSO is confirmed on the QSL card and select <**DXCC submission flagged**> on the Quick QSL form to flag the QSO for your next DXCC file export ▶
2. Repeat as appropriate for any other QSOs on the same card.
3. Find and right-click any of the confirmed QSOs in your logbook, then click <**Submit for DXCC**>.



Hinson tip: while a QSL card is being verified by ARRL, it makes sense to have all the QSOs on the card checked and hopefully verified at once, even if you only 'need' one right now *e.g.* for the basic DXCC award. Confirmed QSOs on other bands and modes, for instance, should count towards band and mode-specific awards that you may desire later on.

To export the *DXCC.ADI* ADIF file, use **File ⇨ Export files ⇨ Export DXCC file ▼**





◀ Select the operator whose QSOs are to be exported, and a filename for the output, then click <**Start**> to run the export.

Copyright © 2026 Robert C. Furzer K4CY

Following the export, you have two flagging options ▶

- **Delete DXCC export flags from the Logbook**: resets (clears) the DXCC export flags.
- **Flag QSOs as submitted for DXCC credit**: ticks the relevant 'submitted' boxes on the Award Status form.



Click <**Finish**> to, errrr, well, finish.

With <**Flag QSOs as submitted for DXCC credit**>, the award table is updated:

- **All QSOs:** if the QSO does not have the *DXCC granted* flag set, the *DXCC submitted* flag is set.
- **All QSOs**: if the QSO does not have the *DXCC_MIXED granted* flag set, the *DXCC_MIXED submitted* flag is set.
- **CW QSOs**: if the QSO date is valid for the DXCC_CW award, and if the QSO does not have the *DXCC_CW granted* flag set, the *DXCC_CW submitted* flag is set.
- **SSB, AM and FM QSOs**: if the QSO does not have the *DXCC_PHONE granted* flag set, the *DXCC_PHONE submitted* flag is set.
- **All other mode QSOs (apart from CW, AM, FM and SSB)**: if the QSO does not have the *DXCC_RTTY granted* flag set, the *DXCC_RTTY submitted* flag is set.

If the granted flag is already set, Logger32 does nothing. There is no need to submit it again!

18.5.6 Updating your DXCC award 'granted' QSO status using LoTW reports

To record the granting of QSOs for various DXCC awards, follow the steps described in the LoTW chapter.

18.5.7 DXCC status update when submitted QSOs are granted



◀ This example lists QSOs in the logbook for which QSL cards have been submitted to the ARRL for credit toward an award.

Assuming they are accepted, update the status of these QSOs from Submitted to Granted by clicking to tick the applicable QSOs. If there are numerous QSOs listed, *all* of which have been accepted for the award, simply click <**Check all**>.

Copyright © 2026 Robert C. Furzer K4CY

Click <**Mark checked QSOs as granted**> to update the logbook accordingly.

For QSOs to show on this list, the corresponding QSL cards should have been received and flagged as submitted using <**Set/show awards credits**>.

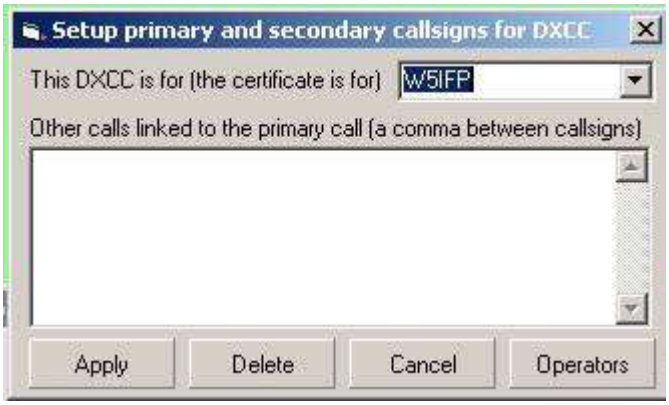
QSL/eQSL/LoTW QSLs received flags cannot be cleared from a QSO once it has been submitted/granted for an award.

Automatic updates are also available from the ARRL using <**Update from ARRL file**>. Insert the path and filename of the data file produced by *ARRL scraper.exe* ▶



You can wipe and recreate all your credits granted, or just record the new ones. Make your choice then click <**Start**>.

If you have more than one callsign linked to your primary callsign, click <**Setup**> and the system will list all operator callsigns found in the logbook ▼

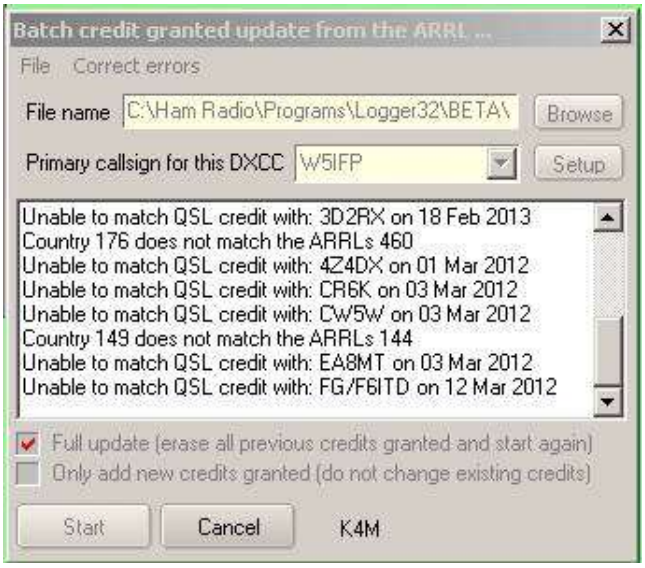


Delete all calls not linked to the master callsign.

◀ Clicking <**Operators**> offers all the operators in the open logbook.

If you have mixed-and-matched DXCCs all in one logbook, then this probably isn't the best option to click.

Once the process is started, Logger32's progress bar shows it working and the update window updates with each record processed ▶▼





When doing a "Full update" of ARRL statistics (*e.g.* your first time through this), the Logger32 statistics dataset is reconstructed without ARRL/DXCC information. Non-DXCC credits are then reapplied to the statistics and the download process starts.

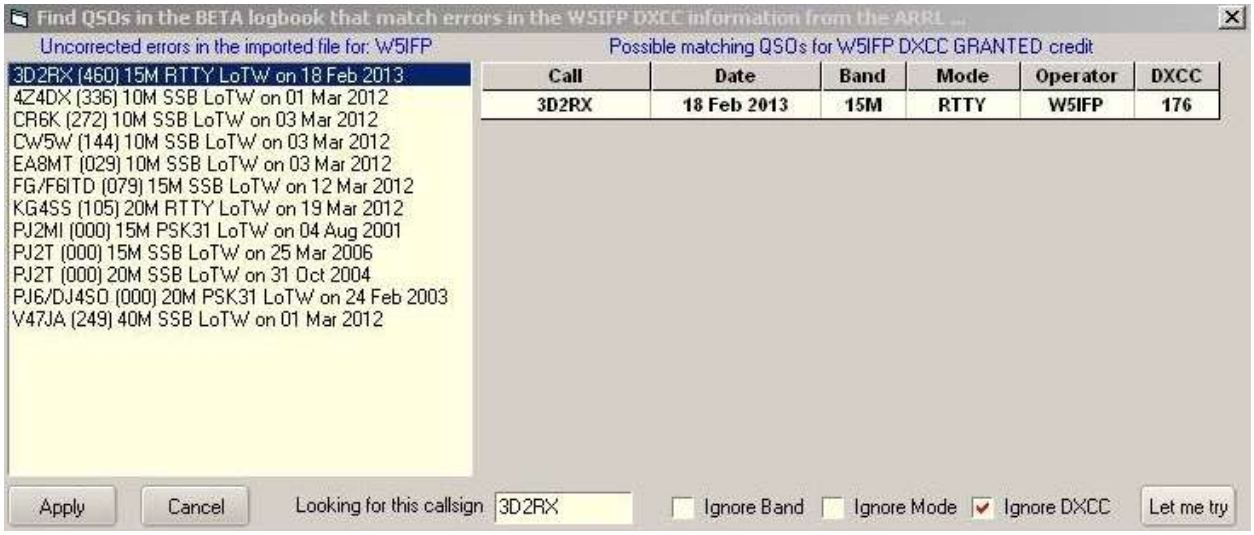
Copyright © 2026 Robert C. Furzer K4CY

Once all the records have been processed, you'll see a curt but reassuring message, namely "Done!" ▶



Then a summary to read and <**OK**>▼





▲ Click <**Let me try**> for Logger32 to look for a matching QSO.
If no match is found, you will see ▼



Having failed to find any direct match, you can manually update the QSOs. Click the upper tab "correct error" for a chart showing the ARRL records from the reject file. These are the records where a complete match could not be found. Click these records and a possible match from the logbook appears on the right side of the window. If you find the correct record listed, click it (right side of chart) and that QSO in the logbook will be updated.

Copyright © 2026 Robert C. Furzer K4CY

You may have to tick options such as <**Ignore DXCC**> to locate the correct QSO record. When you select all possible matches, click <**Save**> to update the ARRL file.

In order to update the logbook records, repeat the update cycle. Click <**Start**> on the "Batch credit" table and be sure <**Add only new credits granted**> is enabled. This process may have to be repeated several times until all records are mapped to the correct QSO.

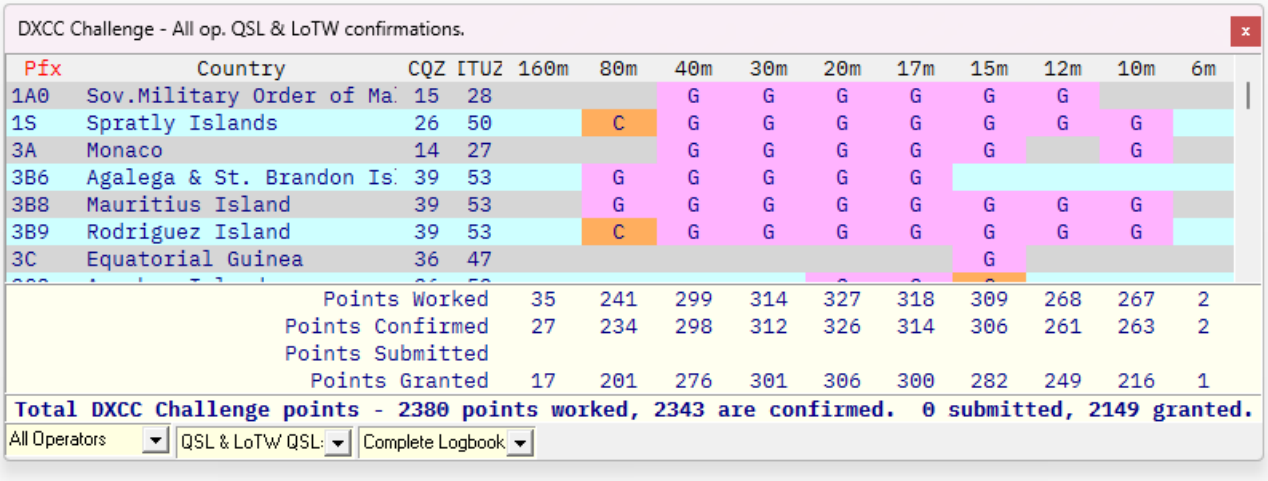
If you can't figure out what the text means (there's some weird stuff sometimes!) for an entry in the left list, right-click the record in the left row and this will take you directly to the QSO record in LoTW. You can see for yourself what the ARRL says for the QSO.

The first time this process is used, extensive manual correction/editing may be required. Some of the Beta crew had as many as 300 credits to be manually corrected/edited using this function. Once all errors in the ARRL data have been mapped to the corresponding QSOs in your log, it *should* be plain sailing …

18.5.8 DXCC Challenge award

"The DXCC Challenge Award is earned by working and confirming at least 1,000 DXCC band-points on any Amateur bands, 160 through 6 meters (except 60 meters). Certificates are available for this award, and there is a distinctive wall plaque available to display your achievement. Plaques can also be endorsed in increments of 500 additional band points. Deleted entities do not count for this award. All contacts must be made after November 15, 1945. QSOs for the 160, 80, 40, 30, 20, 17, 15, 12, 10 and 6 meter bands qualify for this award. Bands with less than 100 contacts are acceptable for credit for this award and you do not have to have an active award on a band for it to qualify." [www.arrl.org/dxcc-challenge].

Since Logger32's statistics function routinely keeps track of our DXCC counts as we log QSOs, a database report (opened using **Awards ⇨ DXCC Challenge**) is all it takes to tot-up our current entity counts and display our DXCC Challenge status ▼



As usual, use the scrollbar on the right to look through the table, and click any of the colored blobs to call up the respective QSOs. Click the headers of the three leftmost columns to sort the table by prefix, country name or CQ zone if you wish. Click-and-drag a corner to resize the window.

Notice the three yellow selector boxes at the bottom of the report:

- If the open log contains QSOs by multiple operators (*e.g.* family or club members sharing a station and Logger32 log), you can select and display the Challenge statistics individually or all together.

Copyright © 2026 Robert C. Furzer K4CY

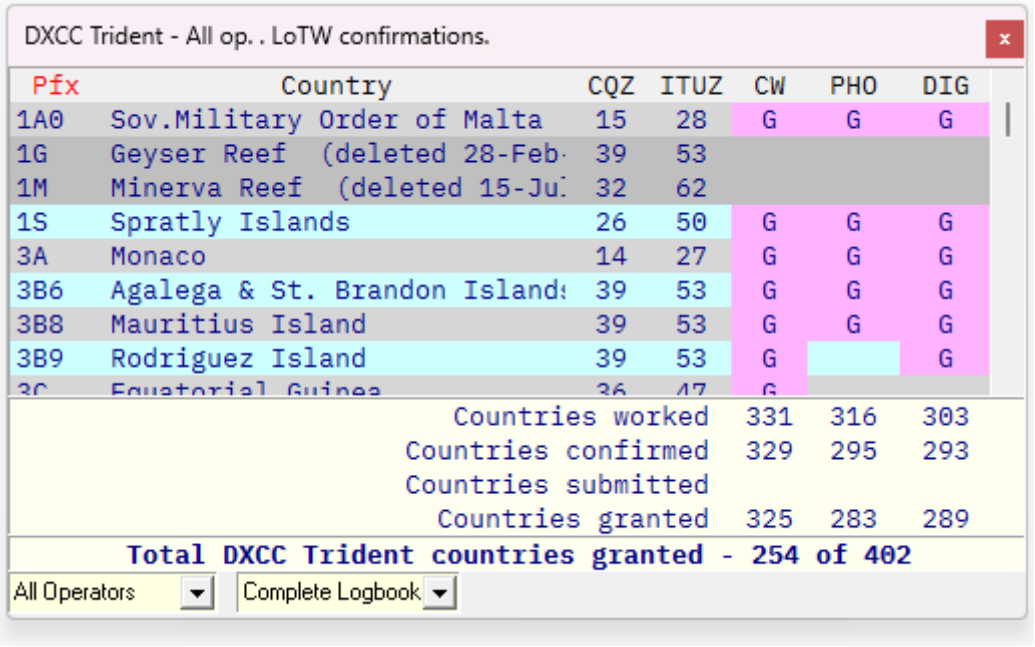
- ARRL requires LoTW or QSL card confirmations. QSL cards have to be verified manually by the DXCC desk at ARRL HQ or by ARRL-authorized card-checkers and added to your DXCC record, whereas LoTW confirmations are verified automatically and added directly.
- You can filter the statistics by particular years if you wish, perhaps using the current year's totals to motivate you to DX harder and compete for the annual De Soto cup.

Once the "Total DXCC Challenge points" reach at least 1000 confirmed, you have enough to apply to ARRL for the award – either a wooden wall plaque or a paper certificate. Endorsements are available as you clear subsequent hurdles.

18.5.9 DXCC Trident award

"The ARRL DXCC Trident award recognizes amateur radio operators who have confirmed two-way radio contacts (QSLs) with 100 or more DXCC Award Countries (entities) on all three modes: CW (Morse Code), Phone (Voice) [and] Digital (FTx, RTTY, PSK, etc.). Once you've met the qualifying DXCC criteria through Logbook of The World (LoTW), you're eligible to order a handsome ARRL DXCC Trident plaque that proudly commemorates your achievement." [www.arrl.org/dxcc-trident-award].

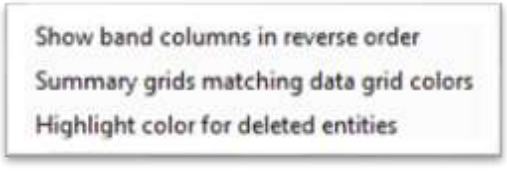
Open Logger32's report using **Awards ⇨ DXCC Trident ▼**



The Trident award rules specify that only qualifying contacts made *after* October 1976 count, so Logger32 dutifully ignores any earlier QSOs for this award. The rules also state that "QSL cards are not accepted for this award" … but QSL cards can, in fact, be submitted for checking under the DXCC rules, and provided your DXCC record is linked to/merged with your LoTW records (a one-time step on request to the ARRL Awards Desk), verified QSL cards count towards your DXCC *and* DXCC Trident awards, and both types of confirmation are shown through LoTW.

Apply for the Trident award plaque here. Endorsement plates are available at the 200, 300 and Honor Roll levels for accomplished multimode DXers.

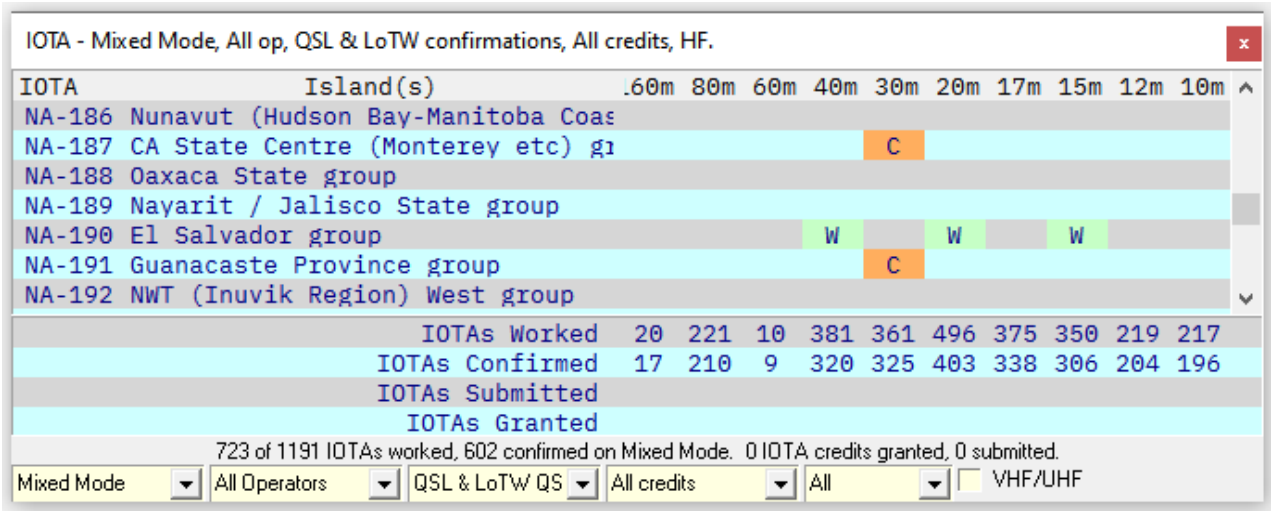
Hinson tip: you might like to try out these three display options under **Tools ⇨ Awards setup ▶**



Copyright © 2026 Robert C. Furzer K4CY

18.6 IOTA award

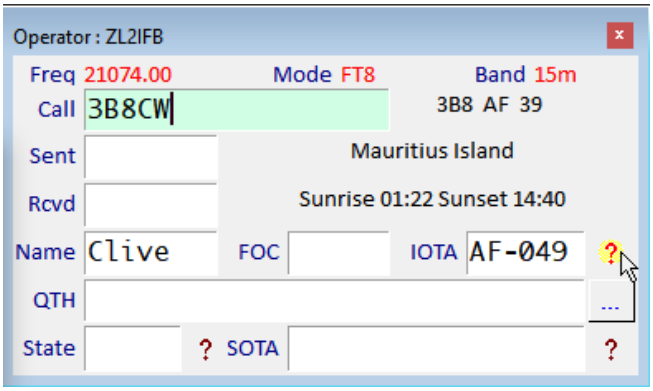
Islands On The Air is another popular set of awards for DXers who submit proof of having contacted the prescribed number of islands or island groups ▼



The standard IOTA award only counts QSOs on the HF bands. To view the VHF/UHF IOTA award instead, click to tick <**VHF/UHF**> at the bottom of the form ▲

Clicking any of the colored cells shows the corresponding **W**orked, **C**onfirmed, **S**ubmitted and **G**ranted QSOs.

18.6.1 IOTA island information/search window

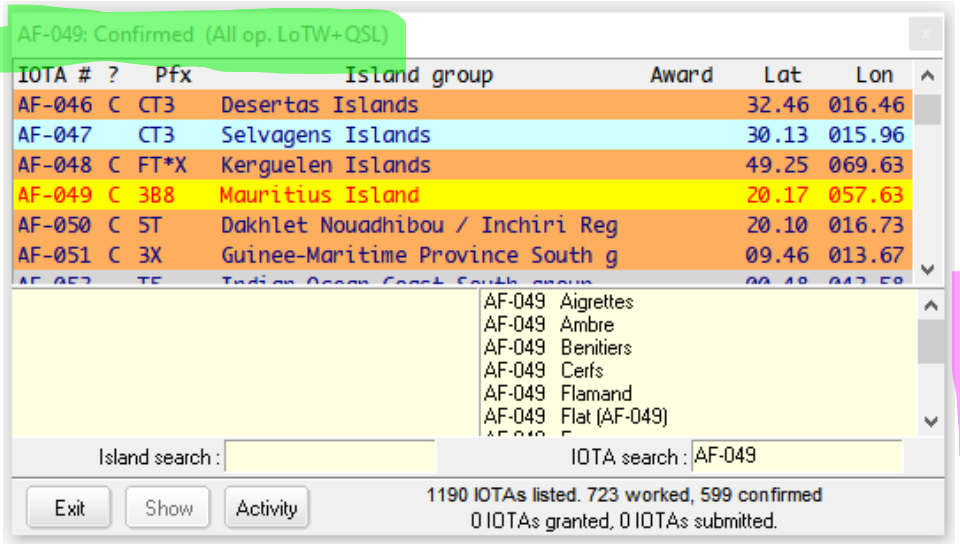


◀ The simplest way to find out more about an IOTA reference is to enter a callsign and IOTA number into the log entry pane[318] then click the **?** beside the IOTA field (if shown).

[318] Choose IOTA for one of the log entry pane user fields, and optionally have it automatically populated from the most recently-logged QSO with the same DX station, also carrying forward the op's name *etc*. as you wish, using the cool QSO mask function.

Copyright © 2026 Robert C. Furzer K4CY

A window opens with additional information about that island group ▶



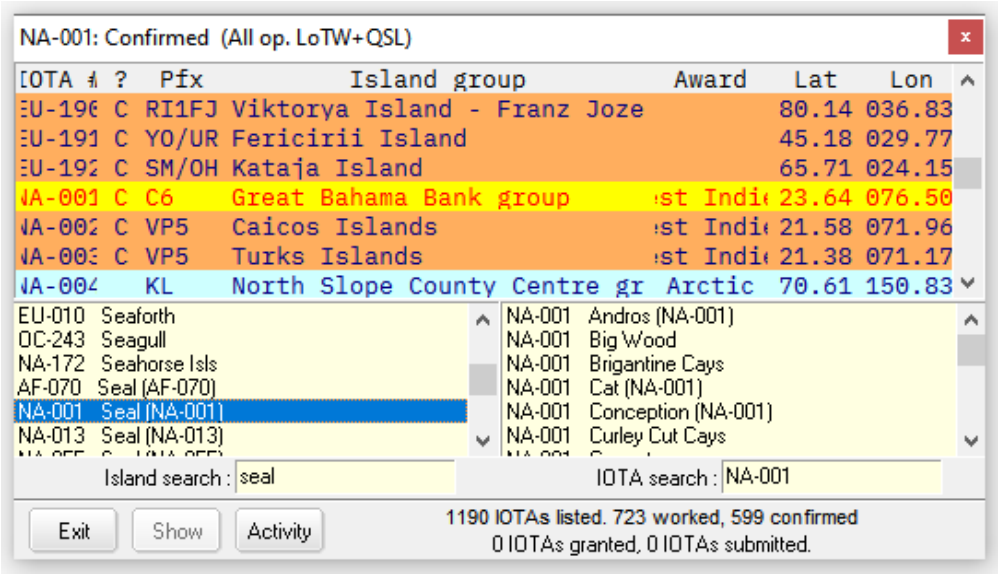
In this example, the window caption indicates that AF-049 has been **confirmed** on **LoTW** and/or **QSL** card/s for at least one of the **op**erators with QSOs in the open logbook.

The data table highlights 3B8CW's island group row with a canary yellow background. The table columns are:

- **IOTA #**: the official IOTA reference number for the island group, in the specified format – two letters for the continent followed by a hyphen then 3 digits.
- **?**: indicates whether the island has been worked (W), confirmed (C), submitted for the IOTA award (sent for checking) (S), and eventually granted (credited) for the award (G).
- **Pfx**: shows the prefix, indicating the DXCC entity for the island group. This can be handy when we work IOTA DXpeditions to obscure islands using unique callsigns.
- **Island group**: is the [Anglicized, official] IOTA name of the island group, or the individual island if it is the only member of the group. AF-049 group, for instance, encompasses several islands: slide the slider to peruse the list of islands with the selected IOTA reference.
- **Award**: shows IOTA award/s for which this island group has been submitted and/or credited. There are progressive certificates for each 100 IOTAs confirmed, plus more for certain areas of the world, and endorsements for specific modes and bands, and trophies.
- **Lat**: shows the degrees of latitude of the approximate center of the island group.
- **Lon**: shows the degrees of longitude of the approximate center of the island group.

Note that Flat Island is listed above with its IOTA reference as "Flat (AF-049)" since there are other Flat Islands elsewhere in the world, in other IOTA island groups. Numerous islands are named after sea birds such as penguin and albatross, for obvious reasons, not to mention seal, crab and so forth: feel free to enter any such terms into the <**Island search**> box to list them.

Copyright © 2026 Robert C. Furzer K4CY

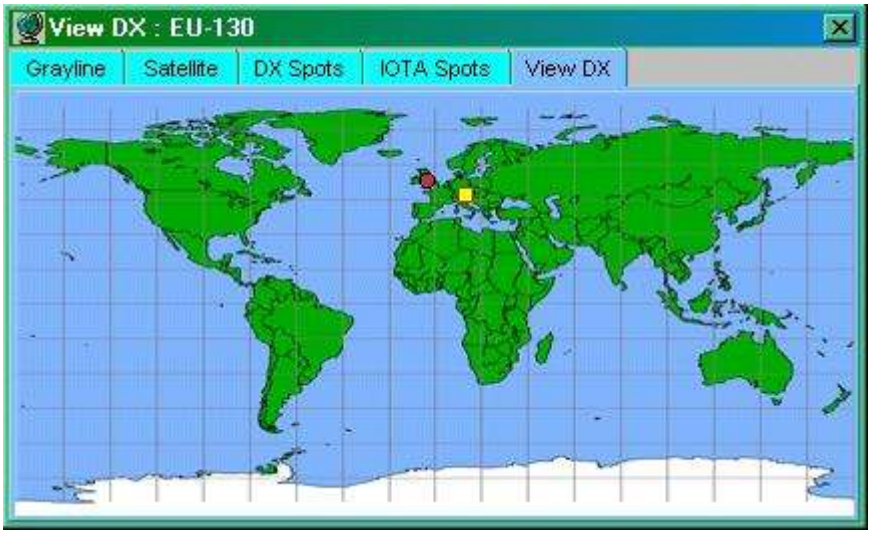


◀ Here, I have entered "Seal" into the Island search box.

Clicking the name in the left pane does two things.

First, all the islands listed in that group appear in the right-hand pane.

Second, the main title is highlighted in the listing – in this case[319], highlighted in yellow with a red script. Colored highlights show island groups confirmed and worked.



◀ Having highlighted an IOTA reference, click <**Show**> to produce a map with the nominal center of the island group shown with a square marker.

With the map displayed, click any island group in the table above to show where it is.

Right-click the map to customize it *e.g.* change the colors and map projection.

[319] Yes, the colors are user-configurable.

Copyright © 2026 Robert C. Furzer K4CY

18.6.2 Download and sync IOTA records

Logger32 can synchronize the <**IOTA Credit Granted**> flags for QSOs in your log to reflect their status in the official IOTA database, updating your IOTA statistics accordingly.

1. Login to the IOTA World website.
2. Open the download QSO page (under <**MY IOTA**>, click <**DOWNLOAD QSOS**>) ▼



Click where shown to download your QSOs in CSV format. A data file called *export.csv* is generated and sent to your browser's default download folder, or to a folder of your choice ▼

```
"Ref. No.","Callsign","UTC","Mode","MHz","Count for","Method","Status"
"AF-001","3B6RF","2001-05-12 08:41:00","CW","28","HF bands","QSL","Accepted"
"AF-002","FT5ZB","1988-07-11 06:51:00","CW","21","HF bands","QSL","Accepted"
"AF-003","ZD8Z","1994-01-08 08:34:00","CW","14","HF bands","QSL","Accepted"
```

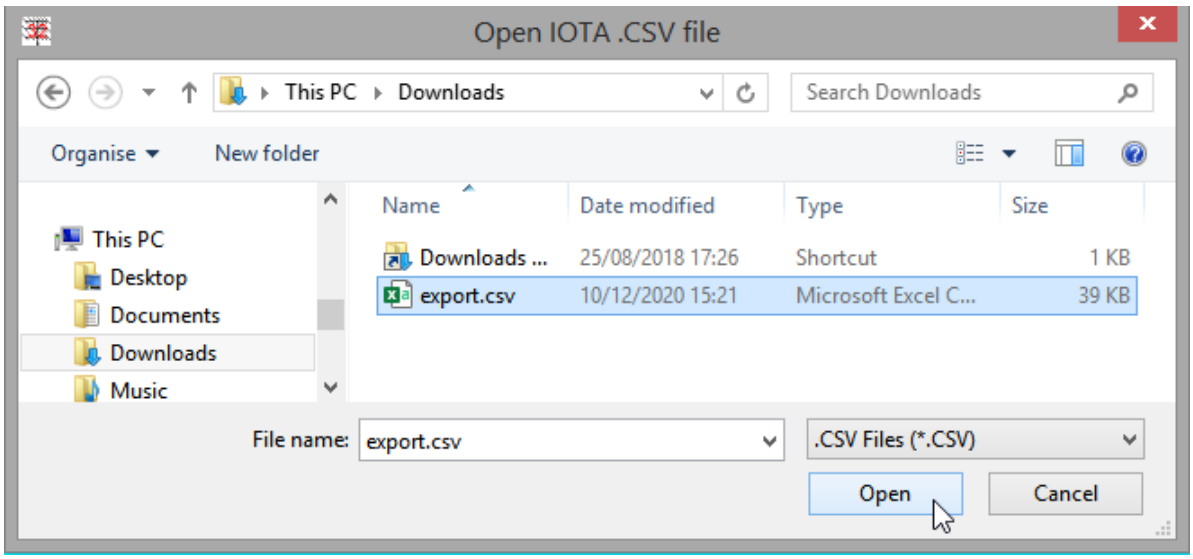
The data file *export.csv* contains a line for each QSO:

- IOTA reference № in the preferred IOTA format (a 2-letter continent ID followed by a hyphen then a 3-digit numeric identifier).
- Callsign of the IOTA station you worked.
- The date and time of the QSO in UTC.
- The QSO mode and band (MHz).
- The relevant IOTA award which this QSO counts towards.

Copyright © 2026 Robert C. Furzer K4CY

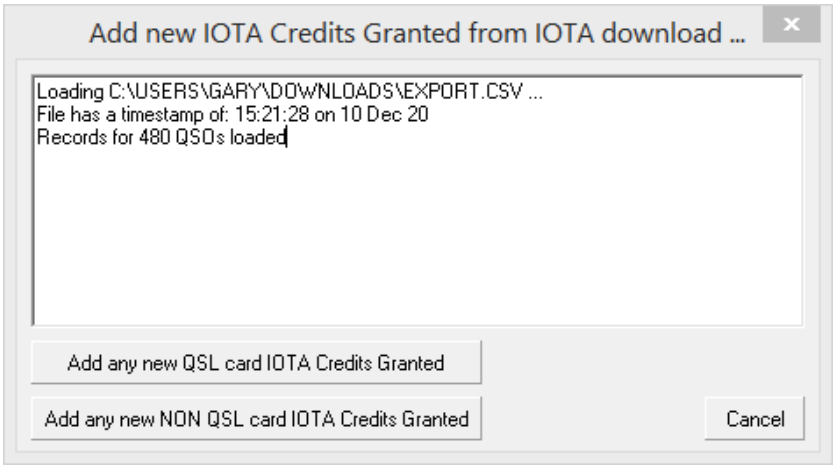
- Confirmation method *i.e.*:
  - QSL - this QSO has been credited for IOTA based on a QSL card.
  - Contest – this QSO has been credited from the IOTA station's IOTA contest log.
- Status *i.e.*:
  - Accepted – the QSO has been credited to the specified IOTA award based on a matching QSO in the IOTA station's log and your log in the Club Log database.
  - Not submitted - not [yet] credited.

3. In Logger32, click **File ⇨ Synchronize IOTA (from a CSV file)**.
4. Navigate to the download folder where *export.csv* file went ▼



5. Click to select *export.csv* then click <**Open**>.

   Logger32 reads the data file into a buffer, checking and counting the QSO records as they are loaded ▶

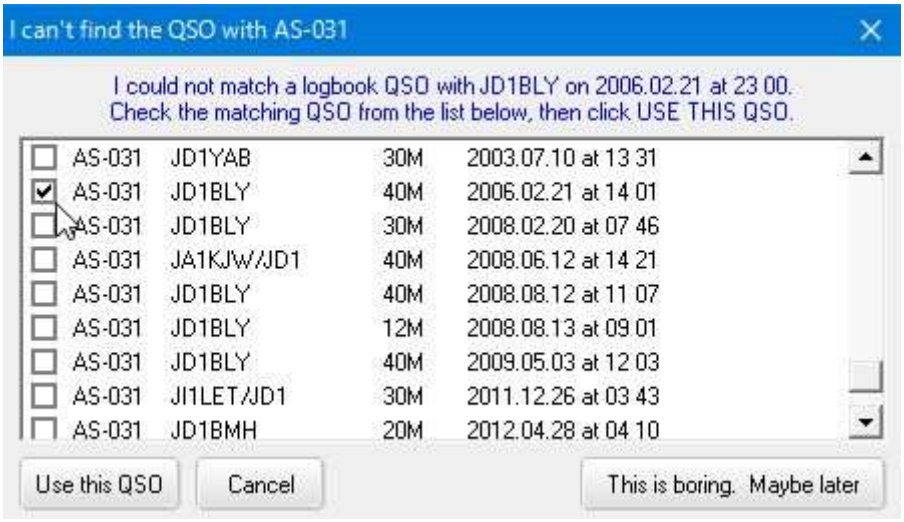


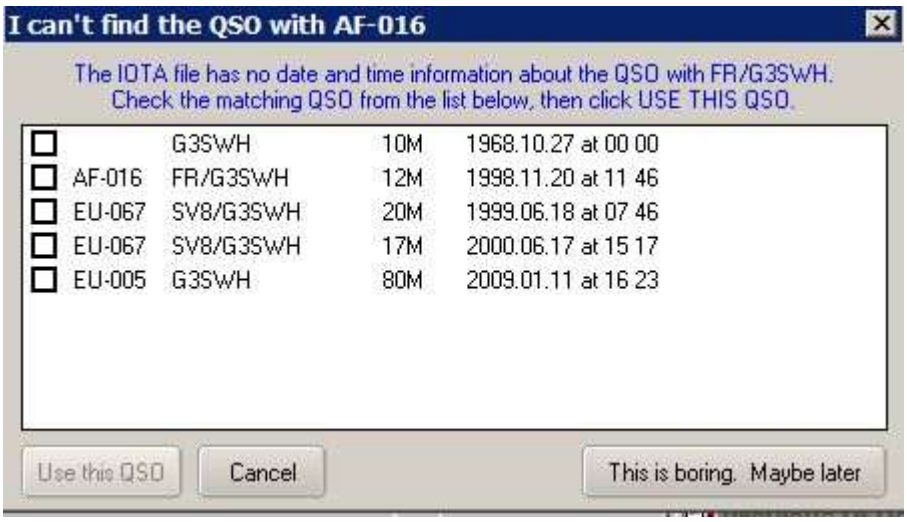
- Click <**Add any new QSL card IOTA Credits Granted**> to mark any QSOs in your log for which you have submitted QSL cards (cards which have been accepted as valid by an IOTA checkpoint) with the <**Credited for IOTA**> flag.
- Click <**Add any NON QSL card IOTA Credits Granted**> to do the same but only for QSOs confirmed and credited by some means other than checked QSL cards *e.g.* QSOs matched in IOTA contest logs or LoTW.

Logger32 hunts systematically through your open log for the corresponding QSOs to flag.

Copyright © 2026 Robert C. Furzer K4CY

If for some reason Logger32 cannot locate a matching QSO with an IOTA station in your log, it offers any alternative QSOs from your logbook with the same IOTA reference, giving you the chance to select and flag one of them as credited instead ▶



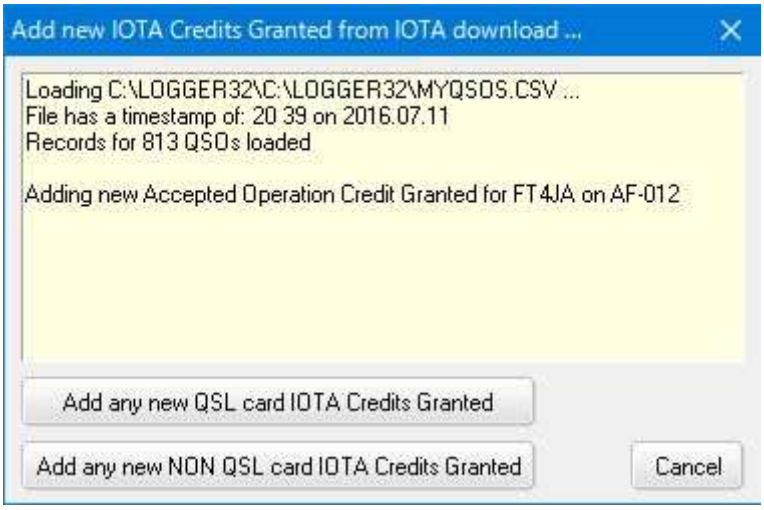


In searching for a match, Logger32 looks 30 minutes either side of the reported QSO time, even checking for dates with days and months swapped over.

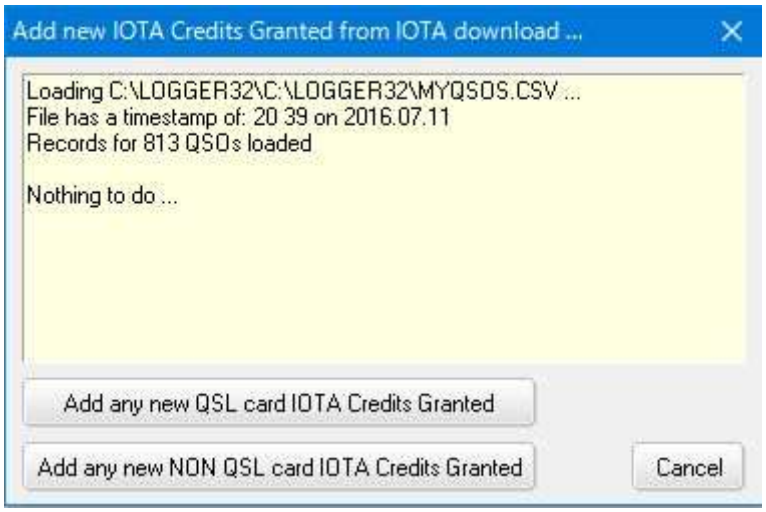
◀ If a record in *export.csv* is missing the date and time, Logger32 looks for possible matches.

6. Click to select (tick) a suitable QSO (if there is one) then click <**Use this QSO**> to flag it as credited. Logger32 tells you what it has done.

Here, just one new credit was found among the 813 QSOs downloaded, since 812 of them had already been flagged ▶



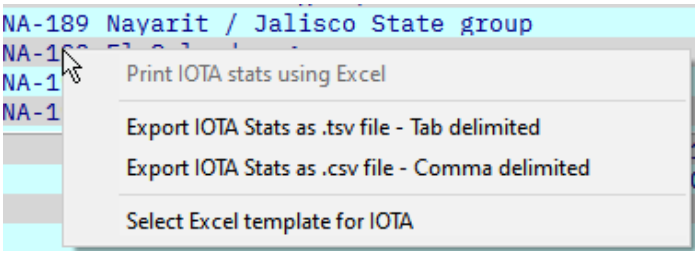
Copyright © 2026 Robert C. Furzer K4CY



◀ If Logger32 can't find *any* QSOs to flag, it lets you know, in Bob K4CY's inimitable style.

18.6.3 IOTA export

Right-click the IOTA award table to export the statistics as text files called *C:\Logger32\IOTA stats.tsv* (with tabs between values) or *C:\Logger32\IOTA stats.csv* (with commas between them) ▶



18.6.4 IOTA manual update

The IOTA award manual update process is essentially the same as for DXCC ▶



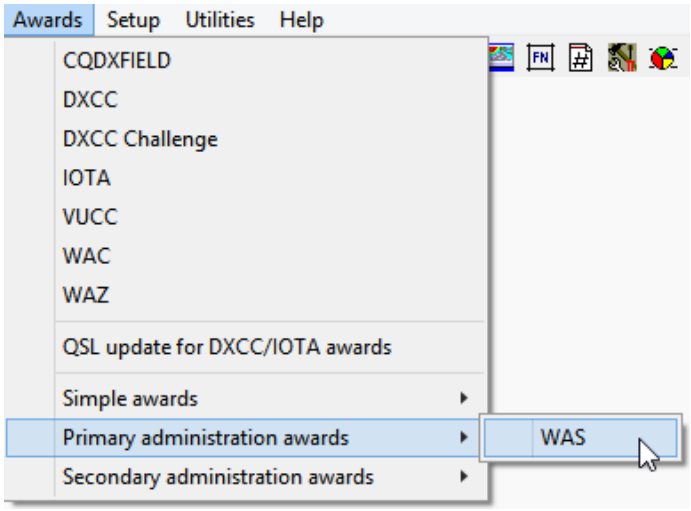
Provided you have flagged relevant QSOs in your log as <**QSO Submitted**> for each IOTA award claim, these will be listed for you, so all you have to do is to check them off one-by-one from a list of QSOs that have been verified, approved and granted to your award.

Copyright © 2026 Robert C. Furzer K4CY

18.7 WAS award

ARRL's **W**orked **A**ll **S**tates award ▶ involves contacting all 50 US states.





◀ **Awards ⇨ Primary administration awards ⇨ WAS** opens a WAS summary with rows for each state and columns for each band, showing the usual **W**orked/ **C**onfirmed markers in the table cells.

Click any filled cell for details of the corresponding QSOs.

Selector panels along the bottom of the report let us specify modes, operators, QSL and credit types, filtering the displayed data accordingly ▼



As with other award summaries in Logger32, the current selections/filters are discreetly saved when we close the WAS summary and automatically re-applied when we next open it.

Hinson tip: if the grid pattern and statistics are not what you expected to see, check those selector panels at the bottom. The summary does not necessarily show 'everything'.

If you have a ADIF contest log from, say, N1MM+ that does not include US states, it is possible to fill-in the missing information using a program such as ADIF Lookup by KO4BB to process the log. ADIF Lookup:

1. Opens the raw ADIF log file.
2. Systematically looks up each logged callsign in an online callsign database such as HamQTH.com
3. Adds missing data (such as states, locators and CQ zones) to the QSOs, where possible.
4. Outputs the updated log as a new ADIF file, ready to import into Logger32.

Copyright © 2026 Robert C. Furzer K4CY

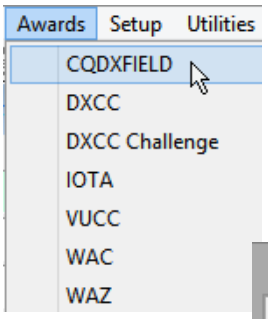
However, the quality of the added information is limited by the quality of the source data. Aside from any data validation performed by ADIF Lookup, errors and omissions in the online callsign database (*e.g.* wrong locations perhaps reflecting reissued callsigns) are likely to end up in the output ADIF.

Hinson tip: tempting though it may be to export your entire log from Logger32 as an ADIF file, process it with ADIF Lookup, then import it straight back into Logger32, don't be surprised to find numerous QSOs with information that is plain wrong. Check the output ADIF file carefully before reloading it into Logger32, for instance using the ADIF editing software ADIFmaster and/or by importing it into a new logbook, leaving your original log intact and pristine. **Take your time! Make a cunning plan! Don't rush into this!**

Hinson tip: if the combination of ADIF Lookup and/or your checking and 'correcting' QSOs somehow turns out to have been a massive mistake, you'll be glad you kept your original log and/or the original ADIF log export file. **Don't say you weren't warned!**

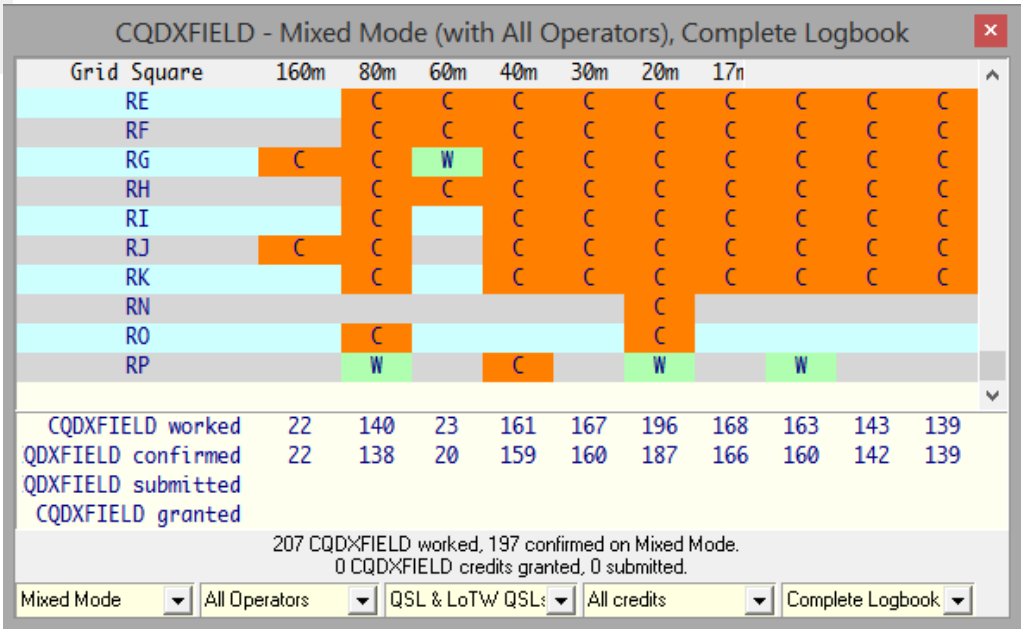
Hinson tip: the same dire warning applies to other activities that involve meddling with multiple QSOs at once. For this very reason, Logger32 is explicitly designed to work on just one QSO at a time and has all manner of data integrity checks and backup prompts built-in, thus limiting the rate and extent of potential damage. **Ignore the warnings at your peril!**

18.8 CQ DX Field award



One of Logger32's built-in awards is the CQ DX Field award for working and confirming at least 50 of the 324 2-letter locator grid fields.

◀ **Awards** ⇨ **CQDXFIELD** generates a report from the open logbook showing the current status like this ▼



As with the GridSquares award, the CQ DX Field report does not list every single 2-letter grid field, just the ones we have **W**orked and **C**onfirmed so far, plus any that have been **S**ubmitted or **G**ranted for the award.

As usual, there are several selections along the bottom of the form to count or ignore various QSOs in the table.

Copyright © 2026 Robert C. Furzer K4CY

18.9 USA Counties Award

Having conquered the **W**orked **A**ll **S**tates award, your next US-centric challenge may be the **USA Counties Award** originally sponsored by CQ Magazine, now run by **M**obile **A**mateur **R**adio **A**wards **C**lub Inc. (MARAC) ▶▼



USACA - Mixed Mode, All op, QSL & eQSL confirmations, QSL & eQSL credits.

rimar	Secondary	econdary subdivisic	160m	80m	60m	40m	30m	20m	17m	15m	12m	10m	6m
AK	First Southeastern	First Southeastern					C	C	C	C	W	W	
AK	Fourth Central	Fourth Central		C		C	C	C	C	C		C	
AK	Second Northwest	Second Northwest						C					
AK	Third Southcentral	Third Southcentral		C		C	C	C	C	C	C	C	
AL	Autauga	Autauga								W			
AL	Baldwin	Baldwin		W			C	W	W	W		W	
		USACA Worked	14	472	3	780	697	1132	1188	1270	492	749	
		USACA Confirmed	5	45		281	49	93	83	110	47	78	
		USACA Credit Submitted											
		USACA Credit Granted											

USACA total - 3077. 1752 USACA worked, 468 are confirmed. 0 credit granted, and 0 submitted.

Mixed Mode | All Operators | QSL & eQSLs | QSL & eQSL credits

USACA involves making and confirming QSOs with US counties plus the 4 judicial districts of Alaska – some 3,077 'counties' to chase[320]. For details, visit CountyHunter.com or contact MARAC.

When importing LoTW confirmations to update your Logger32 log, see the instructions for converting some US locations as indicated by LoTW to their USACA equivalents.

[320] US counties are 'secondary administrative subdivisions' in ADIF-speak, so you'll find USACA listed under **Awards ⇨ Secondary administration awards** … but for convenience the ADIF field name is 'CNTY'.

Copyright © 2026 Robert C. Furzer K4CY

"よくやったネ、Bob
現在、USA County Award は eQSL によるコンファームも有効です。
(LoTW によるコンファームは無効) Logger32 は、eQSL による Logbook の更新を完璧にサポートします。
20 年以上前ですが、当時はまだ Logger16 の時代、USA County Award は"お飾り"程度のサポートで、Bob に改善の要望をしたことがあります。Bob、覚えていますか？"

Aki JA1NLX

18.10 Awards FAQs

Q. My awards table doesn't look right. I'm pretty sure there are credits missing and the totals are wrong. What do I do?

A. First, carefully check the reporting parameters across the bottom of the table. Have you told Logger32 to pick out QSOs made with just one of your callsigns, perhaps, or picked a single-mode award or year? If the problem is with digital modes, check that you have defined the relevant modes as 'digital' under **Tools ⇨ Database maintenance ⇨ Setup phone/digital modes**.

If that's not the problem, recalculate statistics and check again (close and re-open the awards table). Sometimes, for reasons unknown, the statistics lose touch with reality.

Q. I worked and logged P5LID but I'm fairly sure it was not a legitimate P5 station. On the off-chance that it was OK and I receive a confirmation, can I leave it in my log … but not have it count for DXCC (not yet anyway)?

A. With a logged QSO that you know is invalid for DXCC, you can:

- Turn it into an informational QSO by adding an equals sign to the callsign so it becomes **=P5LID**.
- Change the DXCC entity for the logged P5LID QSO by right-clicking the QSO then clicking <**Edit Country info**> and selecting code 000 <**Not accepted for DXCC**> (up at the top of the country list).

- or -

- Simply delete it, expunge it from the record, treat it with the contempt it deserves. Be brave. Bite the bullet and move on. **Hint**: backup your log first.

Your DXCC award statistics are adjusted appropriately either way. Rest assured: having done this, if that was the only P5 in your log, when another P5 is spotted, you *will* be alerted to the new one, and if it ever is legitimately confirmed, you can easily change the informational QSO back to a regular QSO, or correct the DXCC entity number, in order for it to be counted in your DXCC stats. If you deleted it, however, you'll have to manually re-enter the QSO details or recover it from a pre-deletion backup.

Copyright © 2026 Robert C. Furzer K4CY

You still face an ethical dilemma about exporting the logged QSO to your online logs at LoTW, QRZ, Club Log *etc*. If you do export and upload it, it *might* get matched and confirmed, but so long as it remains unmatched, you are in effect falsely claiming to have worked P5. If you do *not* export and upload it, you *may* be missing out on an eventual match and confirmation. It is for you to decide which way to play this.

Q. Why isn't my contact with [some remote, rare or new callsign] showing up under [DXCC, IOTA or some other award]?"

A. There can be a several causes and hence several things to check or do:

- Was the QSO logged with the correct Operator (your station callsign)?
- If you are not using the "Mixed mode" option at the bottom of the award table, have you selected the correct mode for the QSO in question?
- If you are in the habit of editing QSOs in the logbook, try **Tools ⇨ Database Maintenance ⇨ Recalculate statistics|Recalculate statistics & distance** to regenerate the statistics used to compile the awards reports, and optionally also update the short path distances recorded in your log.
- Carefully check the QSO in your log. Are all the data correct, particularly the DXCC entity?
- Does the particular band and mode combination have a "Y" in the statistics column of the Bands & Modes table? If not, the statistics are not updated … so you may want to update your bandplan accordingly.
- Do you have the appropriate logbook open? Logger32 can maintain as many logbooks as you desire, but only one at a time. Do you have the right one open?
- If you are using the 'digital mode' option to aggregate the statistics for QSOs made on various digital modes, are all the digital modes included under **Tools ⇨ Database maintenance ⇨ Setup phone/digital modes**, including any new modes or submodes?
- Does the QSO actually qualify for the award in question? The DXCC CW award, for instance, only counts CW QSOs made after 1974.

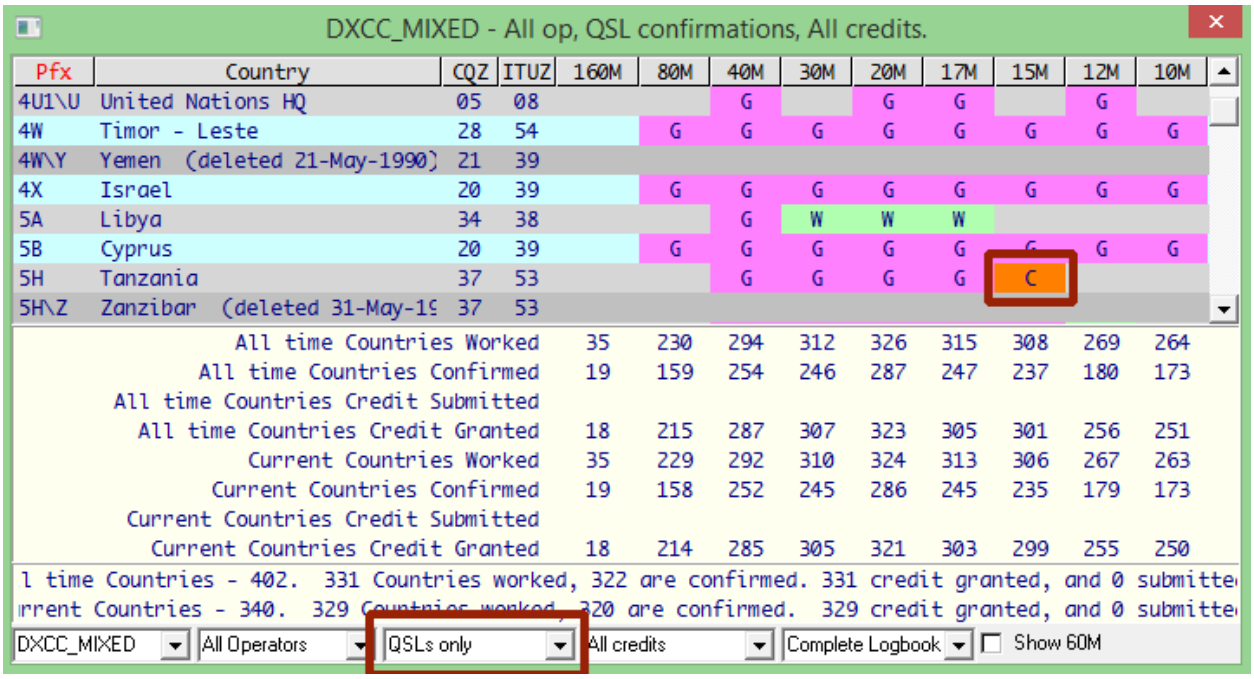
Be aware that CQ zone tables compiled from your logbook may result in erroneous statistical summaries unless you take pains to clarify the status of ambiguous entries. This problem is most likely to occur where CQ zones span political or radio licensing boundaries. For example, all states in the USA 4th call area are in CQ zone 5 *apart from* Alabama in CQ zone 4. In the absence of additional detail when logging a contact, Logger32 treats all 4th area stations as being in Atlanta, Georgia, and accordingly assigns CQ zone 05 to these log entries. Thus, a QSO with W4AP in Montgomery, Alabama will be logged (incorrectly) as zone 05 *unless* you provide more specific locational information. Upon entering "AL" in the State field of the log entry pane, or setting the state for a logged US QSO to Alabama by editing its primary administration in the logbook, the zone entry will be corrected from "05" to "04". If for some reason you later remove the state information from your log entry, the CQ zone for your contact with W4AP will revert to the "05".

Logger32 does not *automatically* use state information from a QRZ lookup, but QRZ and various other online databases (such as the licensing databases, plus HamQTH) plus Google searches and so forth can be used manually to research and identify the likely state for any station … and of course you could ask the station concerned to confirm the details. Good luck hunting through your dusty shoeboxes of QSL cards for confirmations of those elusive states.

Copyright © 2026 Robert C. Furzer K4CY

Q. Can Logger32 help me find and list the QSLs I need to dig out for my online DXCC application?

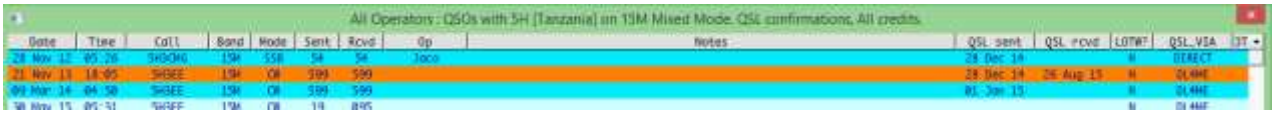
A. Yes! Here's how: start with **Awards ⇨ DXCC ▼**



The **QSLs only** option at the bottom of the report means that any **C**onfirmed blobs on the report (such as the 5H on 15m shown above) are confirmed on QSL cards but *not* on LoTW. This is the critical information you need. Dig those specific QSL cards out of your shoeboxes.

> **Hinson tip**: any QSOs that are *also* confirmed on LoTW should preferably be submitted for DXCC credit through the autuomated LoTW system rather than getting those QSL cards verified manually. It's cheaper, easier and more certain of success.

Click a red blob in the DXCC report to list one or more QSOs with that country on that band. Look for the one or more that has been confirmed by QSL card ▼



Hunt for the relevant QSL card/s and check them carefully, picking out one you believe is valid for DXCC (*e.g.* an authentic, unblemished, unmodified original). Click the confirmed QSO line to see that QSO record in the logbook ▼



Right-click the QSO in the logbook, then click <**Submit for DXCC>** to flag that QSO … and so on for the other red blobs on the DXCC report.

Now generate an ADIF with those flagged QSOs using **File ⇨ Export files ⇨ Export DXCC file**.

Through the Online DXCC Application, upload the DXCC ADIF file, check once more that you have all the cards in the order listed, and submit the online application. Finally, print out and sign the paperwork, then send the completed form and QSL cards off for checking, either to ARRL HQ or to your friendly local DXCC card-checker. Good luck!

Copyright © 2026 Robert C. Furzer K4CY

Q. Why aren't 60m QSOs and QSLs listed on my DXCC reports?

A. 60m QSOs are not [currently] accepted for *any* DXCC awards. They don't count. At all.

However, if you simply want to show how many DXCC entities you have worked and confirmed on 60m out of interest, there is a tick box option at the bottom of the DXCC reports to show the 60m band *as if* it was eligible for DXCC ▼



▲ ▲ ▲ ▲ ▲

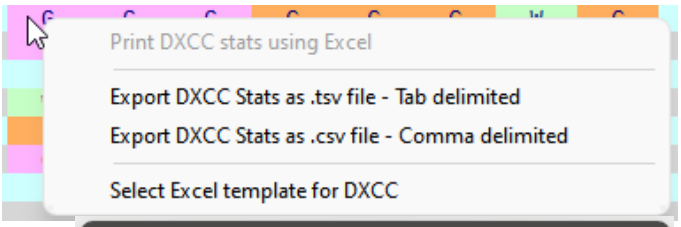
Notice the bottom-line settings on most award reports

Note that you *also* need to have at least one Y in the 60m STATS column in your band plan to collect the 60m statistics. If not, add it *and* recalculate the statistics.

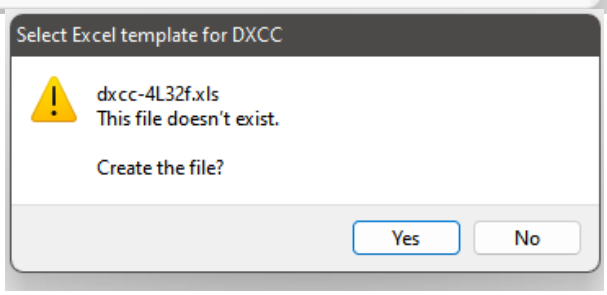
Q. Can I get my DXCC statistics into a spreadsheet?

A. Yes. Here's how:

1. Open the DXCC report using **Awards ⇨ DXCC**. Logger32 takes a moment to create it.
2. Right-click in the body of the report (in the area of the colored blobs) to open a menu ▼
3. Click <**Select Excel template for DXCC**> at the bottom ▶



Logger32 may try to open a non-existent Excel template giving a message like ▶



If so, click <**No**>, then either find and select the default Excel template *C:\Logger32\doNotDelete.xls* instead, or find and select a customized version of that such as one downloaded from the Files area of the Hamlogger Groups.io site that was originally created and kindly shared by Javier EA1AUS (SK) – download the zip, unpack it to a convenient folder such as *C:\Logger32*, then find and select it using the bottom-line menu option in Logger32.

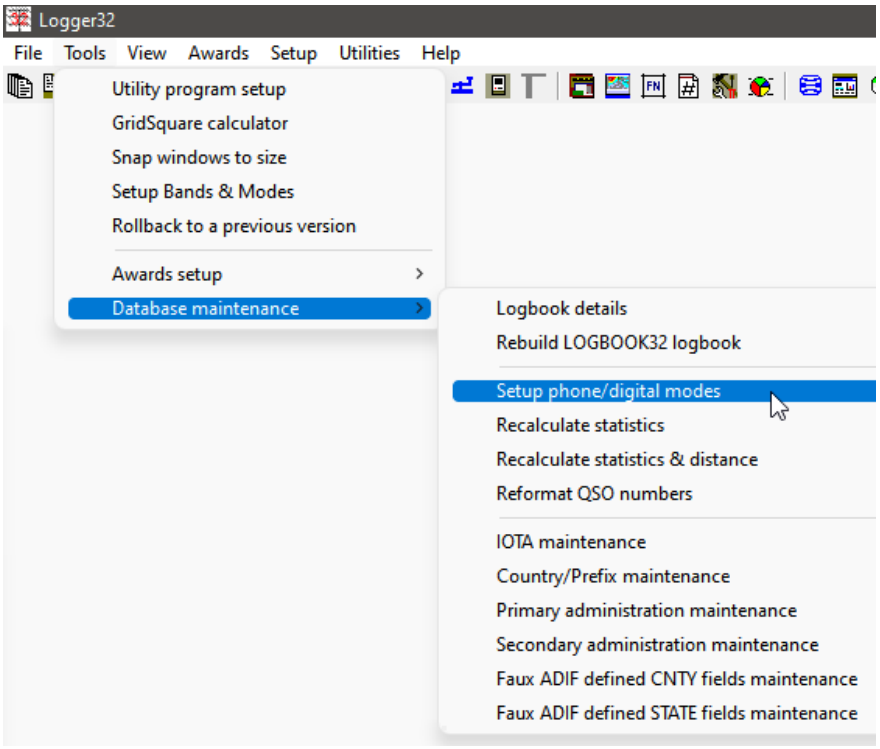
4. If <**Print DXCC stats using Excel**> is grayed out on the right-click menu, close Logger32, remove the line *Check for Excel=No* from *C:\Logger32\Logger32.ini*, re-start Logger32 and find your way back to where you were.
5. Now the <**Print DXCC stats using Excel**> should open the selected template spreadsheet, populate it with your DXCC statistics data, and open the Excel print preview. You can either go ahead and print it out as-is, or cancel the print preview and fiddle around with the sorting and filtering options, colors *etc*. in the spreadsheet, as you wish.

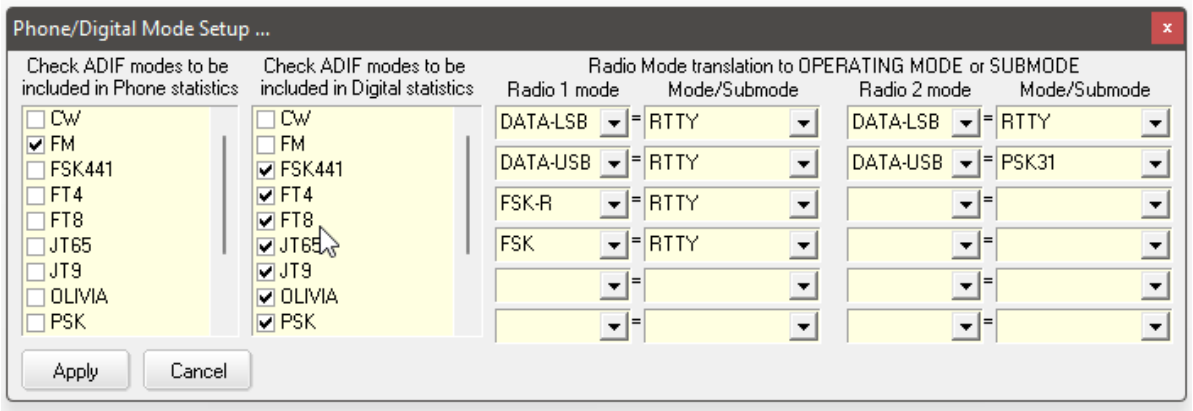
Copyright © 2026 Robert C. Furzer K4CY

Q. Having imported an *.adi* file from JTDX, why aren't my FT# QSOs showing up in the DXCC_DIGITAL report?

A. You need these three things:

1. FT8, FT4 and FT2 in your *C:\Logger32\ADIFmodes.txt* file, telling Logger32 that the modes exist.
2. At least one "Y" in the Stats column against *any* FT# line in your Bands & Modes table, telling Logger32 to collect the statistics for that particular mode *e.g.* FT2 on 6m, FT4 on 10m and FT8 on 80m,
3. A tick against each FT# in the digital statistics column under **Tools ⇨ Database maintenance ⇨ Setup phone/digital modes**, ▼





Copyright © 2026 Robert C. Furzer K4CY

Q. Why do some of my IOTA credits have no date/time?

A. If you received IOTA credits prior to August 2007, *export.csv* may contain QSO records with "00000000" for the UTC, two extra columns (Freq and Mode), "Method" instead of "Status" and a "Status" heading for the righthand field … like this ▼

	A	B	C	D	E	F	G	H
1	Ref. No.	Callsign	UTC	Count for	Freq	Mode	Method	Status
2	AF-001	3B6RF	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted
3	AF-002	FT5ZH	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted
4	AF-003	ZD8T	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted
5	AF-004	EA8CN	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted
6	AF-005	D44BS	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted
7	AF-006	VQ9ZZ	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted
8	AF-007	D68BW	0000-00-00 00:00:00	HF bands	?	?	QSL	Accepted

The reason is that the IOTA rules changed in August 2007. Previously, dates and times were not recorded, and some of the other fields were used differently.

Q. What are the "Active" QSO records at the bottom of *export.csv* ▼?

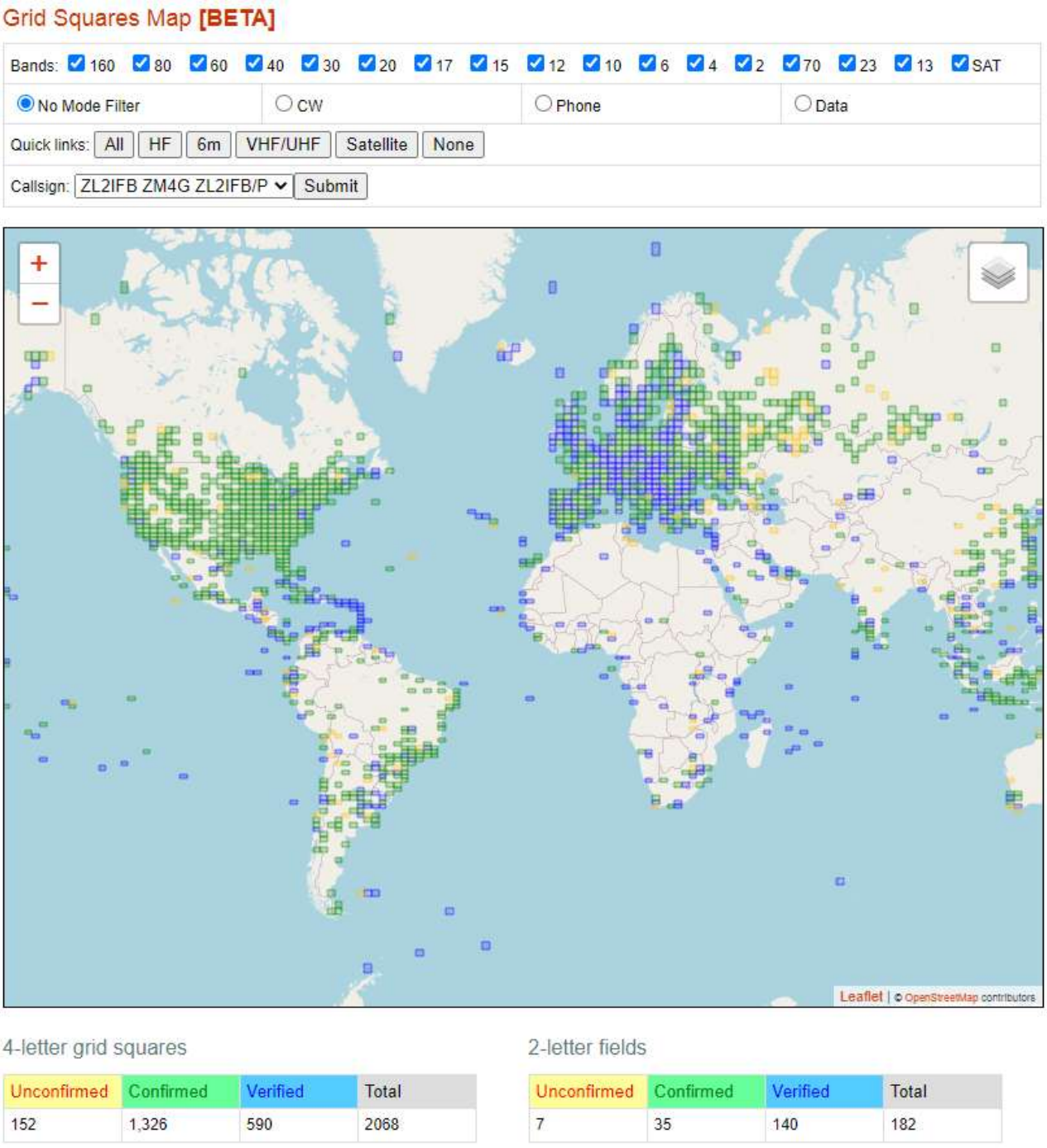
835	NA-112	KS4S	19/06/2004 11:49:00	HF bands	QSL	Active
836	NA-122	HI1UD	29/01/2017 11:34:42	HF bands	QSL	Active
837	NA-147	J3/VE7ACN	04/12/2016 22:01:00	HF bands	QSL	Active
838	OC-092	DU2/JA1PBV	06/06/2017 12:12:00	HF bands	QSL	Active
839	OC-119	DU1UD/8	22/04/2017 7:44:00	HF bands	QSL	Active
840	OC-202	DZ4C	21/04/2017 12:32:00	HF bands	QSL	Active
841	OC-226	V6J	01/07/2017 3:52:00	HF bands	QSL	Active
842	OC-266	VK5MAV/6	08/09/2016 22:44:00	HF bands	QSL	Active
843	OC-267	VK9MAV	18/06/2017 7:40:33	HF bands	QSL	Active
844	SA-033	HD2RRC/4	30/05/2017 7:30:00	HF bands	QSL	Active
845						

A. "Active" QSOs in the data file are *eligible* for IOTA credit but have not yet been claimed or awarded. Logger32 does *not* set the <**IOTA credited**> flag for these QSOs in your log because, until they are claimed, they obviously cannot be credited towards your IOTA awards. The IOTA award administrators are not mind-readers! If you have not even applied for your very first IOTA award, *all* the QSO records in *export.csv* will of course be marked "Active".

Copyright © 2026 Robert C. Furzer K4CY

Q. Can I plot the grids I have worked on a map?

A. Yes … but not within Logger32. The Grid Squares function in Club Log works nicely, plotting differently-colored blobs for worked, confirmed and verified grids ▼



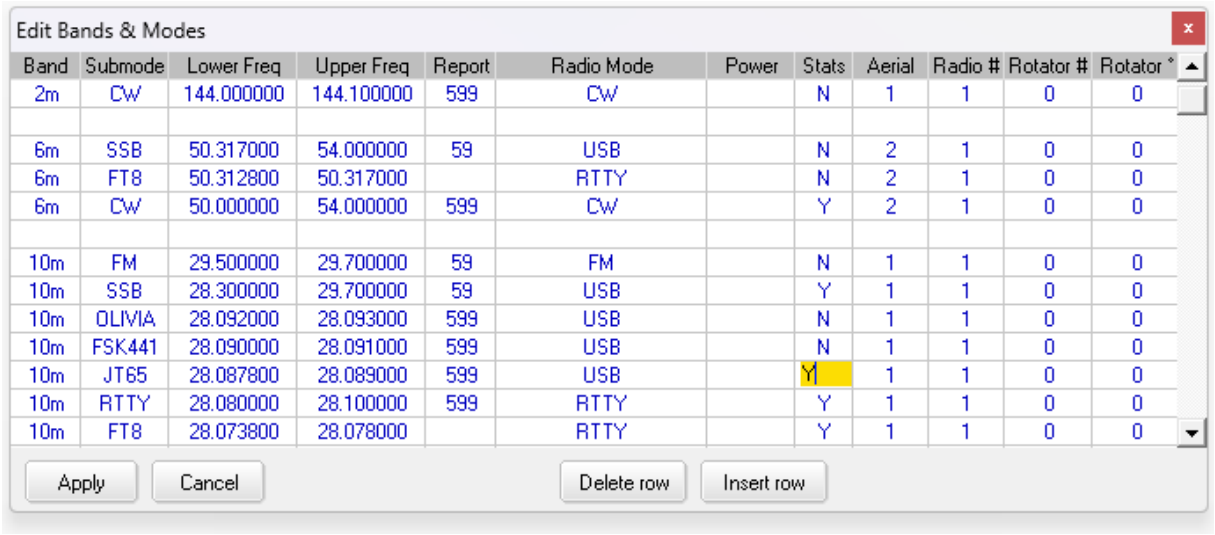
To run the grid mapper, login to Club Log and click the Grid Squares button on the left side menu. It colors the grids either in your uploaded log or determined from other sources (*e.g.* consensus from other Club Log logs, or small DXCC entities that lie entirely within a known grid square). Click the + or – button to zoom in or out, and drag the map around to see as much of the world as you want to see. Using the filter options at the top, you can show grids worked on specific bands and/or modes, while handy summary tables at the bottom show your unconfirmed, confirmed and verified QSO counts for both 4-letter grid squares (*e.g.* FN35) and 2-letter fields (*e.g.* FN). Click any blob for details on a QSO.

Copyright © 2026 Robert C. Furzer K4CY

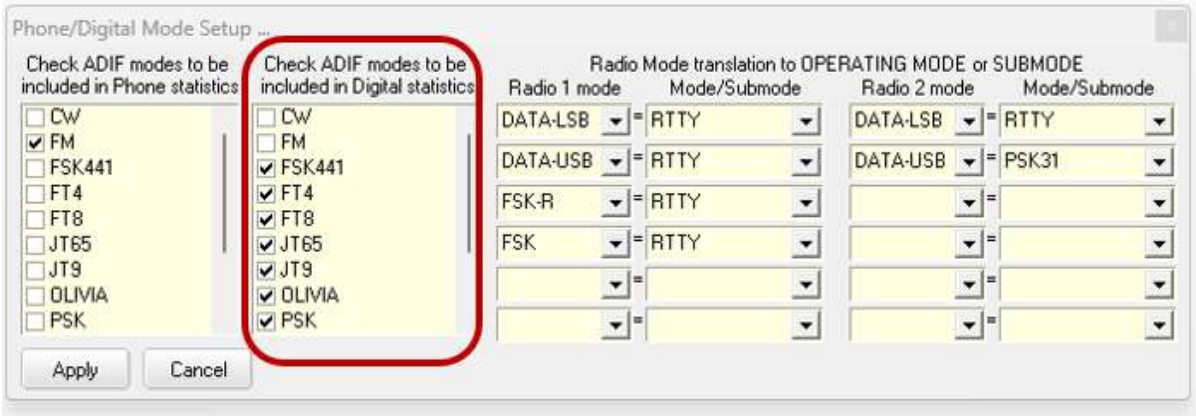
Q. Why aren't the grids I have worked on 6m JT65 showing up in my CQDXFIELD Digital Modes table?

A. Check these three areas:

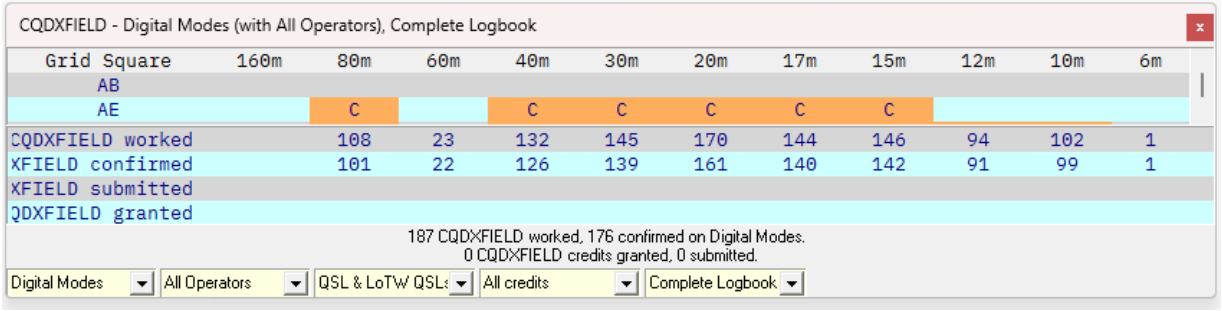
1. Logger32 needs to know that JT65 is a mode for which you want to keep statistics … so at least one JT65 row of the Bands & Modes table must have a Y in the Stats column, on any band *e.g.* ▼



2. Logger32 needs to know that JT65 QSOs count as digital … so make sure JT65 is ticked among the ADIF modes and submodes you consider to be digital ▼



3. Your CQDXFIELD award report needs to count the 6m digital QSOs you have made … so check the filtering options at the bottom of the award report ▼



Copyright © 2026 Robert C. Furzer K4CY

Q. Why doesn't Logger32 know the IOTA reference for the DXpedition to TX9A?

A. TX is an ambiguous prefix for *one* of the French overseas territories, but *which* one?

As this is a DXpedition listed at DX-World.net, the latest DXpedition information should tell Logger32 that it is the Austral Islands. However, as the IOTA reference is not also provided in the DXpedition data file from DX-World, Logger32 remains in the dark about their IOTA.

The TX9A DXpedition website says their IOTA is OC-152.

If you haven't worked the Australs before, DX spots for TX9A should be highlighted as a new DXCC in the normal manner, dinging your audio alerts *etc*. as usual. However, if you have worked the Australs but not OC-152 before, DX spots will be shown in bold for a listed DXpedition but will not be highlighted or dinged as a new IOTA.

A workaround is to edit the downloaded DXpeditions data file locally, adding the IOTA reference to the end of the relevant line:

1. Download the latest DXpeditions file if you haven't already done so this month using **Setup ⇨ Updates ⇨ Enable automatic download of DX-World DXpeditions list**.
2. Open the current month's downloaded data file in a plain text editor such as Notepad or TED. The data file for DXpeditions QRV during May is called *C:\Logger32\DXpeditions 05.txt* (where 05 is the current calendar month number – adjust as appropriate for other months).
3. Find the relevant line for the DXpedition. In this case, it is:

TX9A^45772^45777^https://www.dx-world.net/tx9a-austral-islands-dxpedition-202/

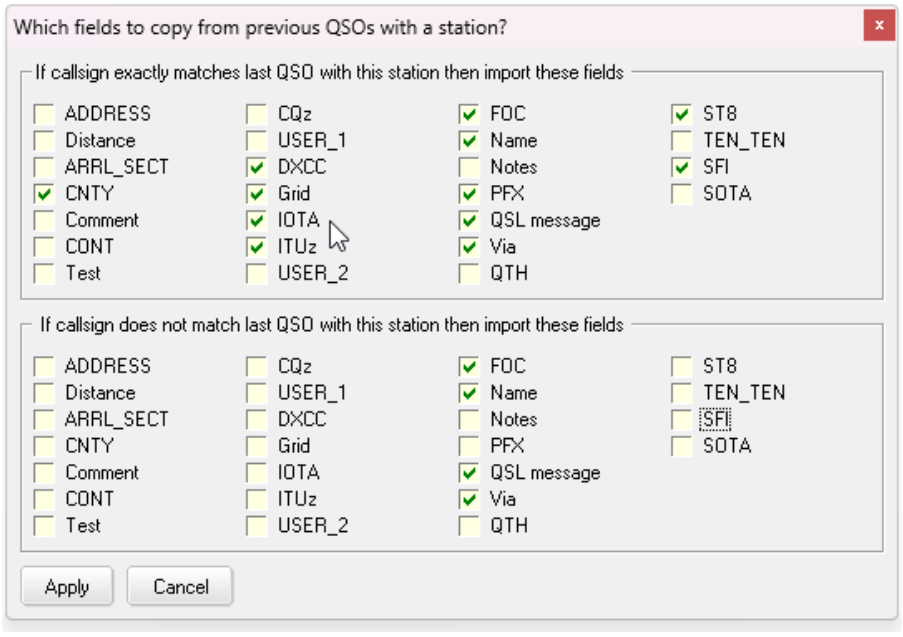
4. Append a circumflex (^ kappie) followed by the IOTA reference in the correct format *i.e.*

TX9A^45772^45777^https://www.dx-world.net/tx9a-austral-islands-dxpedition-202/**^OC-152**

5. Save the edited file and exit the editor.
6. Stop and re-start Logger32 to pick up the modified DXpeditions data file.
7. Rejoice when TX9A is spotted on an open band with an alert for the new IOTA.

Alternatively, simply watch out for those TX9A spots and work them when you can.

Once logged, you can add the IOTA reference OC-152 to your first QSO and then any subsequent QSOs should automatically carry forward and log the same IOTA reference provided you have ticked the IOTA box on the QSO mask ▶



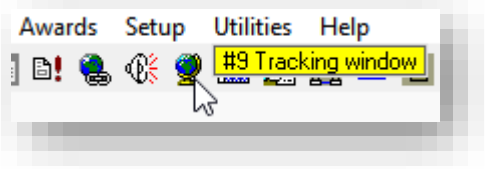
Copyright © 2026 Robert C. Furzer K4CY

19 Tracking window (maps)

"Not all those
who wander
are lost"

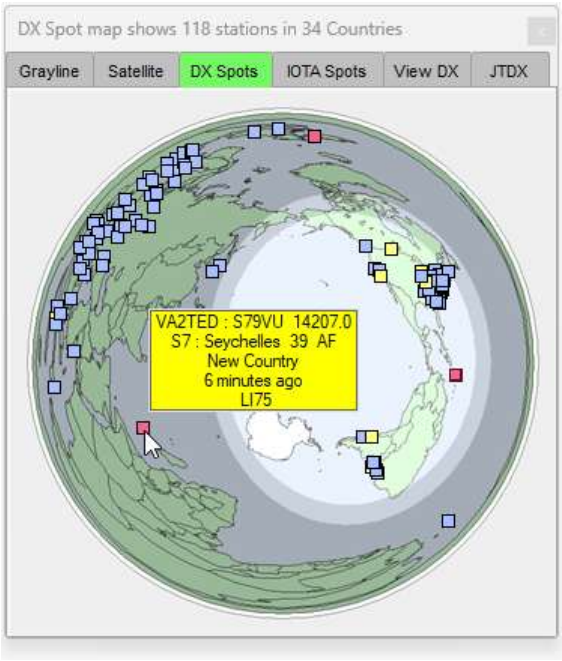
J.R.R. Tolkien

Open the tracking window using globe icon #9 on the toolbar ▶



The Tracking window displays a set of six world maps (one per tab) overlaid with additional info:

1. **Grayline tab**: shows the transition area between day and night. The grayline overlay is also available as an option on the other tabs.
2. **Satellite tab**: displays selected satellite orbital data, paths and footprints.
3. **DX Spots tab**: shows the locations of DX stations spotted on DX cluster, with yellow 'tooltip' popups showing additional information when we cursor over any of the color-coded spots ▶
4. **IOTA Spots tab**: shows the locations of IOTA island stations spotted on DX cluster with "IOTA" in the spot comments.
5. **View DX tab**: shows the location of the station we are currently logging in the log entry pane.
6. **JTDX tab**: shows the locations of digimode stations recently decoded by JTDX on the PC.



Whereas Logger32 calculates where the sun is directly overhead and the grayline position algorithmically, it relies on other information for various locations (including your own!), hence the marked positions are approximate. There are various sources of error *e.g.* a station spotted on DX cluster as "W1ABC/P5" could be a US station portable in the 5th call area or an intrepid DXpeditioner in North Korea. As to the station's precise location within the call area or country, Logger32 can't tell without additional information such as a Maidenhead locator (grid square) – and there's no guarantee even that is correct: some land-based stations claim to live in locators that are way out at sea, or somewhere entirely different in the world.

Copyright © 2026 Robert C. Furzer K4CY

Furthermore, the maps themselves are merely visual two-dimensional *representations* of the Earth's true four-dimensional geography. Viewers out in space would at most see just the facing hemisphere, less areas in darkness or under cloud.

In summary, Logger32's maps are just a guide, close enough for most amateur radio purposes, on LF/HF anyway. On UHF and microwaves, where amateurs need to focus their pencil-thin beams on each other, technologies such as GPS and sighting telescopes are preferable for greater accuracy and precision.

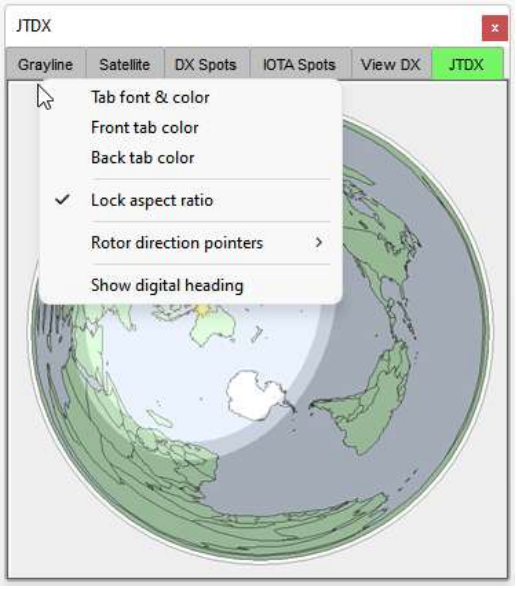
19.1 The tabs

Click any of the tabs to open a map.

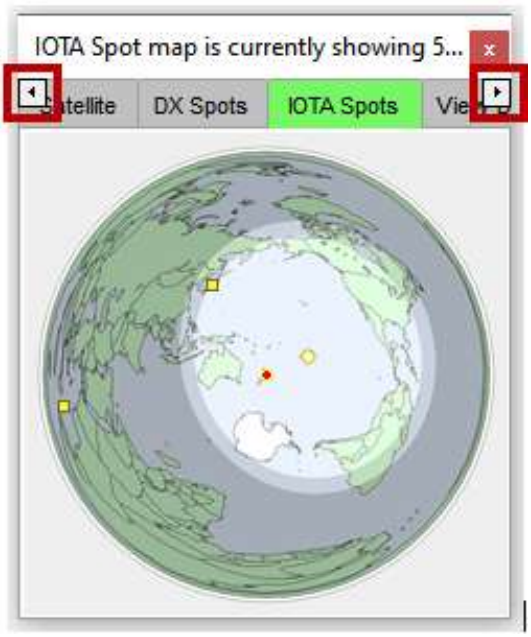
Hinson tip: the kind of map and overlay information I need depends on what I happen to be doing at the time. I might be chasing DX on a wide-open band, hunting IOTA operations, working the satellites, looking for grayline DX on the low bands, turning my beam to check the long path or whatever. By preconfiguring all six tabs to my liking, I can easily swap to a different world view on a whim. For instance, whereas I generally prefer great circle maps with the compass, one tab shows a conventional rectangular map, and one de-cluttered great circle map has no compass.

19.1.1 Tab right-click options

Right-click any tab (the gray tab itself, not the map) to open a little menu of options that apply to *all six* tabs ▶



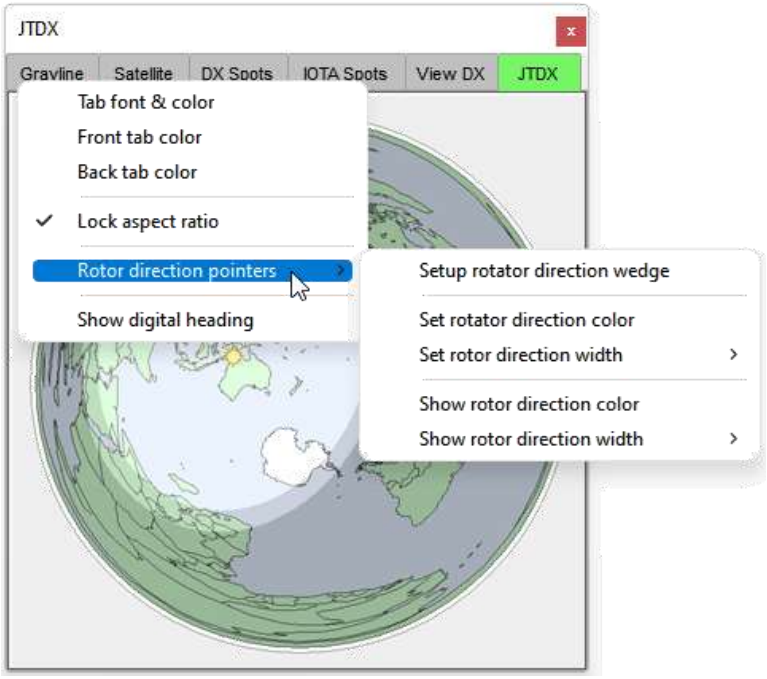
- **Tab font & color**: choose the text font for the tab labels, and the text color.
- **Front tab color**: choose the color for the front (selected, active, open) tab – preferably something bright to distinguish it from the remaining (unselected, inactive, closed) tabs.
- **Back tab color**: choose a relatively bland, inconspicuous color for the back (unselected, inactive, closed) tabs *e.g.* boring gray.



◀ By the way, the *tabs* don't change size as you re-size the maps, so if you shrink the window beyond a certain point, little arrows appear to shift the tabs left and/or right as appropriate.

Copyright © 2026 Robert C. Furzer K4CY

- **Lock aspect ratio**: with this setting enabled (ticked), Logger32 locks the map window shape according to the type of map projection - square for the great circle maps or rectangular in the conventional proportions for Miller projections. Drag an edge or corner to resize the window, retaining the shape. With it *un*ticked, you can drag the sides separately to make the maps low and wide or tall and skinny to fit a convenient space on your display maybe, ignoring the distorted proportions.



- **Rotor direction pointers**: lines can be drawn in your choice of color and width, showing which way your connected digital antenna rotator is or should be pointed to beam at the DX ▶

- **Show digital heading** overlays an azimuth value in degrees relative to true North.

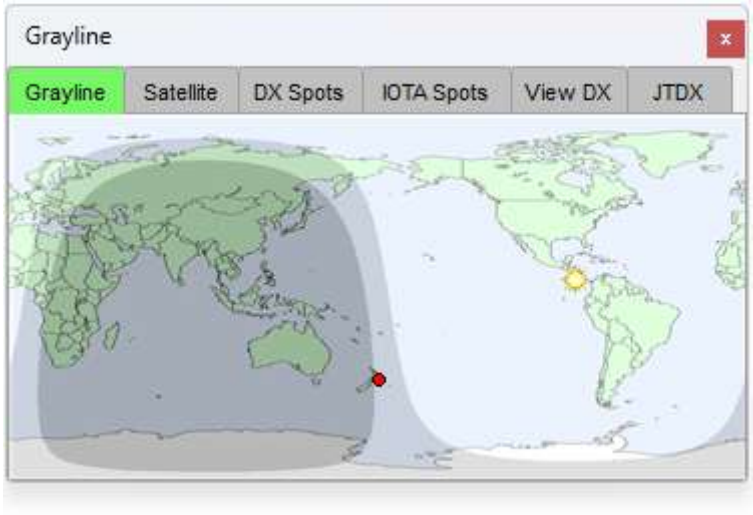
19.2 The maps

The Tracking window shows multi-layered images comprising a static base layer (a world map) overlaid with additional information (such as the overhead sun position, day/night, grid lines, compass and rotator directions – some of which changes dynamically, the overlays being updated every so often) to build up the images shown.

19.2.1 Base maps

Each tab's underlying base layer can show one of three types of map:

1. Logger32-generated **equidistant cylindrical projection** (a 'wall map' style).



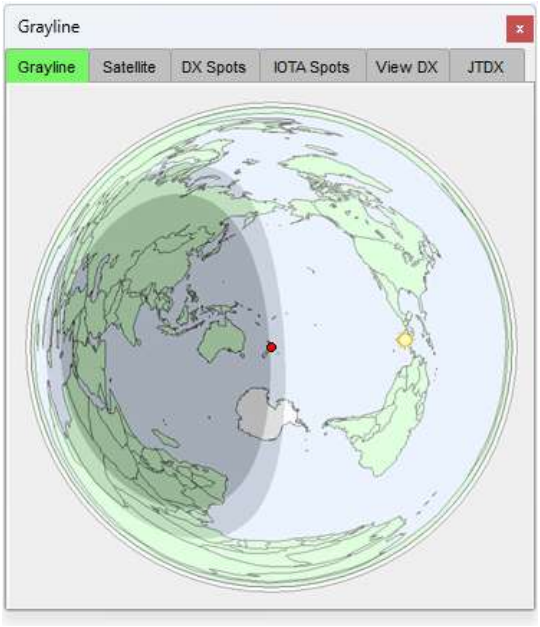
This represents the near-spherical globe as a rectangular image, as if the Earth was a cylinder ▶

Although we are familiar with this world view (actually, the narrower Mercator projection is more common), it is (like all maps!) a two-dimensional topographical representation of the three-dimensional globe, a distortion to fit the map space.

Copyright © 2026 Robert C. Furzer K4CY

Whereas the cylindrical projection quite accurately represents places directly North or South of a given location, countries and features near the equator appear shrunken while those nearer the poles are expanded. Equatorial Zaire, for instance, *appears* to have about half the land area of Greenland, whereas they are very similar sizes in fact. Antarctica is no small continent at 14 million square kilometers, but the white area at the bottom of the map is disproportionately large compared to, say, Africa's 30 million square kilometres.

2. Logger32-generated **Azimuth Equidistant projection**, known as 'great circle' maps among radio amateurs and others ▶

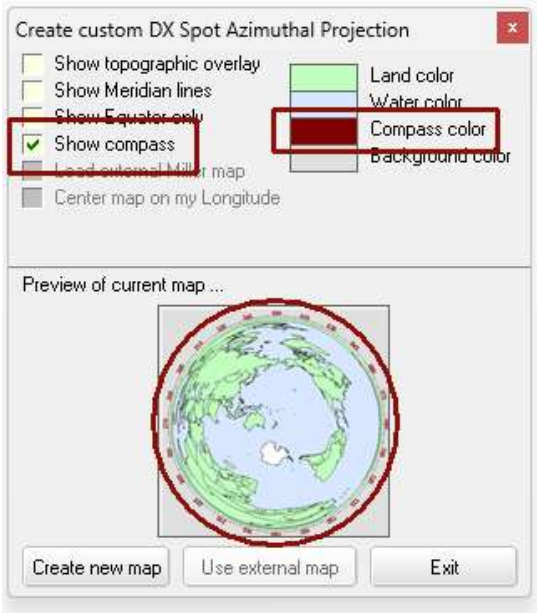


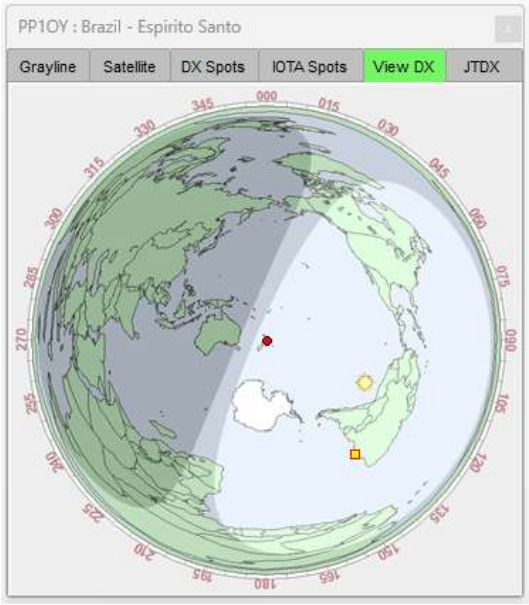
The shapes on a great circle map are even more distorted than on a rectangular Miller map *but* this projection accurately [321] represents both great circle directions and radial distances from the center point – normally our location.

With the map centered on our station location (which it is, by default, for great circle maps generated by Logger32), we can easily tell which way to point the beam – either directly to the DX station along the 'short path', or 180° offset for the 'long path'. The outside edge is about 20,000 km from the center.

Hinson tip: as seen here, the great circle projection is particularly useful for LF DXers around dawn and dusk as it shows the ever-changing direction, orientation and extent of the grayline. DX stations, even on the far side of the Earth, may be workable (briefly!) while the grayline links us together via the ionosphere.

A useful feature of the great circle maps is the "compass" - a set of numeric bearings arranged around the world perimeter in whatever color you like ▼





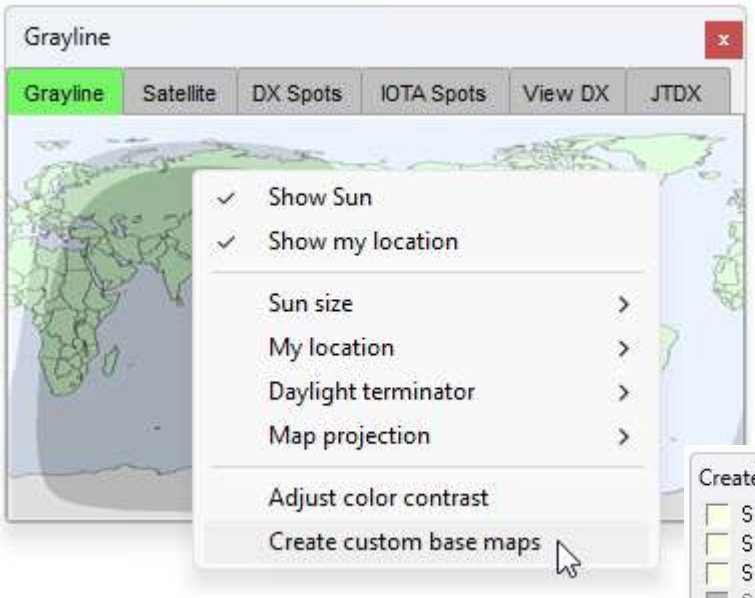
[321] The map image should be circular: if yours is oval, drag one side of the window to correct it. Better still, right-click any tab the click to tick <**Lock aspect ratio>** and then adjust any map side or corner a little to trigger a truly circular re-draw.

Copyright © 2026 Robert C. Furzer K4CY

3. If you prefer, you can supply an 'external' [322] Miller cylindrical projection map image downloaded or generated independently of Logger32 in an image format such as *.bmp*, *.gif* or *.jpg*. It can be of any size - Logger32 will shrink or expand the image as necessary. Given the choice, pick one with at least the resolution needed for the largest size map you expect to view *e.g*. filling the entire screen. As with aerials, bigger is better since resolution can be poor if a small picture is over-stretched.

> **Hinson tip**: each tab has its own base map, configured and customized for that tab. I prefer great circle maps mostly, apart from Miller-style greyline and satellite maps. Other than that, I find it helps to configure the same land and sea colors but options such as the topographic overlays may vary between the tabs. If the current map display doesn't suit me, I either click to open a different tab with different settings, or right-click the map to reconfigure it.
>
> By the way, the map configuration options (colors *etc.*) aren't saved and retrieved by Logger32, so you might like to make a note of them if you want to recreate them at some later date. Otherwise, it's easy enough to try out different settings until you're happy, then stop.

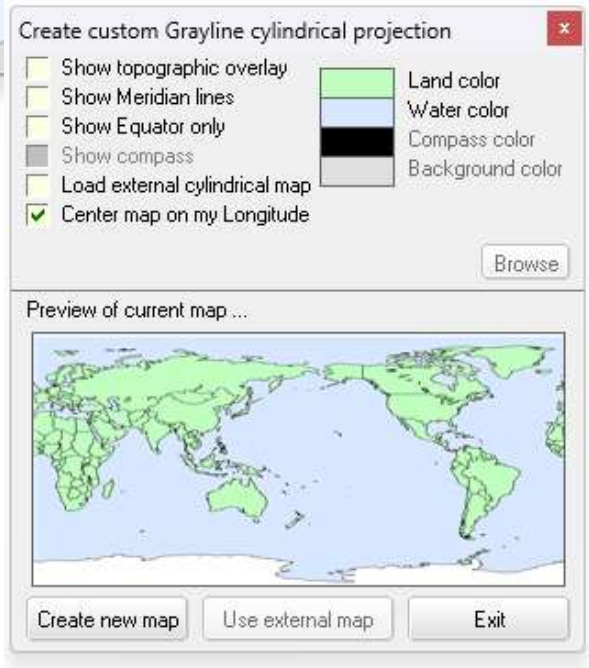
19.2.2 Customize base maps



◀ Right-click any map for its configuration options.

Click <**Create custom base maps**> to customize the world map image base layer for whichever tab is currently open (*each tab has its own base map*).

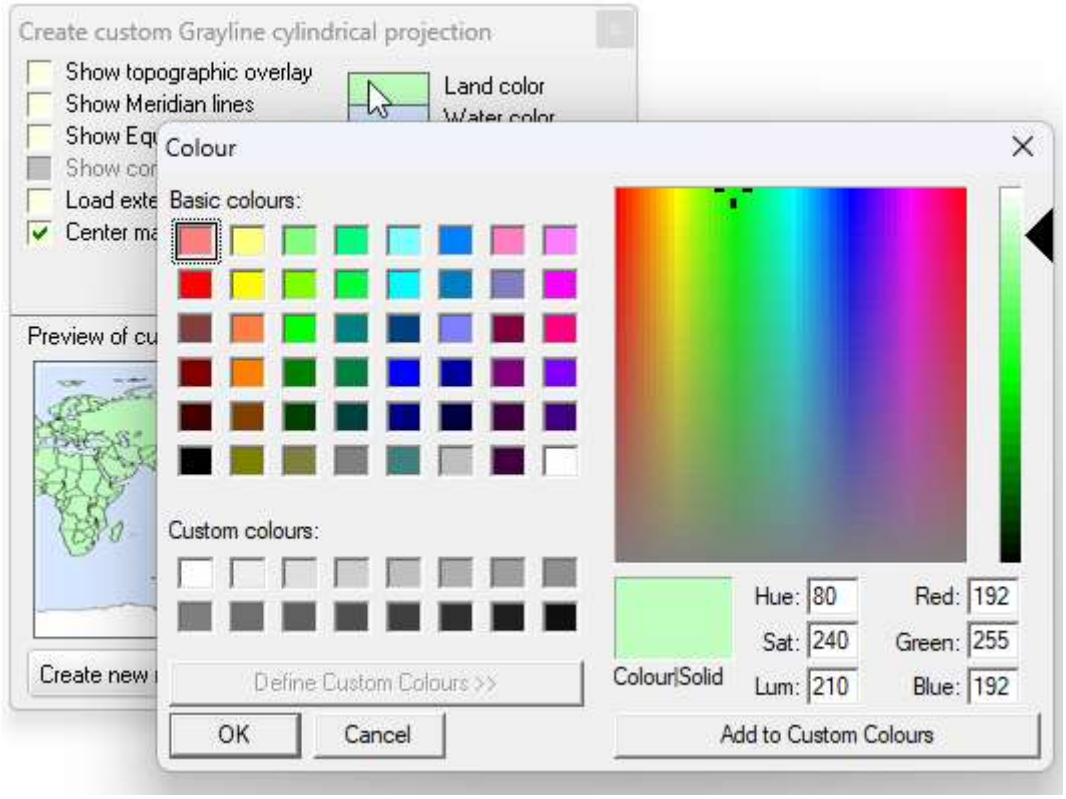
The base map customization form opens ▶



[322] 'External' means 'outside Logger32' *i.e*. not provided as part of the Logger32 installation package.

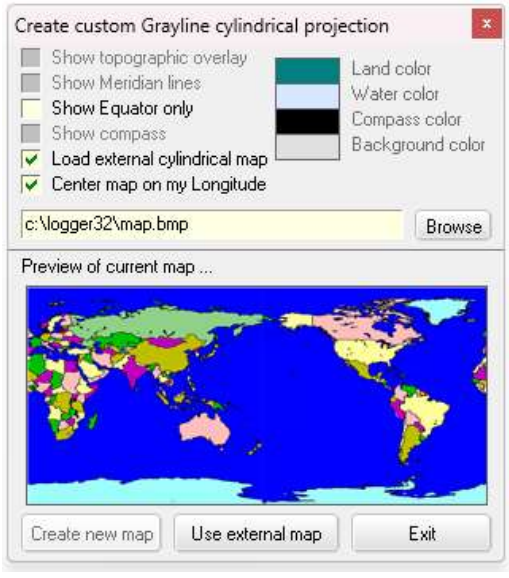
Copyright © 2026 Robert C. Furzer K4CY

On the customization form, you can choose the **colors** for the land, sea, background and (for great circle maps) the compass rose around the circumference. Click a colored rectangle to open its color-picking palette ▼ …



… then:

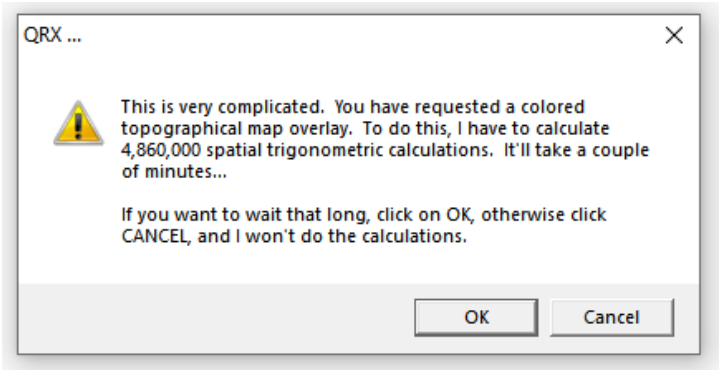
- Click one of the 'basic colors' you desire from the color grid; or
- Click one of the Custom colors beneath - a monochrome grey-scale selection, by default, but once you have chosen a specific color, you can click <**Add to Custom Colors**> to save it on the lower grid (as I have done here for a nice watery shade of blue), making it easier to select the same colors for other maps; or
- Click a color on the spectrum, perhaps adjusting the intensity by dragging the black arrow head slider on the right up or down; or
- Enter numeric values for the hue, saturation and luminance values, or for the red, green and blue color mix (or click a similar color then adjust the numbers); and
- Once you are happy with the color preview in the Color|Solid rectangle, click <**OK**> to use it and close the color picker – or <**Cancel**> if you decide to leave it as it was.

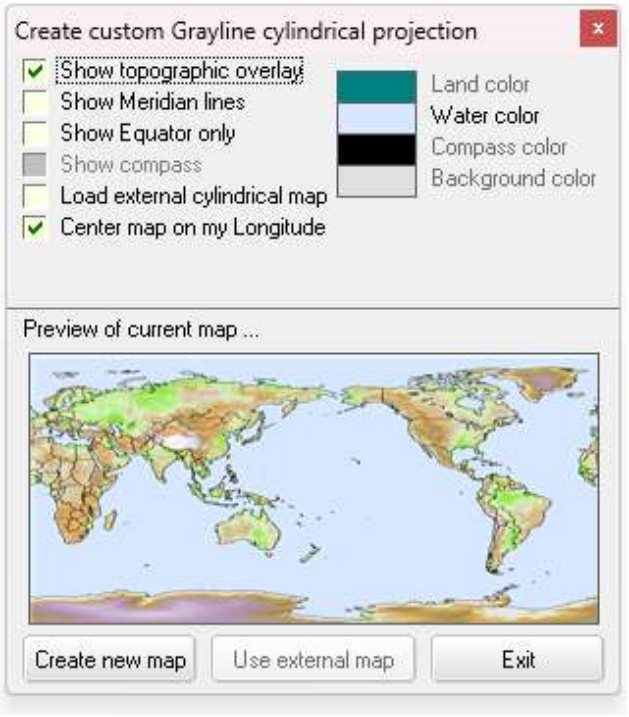


- Tick <**Load external cylindrical map**> if you prefer to replace Logger32's built-in cylindrical projection world map with your own map image (one you have already found and saved on disk). That tick magically opens a data entry box in which to type in the disk, folder and filename of your map, or you can click the <**Browse**> button that also appears to find and select your map file. Then click <Use external map> ▶

Copyright © 2026 Robert C. Furzer K4CY

- **Topographic overlay**: if the monotonous coloring of land on one of the built-in maps isn't to your liking, try the topographic coloring. Due to the CPU-intensive calculations required, generating a topographical overlay may take a few minutes on a slow PC. Luckily, you only need to do this once per tab, *unless* you change your mind on whether to show the topo overlay, or move home QTH and need to update the maps … so get each tab's map showing the correct station location first *before* adding the topo overlay ▶





> **Hinson tip**: the topographic colors make it a little harder to see all the spots on a 'busy' map such as the DX Spots and JTDX tabs, since some spots blend into the similarly-colored areas of background. The topo overlay is pretty cool on the less-busy View DX tab though.

When you have selected the colors and other options, click <**Create new map**> to generate the base map image used for that tab. You should see the effects on the Tracking window and on the preview image on the configuration form. If necessary, adjust and regenerate, and <Exit> when you're happy. Tackle the remaining tabs whenever you like.

> **Hinson tip**: there is no *need* to generate maps for tabs you don't use … but chances are, at some future point, you will click all the tabs and wonder why some of the maps look wrong. So, I suggest generating them all in one sitting. Get it out of the way. Make it so.

19.3 Six tabs for six maps

The Tracking window's six tabs[323] are used to select and show any one of six maps[324]. This section concerns each map's overlays showing additional information, mostly relating to the tab's name.

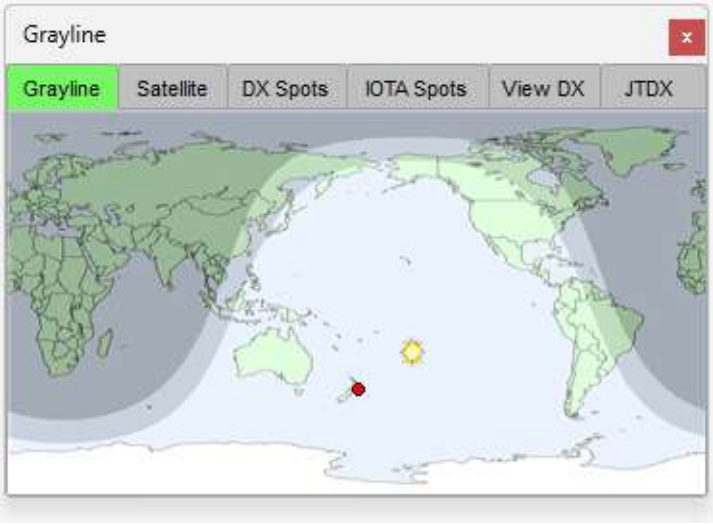
Right-clicking any map (more accurately: right-click a plain area of the map itself, not just its tab, nor a marker *e.g.* in the corners 'outside the globe' on the great circle maps) opens a configuration menu with overlay options relating to that map's purpose.

[323] If the tracking window isn't wide enough, you won't see all 6 tabs but can click the little right or left-facing arrows-in-white-boxes to shift the visible tabs left or right. Or enlarge the window to see them all.

[324] Given the extensive customization facilities, pedants like me might claim that there are *way* more than 6 maps … but once configured, there are just 6 readily accessible ones.

Copyright © 2026 Robert C. Furzer K4CY

19.3.1 Grayline tab

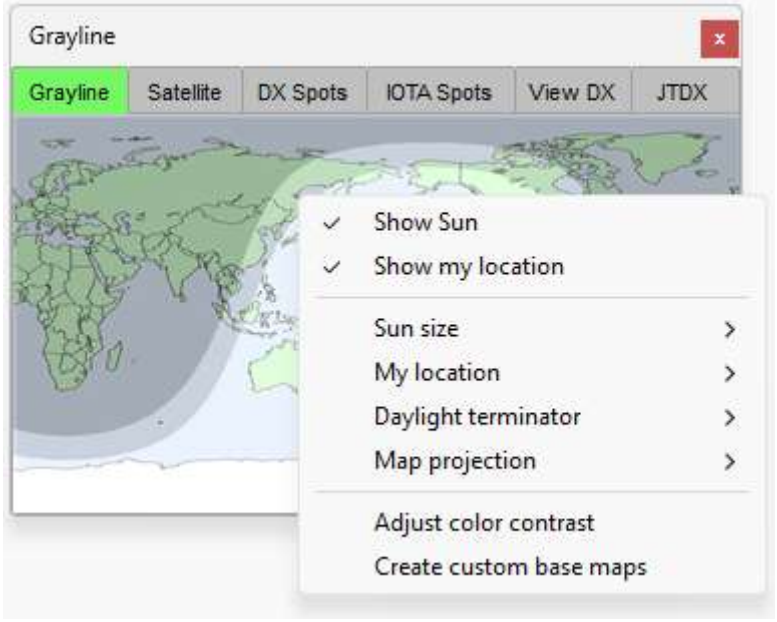


◀ The grayline tab overlays the world map with three zones of shading to distinguish areas of the world currently in daylight (daytime), in darkness (nighttime) or in a configurable transition zone between day and night (the dawn and dusk grayline, either side of the terminator).

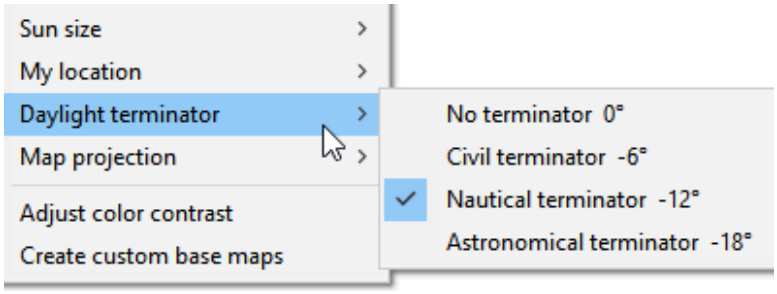
In the grayline area, the transitional ionospheric layers can refract radio signals around the globe rather than letting them escape to outer space or reflecting them back to Earth.

The sun's overhead position on Earth and the shaded areas are recalculated by Logger32 every few minutes, updating the map. Although this is not a continuous/real-time process, periodic map updates reduce the CPU load and are 'close enough for government work'.

Right-click the Grayline map for its configuration menu ▶



- **Show Sun**: indicates the point on Earth where the sun is directly overhead at solar noon.
- **Show my location**: puts a marker at your station location.
- **Sun size**: set the size of the sun *marker* displayed on the map. It does not literally affect the size of the sun: not even mighty Logger32 can do that.
- **My location**: set the color and size of your station location *marker*. Does not affect the size and position of your actual QTH. Does not even affect the position of the marker on the map.
- **Daylight terminator**: set the width of the indicated gray zone in degrees of longitude ▶



Copyright © 2026 Robert C. Furzer K4CY

Although the sun disappears from view when it dips below your visual horizon (it 'terminates', although not in the Arnold Schwarzenegger sense), night does not fall instantaneously as if someone just switched off all the lights in a coal mine. What we call 'dusk' starts somewhat earlier[325] as the sunlight is filtered and refracted through the murky atmosphere between us and the horizon (very murky if you live near a polluted city or burning forest). At sunset as the sun's disc slips below the horizon, the direct sunlight gradually fades away … but even after the disc is entirely out of sight, refraction and reflection from water vapor and murk in the atmosphere keeps the sky somewhat bright for a while longer.

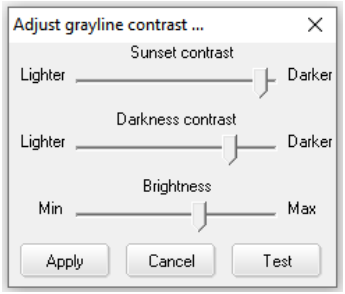
This is about the time that you notice the first few bright stars and planets such as Venus appear, provided there is not complete cloud cover, and provided local light sources (*e.g*. city lights) don't cause visual QRM. Far from land on a boat, mariners see loads more heavenly bodies than most of us. Actual astronomers, however, have to wait patiently on their mountain peaks or dark sky places ▶ for the moonless nights necessary to see the very weakest (*real* DX!) stars.



'Dawn' is similar, only different.

Hinson tip: it's up to you to decide how wide to make the dawn/dusk area on the map: I have noticed grayline propagation effects on 30m and 40m starting about 3 *hours* before my local sunset/sunrise, so the 'astronomical' setting might be appropriate for me on the mid bands. Topband DXers tell me the grayline effect may only last for a few minutes on 160m, so the 'civil' setting more closely reflects their observed radio effects.

- **Map projection** and **Create custom base maps** are described above.



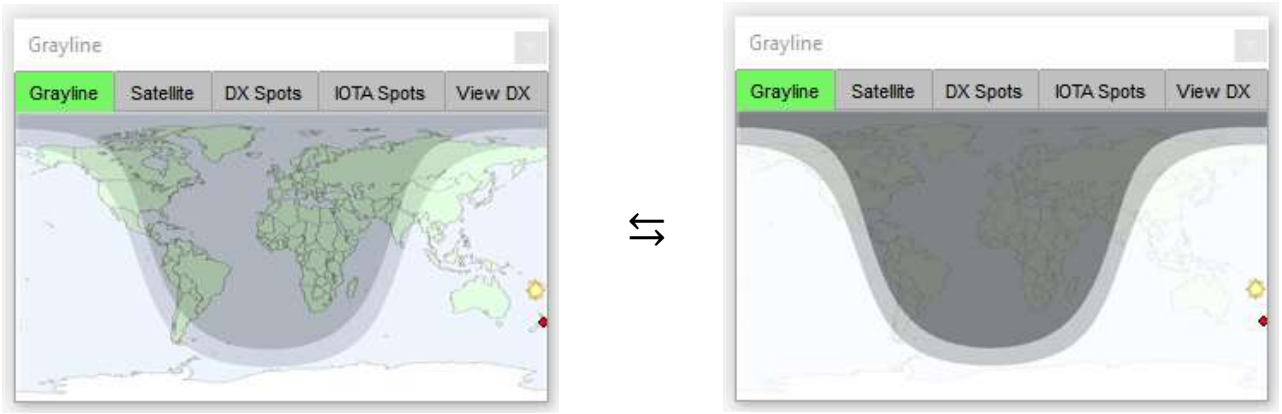
◀ **Adjust color contrast**: set the overall brightness level, and the contrast for the map's dusk/dawn and nighttime zones.

Adjust the color contrast sliders then click <**Test**> to see the effect on the map.

[325] The transition between day and night goes slower towards the polar regions (which may enjoy day or night for *months* at a time), and faster in the equatorial tropics.

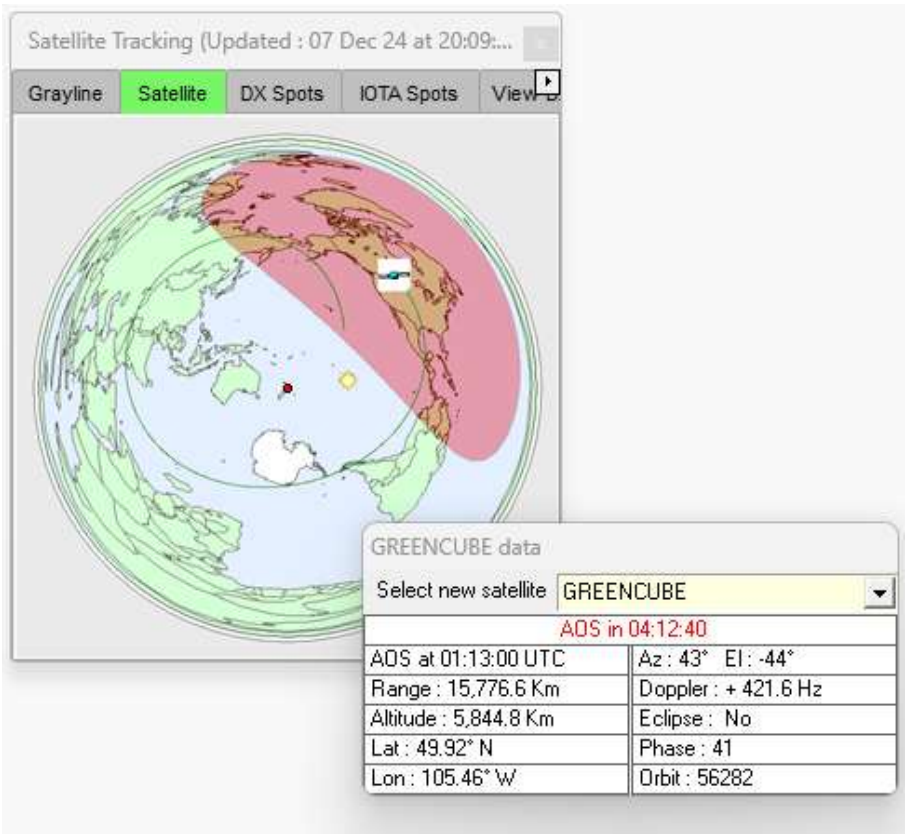
Copyright © 2026 Robert C. Furzer K4CY

▼ On the left, I set all sliders to 20, increasing to 80 on the right ▼



Adjust until happy, then click <**Apply**>.

19.3.2 Satellite tab



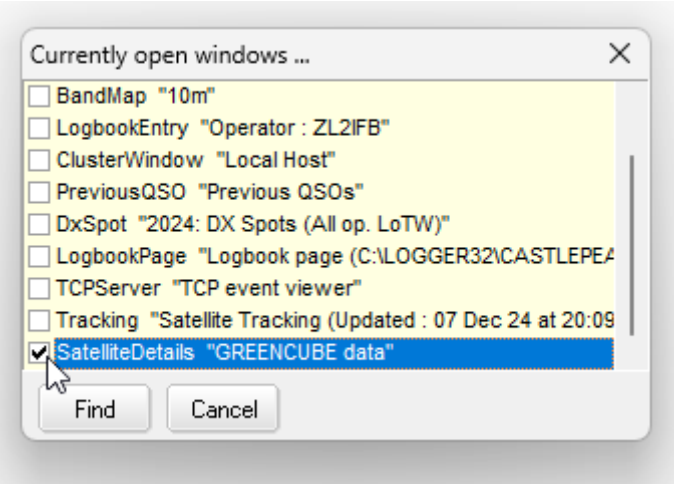
The <**Satellite**> tab ▲ overlays the world map with:

- The selected satellite's current directly-overhead position on Earth, with a little icon in a white square to represent the satellite (it is just an icon, not a photo, not to scale!);
- Its predicted track to next **A**cquisition **o**f **S**ignal or **L**oss **o**f **S**ignal at your station location; plus
- Its 'footprint' – the ground area from which the satellite is above the horizon.

Choose which satellite to track and display from the drop-down list in the table that normally appears when the <**Satellite**> tab is opened.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: the table is a separate window that can be moved around your screens independently of the map by clicking-and-dragging it. It may not be adjacent to the map as shown above, but it is convenient to do so. If it is not visible anywhere, right-click the satellite map and tick <**Show satellite details**>. If it is *still* stubbornly hiding, use ▶ **View ⇨ Find lost windows** to move the window to the top left corner of the main display where you can see it.



Aside from the satellite's name and ID in the caption, the satellite table shows:

- The hours, minutes and seconds remaining until the next **AOS** or **LOS**. When the satellite reaches AOS, a new track is drawn on the map from AOS to LOS and the time indicator on the top row changes to show the time remaining to LOS. Once LOS is reached, a new track will be drawn to the *next* AOS, and so on.
- The **Range** (distance to the satellite), **Az**imuth (bearing) and **El**evation (angle above the horizon) of the satellite, calculated for your station's latitude and longitude but at sea level[326].
- The satellite's current position, specified as the **Lat**itude and **Long**itude on Earth where the satellite is vertically overhead, and its **Altitude** above sea level at that place.
- The **Doppler** shift, calculated as the satellite approaches you, swings past and then recedes at a significant speed. If the satellite has a CW telemetry beacon, you can expect to receive it on its nominal frequency *plus* the Doppler shift as it approaches you. You will probably notice the tone falling markedly as the satellite passes by and recedes, just like those old-fashioned two-tone police sirens.
- The **Eclipse** status: is the satellite currently in full sun or eclipsed *i.e.* in the Earth's shadow? This affects the amount of energy being captured by its solar panels. If its batteries are failing or dead, low voltage may prevent the transponders and beacons transmitting reliably.
- The **Phase** and **Orbit** numbers tell you how many times the satellite has circled the globe.

The indicated satellite position, track and footprint on the map, and the information on the form, is reasonably accurate but only *if*:

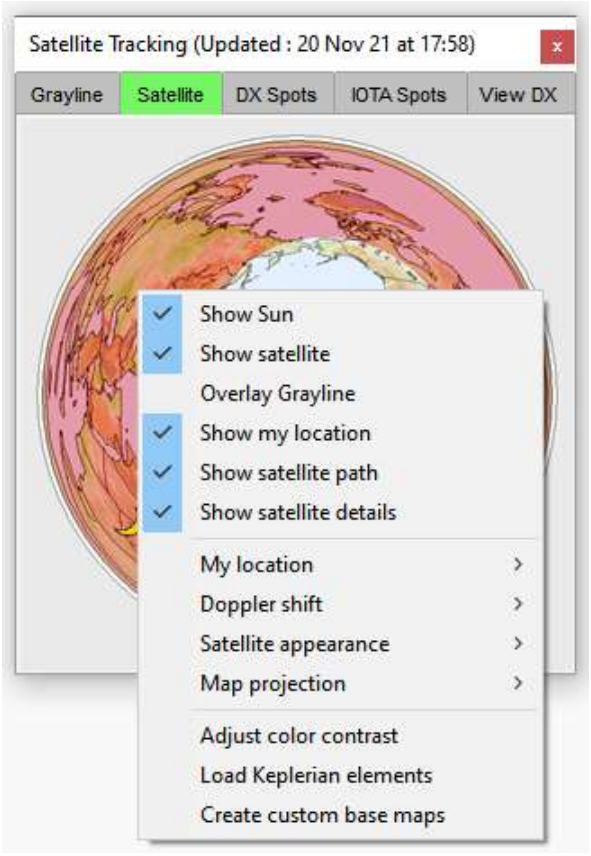
- Your station's latitude and longitude are correctly defined.
- The satellite's Keplerian elements are up to date, taking account of any orbital adjustments and the variable atmospheric drag, solar radiation *etc*. Keps that are months or years old are likely to be inaccurate, particularly for satellites with low orbits, surfing the turbulent outer reaches of the Earth's atmosphere.
- Your PC clock has the correct date and time.

[326] If you are lucky enough to be perched on a hilltop or mountain peak, satellites *may* be visible at low elevations a little before the calculated sea-level AOS and after the calculated sea-level LOS, depending on the directions and local topography. The effective footprint may also be extended by sporadic-E or F-layer propagation, knife-edge refraction and other esoteric anomalies beloved of VHF/UHF/microwave DXers.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: the markers, lines and zones on the map depend on how the map is configured using its right-click menu, while the list of satellites available in the <**Select new satellite**> drop-down on the table comes from the most recently downloaded Keplerian elements. Read on for details on both.

As usual, right-click the satellite map to open its configuration menu ▶



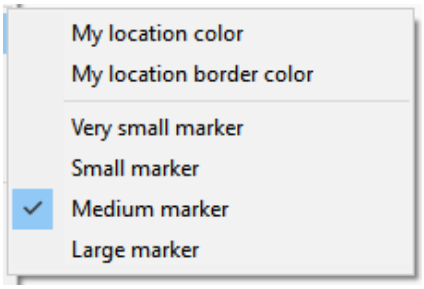
As with almost everything in Logger32, you can customize the satellite map according to your preferences:

- **Show Sun:** puts a marker on the map where the sun is directly overhead.
- **Show satellite:** it's hard to think of a reason that you might want to display the satellite tab *without* showing a satellite but, hey, if that's what you want to do, go right ahead.
- **Overlay Grayline**: shows the day, night and gray zones. This is a rough guide to whether the satellite is likely to be eclipsed, but remember that it is a long way up, so *its* visual horizon is a lot further away than ours.
- **Show my location**: puts a marker where you are – or rather, where you have told Logger32 that your station is located. If the marker is patently in the wrong place, guess what: you have most likely failed to configure your latitude and longitude correctly. Errors significant enough for you to notice are definitely worth correcting.
- **Show satellite path**: shows where on Earth the satellite has flown, or is going to fly, directly overhead.

Hinson tip: the satellite position and footprint are automatically recalculated and redrawn on the map and the data table every 5 seconds. If you simply cannot wait that long, click the map to update it *immediately*.

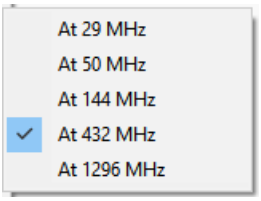
- **Show satellite details**: the data table can be shown or not – your choice. You might perhaps show just the map normally, keeping an eye on a satellite's approach, then show the data table when it gets close to AOS and for the duration of the pass, hiding it again at LOS. Or not.
- **My location**: opens a submenu to configure the appearance of your home QTH marker on the map ▶

  Its position on Earth is determined by your station latitude and longitude.



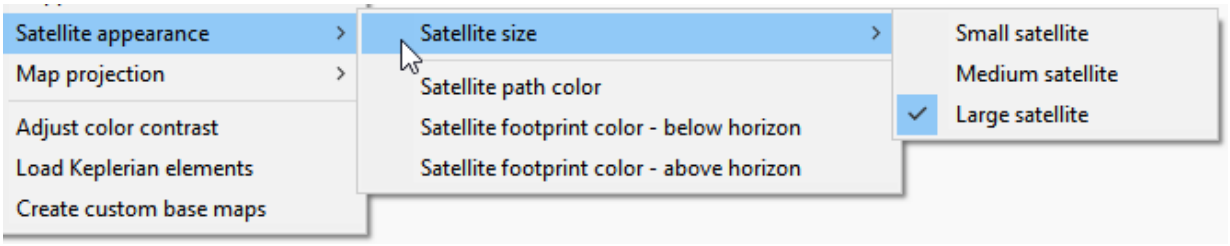
Copyright © 2026 Robert C. Furzer K4CY

- **Doppler shift**: specify the frequency at which the Doppler shift is calculated for the data table ▶

  Doppler shift is proportional to the transmission frequency, so the effect is more marked at higher frequencies. It also reflects the speed of the satellite relative to you, changing more rapidly as it passes overhead than when it is near either horizon.

- **Satellite appearance**: ▼

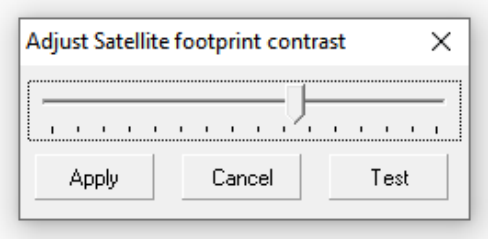
  

  - **Satellite size**: display a satellite icon in one of three sizes. In case you wondered, this neither affects or represents the size of the actual satellite!
  - **Satellite path color**: show the satellite's orbital path in any color you like.
  - **Satellite footprint color**: configure the color of the footprint area on the map. The footprint can magically change color when the satellite pops in view *i.e.* it comes above the horizon at your QTH (*if* you were at sea level and *if* there were no obstructions in its direction), changing back as it falls below the horizon, slipping quietly away out of sight for another trip around the world.

    Hinson tip: you might like to use the footprint color change as a visual reminder in the shack to break away from your web browsing, soldering, HF DXing or whatever in order to catch each pass.

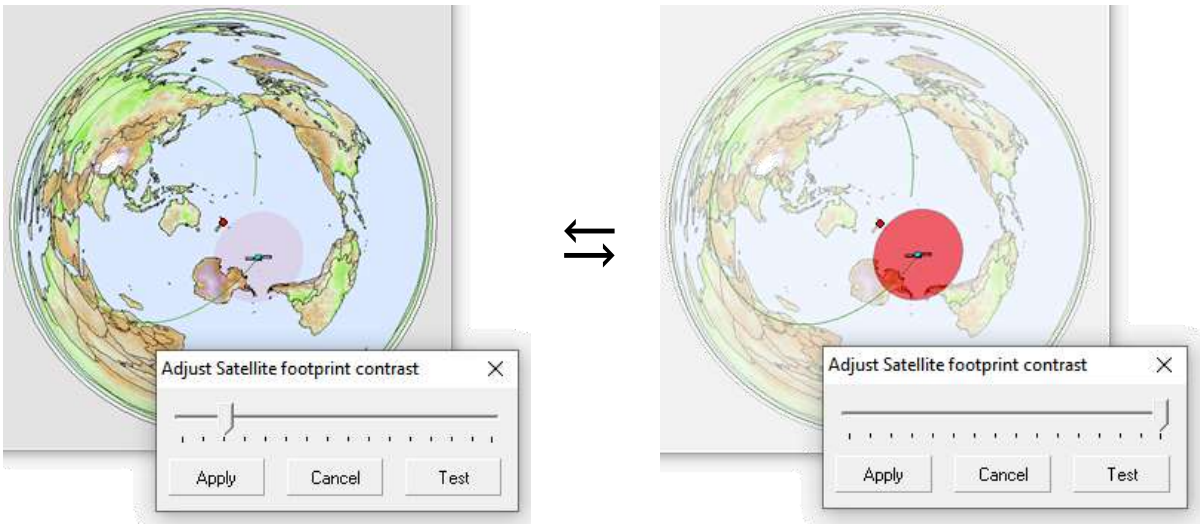
- **Map projection** and **Create custom base maps:** see above.
- **Adjust color contrast**: differing colors and intensities between the footprint area and underlying map enable you to distinguish these features – or not if the contrast is insufficient.

Click and drag the slider right or left to increase or decrease the contrast, making the footprint area more or less distinct ▶



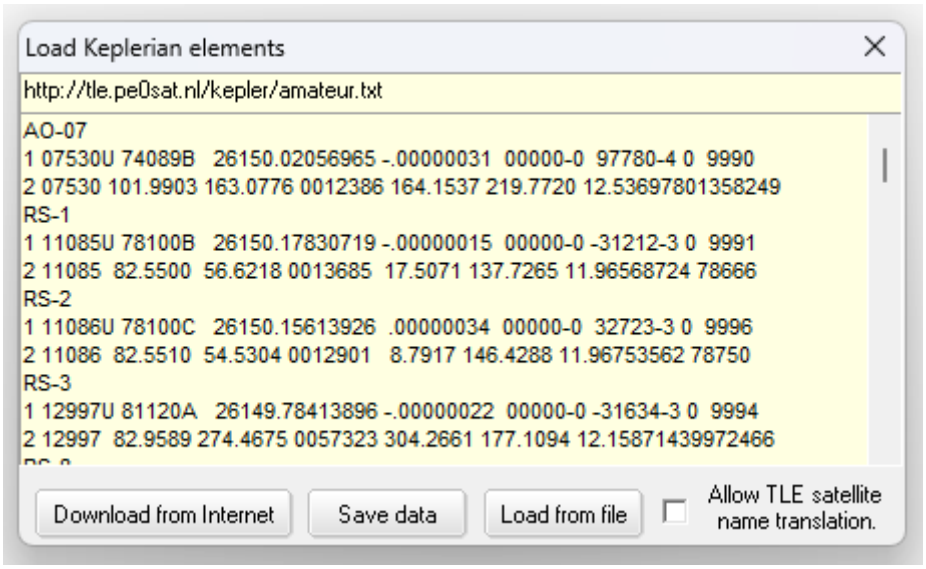
Copyright © 2026 Robert C. Furzer K4CY

Click <**Test**> to see the effect ▼



When you are happy with the contrast, click <**Apply**> to save and apply that setting, closing the little configuration form. The image contrast changes when the map is next updated.

- **Load Keplerian elements**: ▼



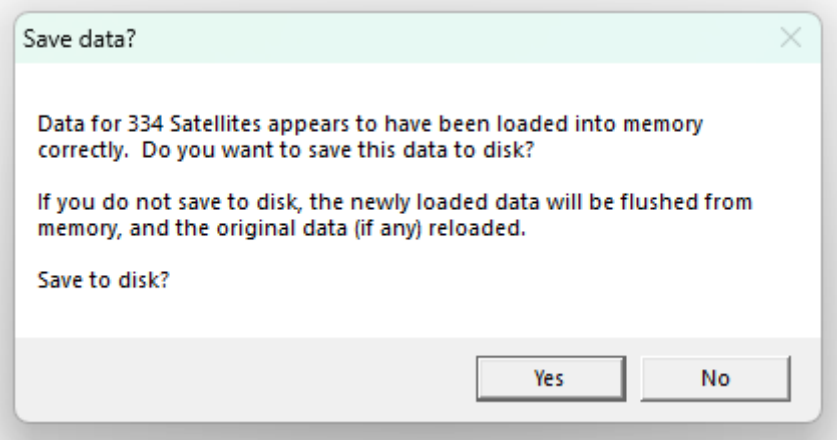
Note: the Celestrak URL (http://www.celestrak.com/NORAD/elements/amateur.txt) no longer works with Logger32 due to a change of format. Instead, replace the URL at the top of this pane with one in the correct Two Line Elements DOS or Unix format, such as:

http://tle.pe0sat.nl/kepler/amateur.txt

Then click the <**Download from Internet**> button to download and check the TLE data file from the URL. After a short delay, the data appears in the body of the form and a "Save data?" pane appears. Behind the screen, Logger32 has integrity-checked the downloaded TLE data and warns you if it is unsuitable.

Copyright © 2026 Robert C. Furzer K4CY

You now have the opportunity to scroll through the listed satellites to see if the TLE for your favorite satellites are actually downloaded: PE0SAT lists 300+ satellites. AMSAT lists fewer than 100. So, PE0SAT probably has your favorite satellites listed, while AMSAT may not. If you're happy, save the data to disk by clicking <**Save data**> to have Logger32 offer to save the TLE to the file *C:\Logger32\KEPS.TXT* ▶



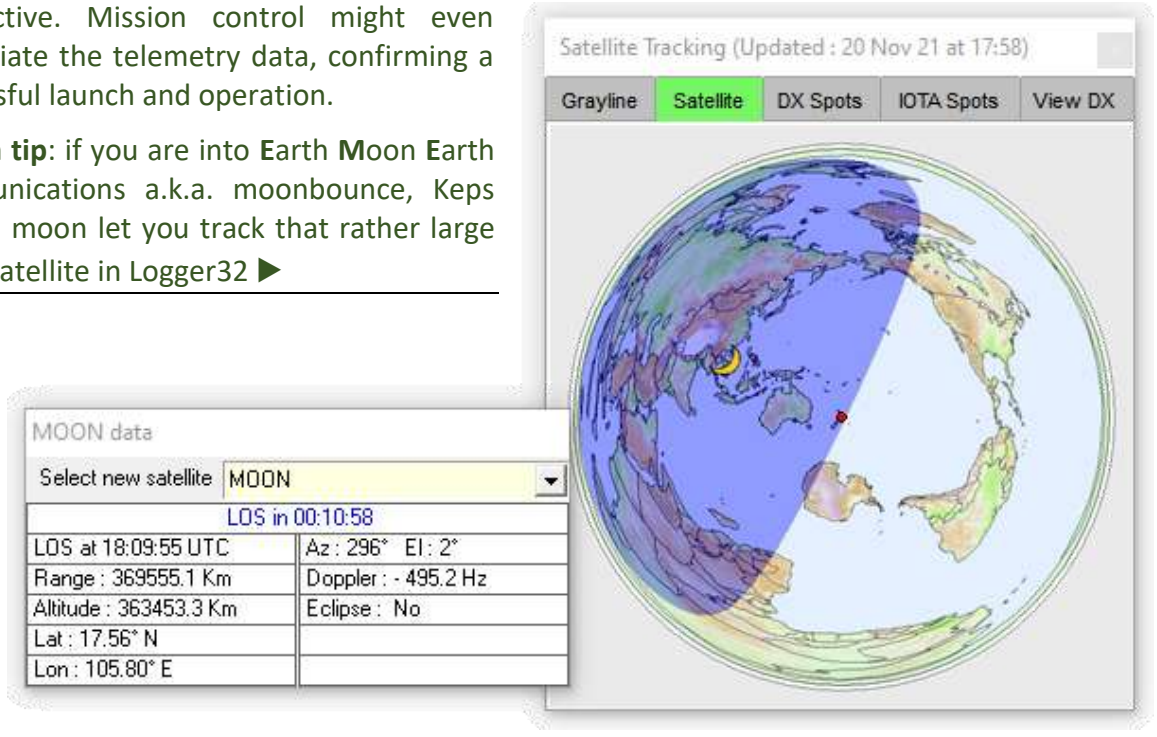
Click <**Yes**> to save the new TLE data, or <**No**> to discard it.

Alternatively, click <**Load from file**> if you have already downloaded (or calculated, from your own observations!) a TLE data file that you wish to use. That button opens File **E**xplorer for you to locate and select the TLE data file on your PC.

<**Allow TLE satellite name translation**> lets us rename satellites from whatever is used in the TLE data to whatever we wish *e.g*. while the TLE may list the International Space Station as "ISS", LoTW expects "ARISS". The file *C:\Logger32\FauxTLE.TXT* is provided as a starting point for name translation. Each line in *FauxTLE.TXT* contains a TLE satellite name and Logger32's translated name, separate by a comma. Add as many lines as you like in a text editor. Spaces either side of the comma are ignored but make the file more ham-readable.

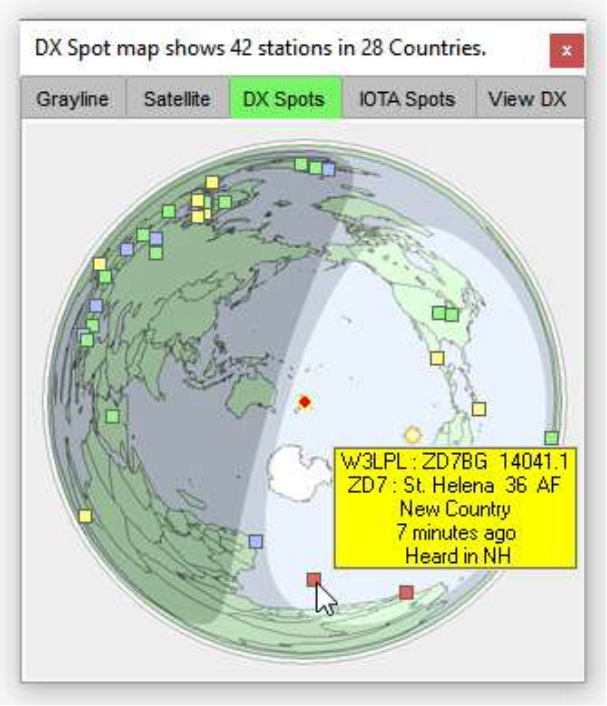
Hinson tip: to maintain accuracy, update your Keps at least once a month if you are actively tracking and using satellites. Also update soon after a new amateur satellite is launched: you should be able to copy its telemetry beacon/s when in range, even if its transponders are not yet active. Mission control might even appreciate the telemetry data, confirming a successful launch and operation.

Hinson tip: if you are into **E**arth **M**oon **E**arth communications a.k.a. moonbounce, Keps for the moon let you track that rather large Earth satellite in Logger32 ▶



Copyright © 2026 Robert C. Furzer K4CY

19.3.3 DX Spots tab



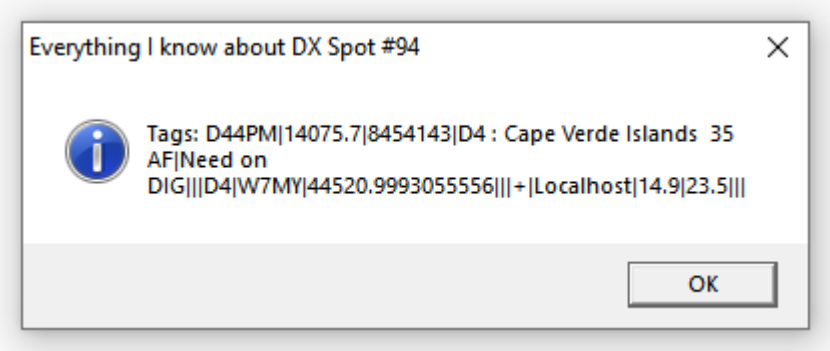
◀ As they arrive from DX cluster and are filtered and processed through the DX Spots pane, DX spots are plotted with square markers on a rectangular or circular world map.

The markers are customizable and can use the same highlighting colors for 'new ones' as on the BandMaps.

The round marker (always dead center of the great circle map) shows your station[327]. You can customize the marker's color and size.

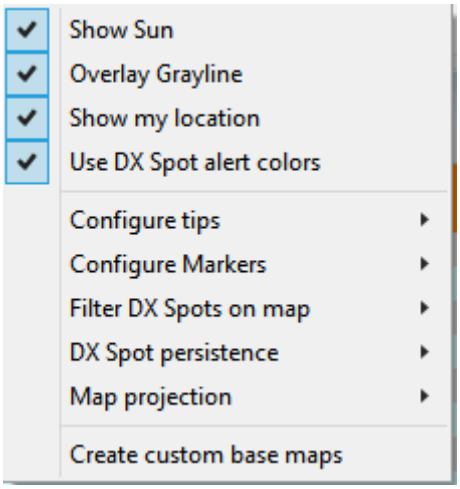
Clicking a map marker is just like clicking a spot in the DX Spots pane or BandMaps *i.e.* the radio QSYs and the log entry pane is pre-filled with the DX station's callsign and other information.

Right-clicking a marker reveals all the information Logger32 holds concerning that DX spot ▶



Mouseover a marker for a yellow tooltip (as seen above) with pertinent information about that station.

Right-clicking relatively blank areas on the map (avoiding the icons) lets us configure and customize it ▶



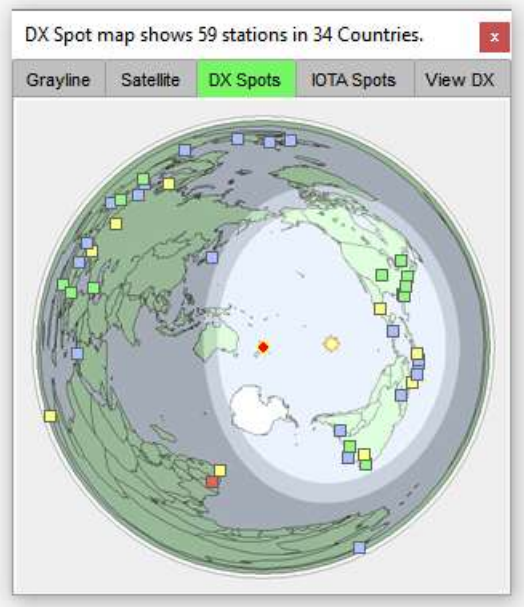
- **Show Sun**: puts a marker at the position on Earth where the sun is directly overhead, blasting its rays at the ground.
- **Overlay Grayline**: shows the area in transition between daylight and darkness. See the grayline overlay section for more on this.

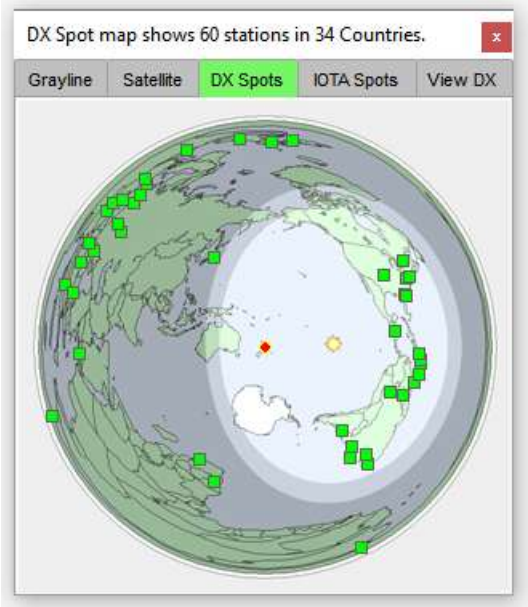
[327] If the red dot is *not* at your station location (normally, where you live), you either need to create a custom base map for this and perhaps other tabs, or correct your station location.

Copyright © 2026 Robert C. Furzer K4CY

- **Show my location**: puts a colored circle on your home, like a marksman's laser dot.
- **Use DX spot alert colors** means show the DX spots in the same colors as the worked/confirmed table and BandMaps, indicating which if any are 'wanted' (*e.g.* new ones all time or just this year, new on this band, new on this mode) ▼

If this option is *not* selected, the **Marker fixed color** defined under <**Configure Markers**> is used for *all* plotted DX spots (not your home QTH or the sun) ▼

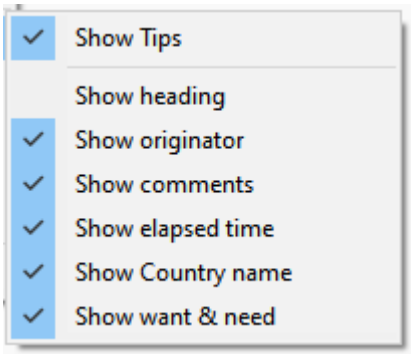


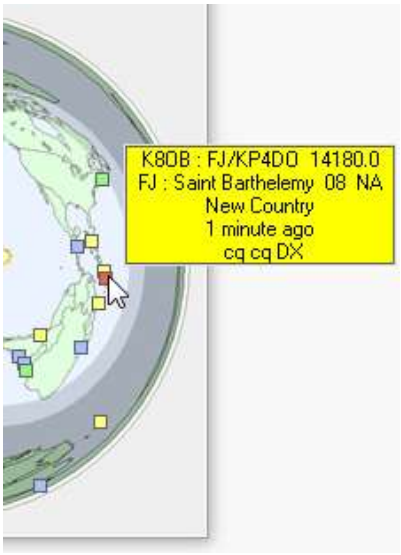


Spots for known DXpedition callsigns have thickened borders relative to ordinary DX spots.

Hinson tip: over the years, having grown accustomed to the particular colors I have chosen, I find them invaluable to pick out the few DX stations most worth chasing out of all those recently spotted. However, humdrum spots for ordinary stations are still of interest as they indicate general areas of activity. Notice on the maps above, for instance, that there is evidently a lot going on in Europe and the Americas, but relatively little in Asia, Africa and the Pacific … although the pair of red and yellow spots in Africa on the left map catch my beady DXer's eye.

- **Configure tips**: choose what information is shown, if any, when you mouseover any of the spots on the map ▶



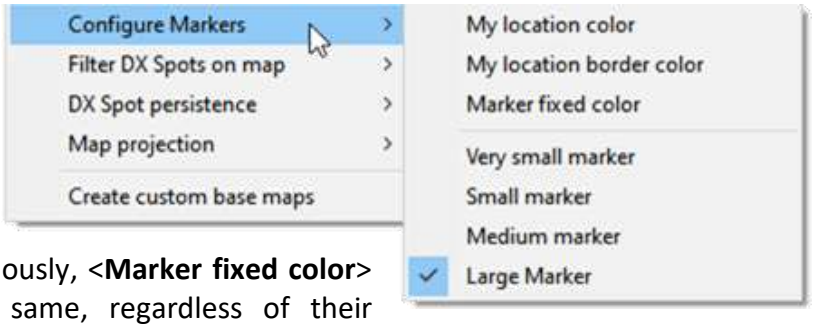


◀ Here I have pointed the mouse at one of the red spots to see:

- Who spotted it (the originator, the spotter), the spotted DX station's callsign, and the frequency on which he was spotted.
- The DX station's prefix, entity, CQ zone and continent.
- 'New Country' (*i.e.* I hadn't worked FJ yet this year) which explains why the spot was red.
- When he was spotted (is the spot fresh or stale?).
- And the spot comments (is he working split?).

Copyright © 2026 Robert C. Furzer K4CY

- **Configure Markers ▶**

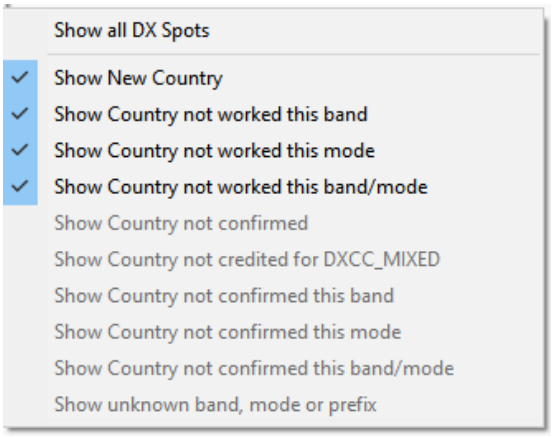


  Here we set the color filling the round marker indicating our station location, and color its border (the line around its circumference).

  As mentioned and shown previously, <**Marker fixed color**> colors all the DX spots the same, regardless of their wanted/needed status, if we *don't* <**Use DX spot alert colors**>.

  Finally, we choose a size for the DX spot squares.

- **Filter DX Spots on map** lets us either selectively filter and only display spots that satisfy the criteria defined on the sub-menu, or **Show all DX Spots ▶**



  To display all spots *except* those for which we have the DXCC entity fully confirmed on the spotted band and mode, tick *all* the selections *other* than **Show all DX Spots**.

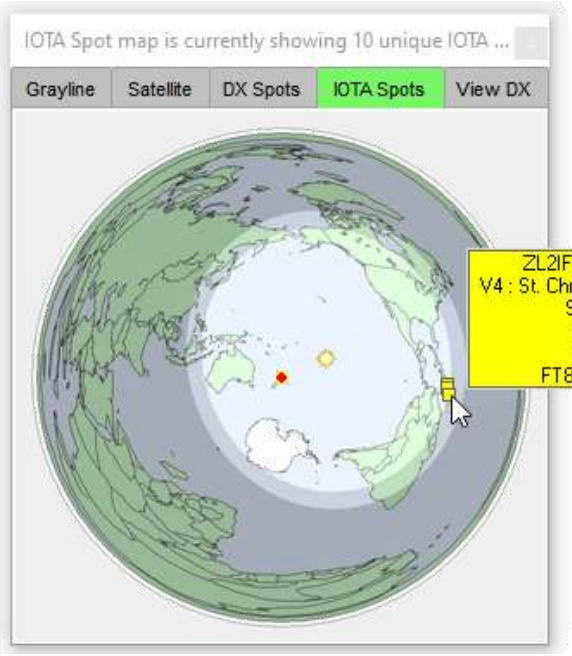
- **DX Spot persistence:** how long should DX spots remain visible on the map? Select a time from 5 minutes to infinity (meaning you are happy to see stale spots for stations that are unlikely to still be QRV).

> **Hinson tip**: if I **Show all DX Spots**, there are far too many for my liking on HF, even if I reduce the spot persistence as far as it goes. It's even worse if I am being *hosed* with FT# spots from the cluster. Things are different, though, if you are monitoring (say) 6m or 2m, where there are few if any spots normally: the appearance of a *rash* of spots indicates a band opening, a 'lift' of some sort.

- **Map projection** and **Create custom base maps** work the same as the other tabs.

Copyright © 2026 Robert C. Furzer K4CY

19.3.4 IOTA Spots tab

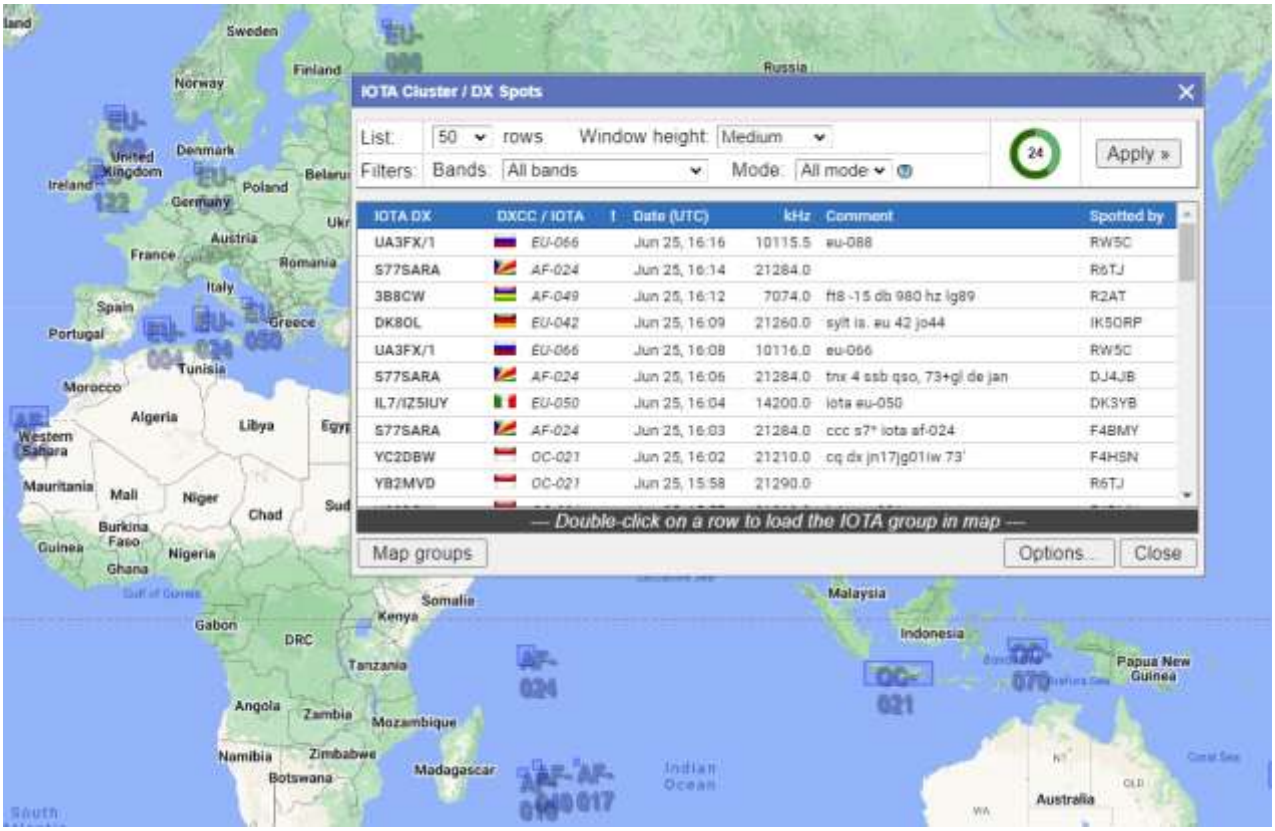


◀ The IOTA Spots map displays markers for DX stations that are (or at least claim to be) operating from islands within the IOTA award scheme.

The IOTA status is determined either from the DX spot comments, or from the received callsign if the prefix unambiguously indicates a specific island with a unique IOTA reference (*e.g.* the V47JA spot in this example). Logger32 is limited by the quality of the information available *e.g.* IOTA stations are sometimes spotted with the wrong IOTA references, while DX spots with the wrong prefixes sometimes take on a life of their own *e.g.* mis-copied Swiss HB4 CW stations repeatedly spotted as Cypriot 5B4s. Logger32 corrects common formatting mistakes and displays a ? in the yellow tooltip to draw your attention to the fact that the data (even corrected) may not be accurate.

Hinson tip: as always, it is up to you as the station operator to make *sure* that you log the stations you work accurately. In some jurisdictions, that *may* even be a legal obligation under your license.

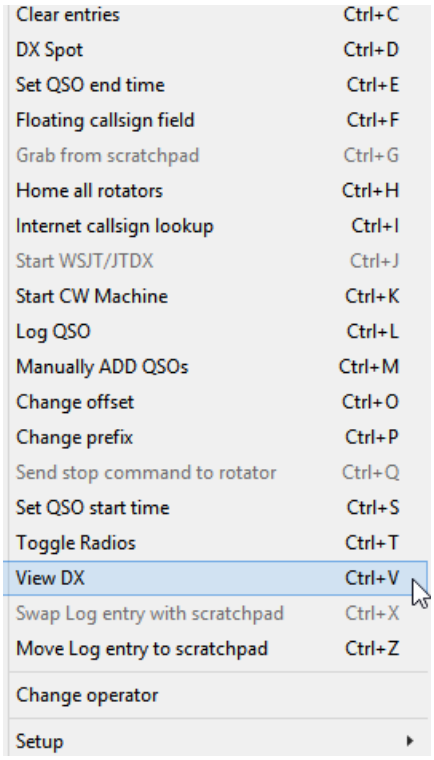
By the way, we can map current IOTA activations and display a table of recent IOTA DX cluster spots through the IOTA website: go to www.iota-world.org/iotamaps/ for this ▼

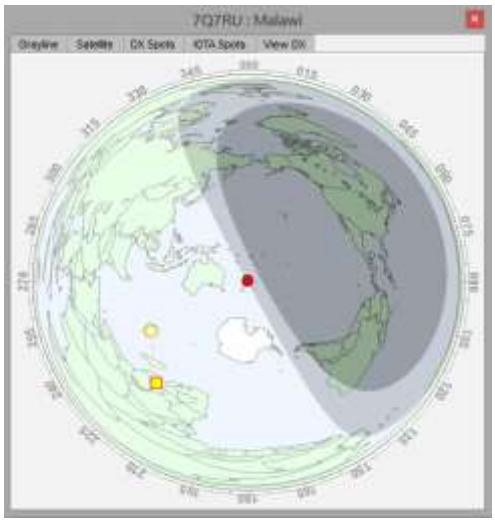


Copyright © 2026 Robert C. Furzer K4CY

19.3.5 View DX tab

Whereas the DX *spot* map normally shows a load of DX spots from DX cluster, the *DX* map plots the likely location of the single **station you are currently working** and logging in the log entry pane, according to the station's DXCC entity (determined primarily from his/her prefix) or the center of a specified IOTA Island Group ▶

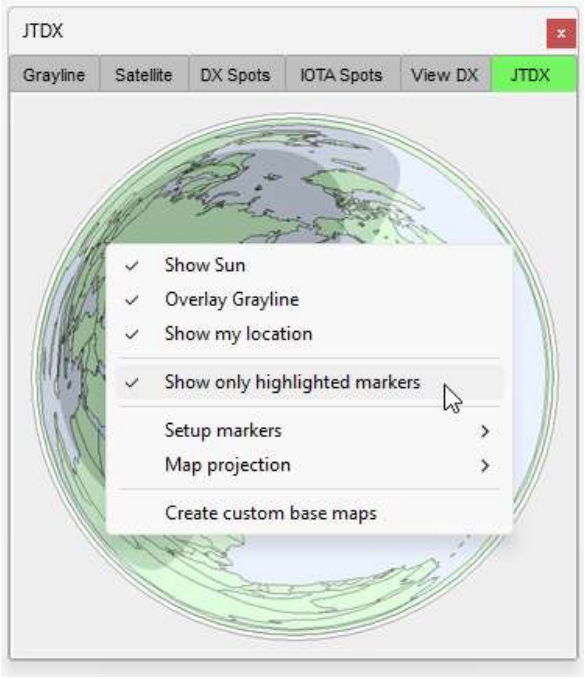




◀ Pressing <**Ctrl**+**V**> with the focus on the log entry pane opens the <**View DX**> tab and map in the Tracking window.

19.3.6 JTDX tab

Digimode stations decoded by JTDX on your system and passed through to Logger32 can be plotted on the JTDX tab ▶



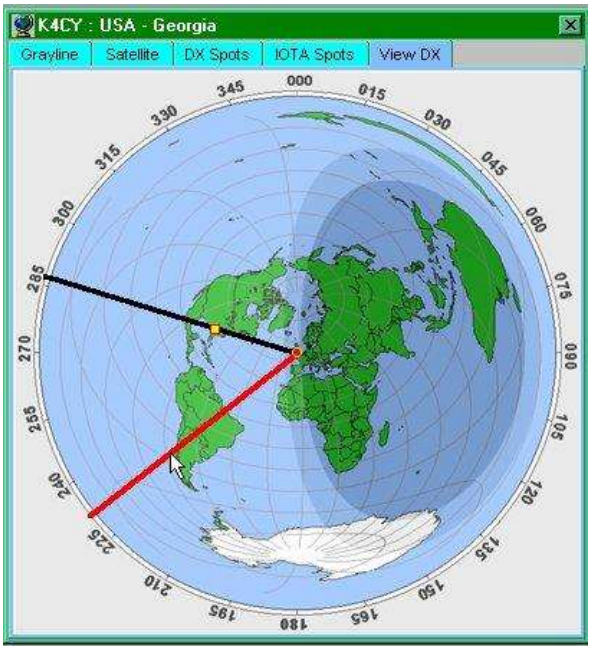
Mysterious digimoders with curious callsigns, unrecognized prefixes and hence unknown locations are plotted by Logger32 at your station QTH. Virtually all of them are caused by mistaken decoding – bit errors in the data streams received over noisy radio channels that pass the limited error-checking/integrity controls typical of digital protocols such as FT#.

Right-clicking the map opens the usual map configuration form available on other tabs but with an extra option: rather than plot *all* decoded stations, you can cut the clutter by plotting only highlighted spots (your 'new ones'). See the UDP BandMap section for more.

Copyright © 2026 Robert C. Furzer K4CY

19.3.7 Rotator overlay

Great circle maps in the Tracking window (*except* for the Satellite tab[328]) can display and control headings for computer-controlled rotators ▶



Overlaid colored lines can indicate:

- The short path directions to DX stations.
- The commanded antenna direction (simply click the map to beam that way).
- Its *current* direction.
- Its beamwidth.

See the Antenna rotators chapter for details.

The compass rose and indicated directions are bearings relative to **true** North, ignoring the magnetic declination (compass) offset at your QTH.

19.4 Map FAQs

Q. *I* am the very epicenter of my universe! Why aren't the maps centered on me, me me *me*?

A. There are two key things here.

Firstly, Logger32 needs to know where in the world you are, so right-click the log entry pane, then click **Setup ⇨ My QTH Lat/Long** and tell it, or at least confirm its knowledge.

Secondly, Logger32 needs to **generate** maps centered on your QTH for *each* of the six tracking window tabs (the one/s you use, at least). See above for instructions. There is no option to 'center map on my location' for the great circle maps … because they are *automatically* centered on the currently-defined station QTH. For Miller-style rectangular maps, however, click to tick <**Center map on my Longitude**> before generating them.

> **Hinson tip**: if you are setting up Logger32 for the first time, working systematically through this User Manual perhaps (you brave soul you!), or you are updating the maps having recently moved station QTH, I recommend taking a moment to *generate maps for all six tabs*, rather than just doing one or two. That way, when you come to use the Tracking window at some future point and happen to open tabs that you don't normally use, you won't be quite so perplexed to discover apparent problems with the maps.
>
> **Hinson tip**: sometimes it is handy to see the world from the perspective of a DX station, for instance to figure out who they might be beaming towards on the short path for you to work them on the long path at the same time. Although Logger32 *can* generate great circle maps

[328] Dedicated satellite tracking applications can control an Az-El rotator to follow the calculated trajectory of a satellite as it rises above the horizon at AOS, reaches its zenith, then descends below the horizon at LOS. Logger32 is not so dedicated. It's more of a casual take-it-or-leave-it satellite tracking app.

Copyright © 2026 Robert C. Furzer K4CY

centered on any location currently defined as the station QTH, it is fiddly to change station QTH to the DX location, redraw the map, save a screenshot, then revert to your station QTH and redraw the map again. Mapping apps and functions are better for this kind of thing – check out Club Log's great circle mapper, for example, or GcmWin from SM3GSJ, or DX Atlas from VE3NEA. Google Earth is great for satellite images, though not great circle maps.

Q. How do I connect my S.A.T. to Logger32 to log satellite QSOs?

A. The CSN Technologies **S**elf-contained **A**ntenna **T**racker is capable of calculating the positions and turning a connected rotator to track passing satellites (or the moon or sun), interfacing to ICOM radios via CAT or to SDR programs via Hamlib, and recording up to 100 QSOs in its internal log.

It can be connected wirelessly to your shack WiFi, allowing you to download the stored log as an ADIF file to the shack PC, which you can then import into your Logger32 shack log in the normal way … and provided that goes well, you probably want to clear the S.A.T.'s internal log to leave space for another 100 satellite QSOs.

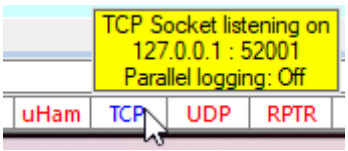
Hinson tip: if you're as paranoid as me, keep the downloaded ADIF as a justincase backup.

It can also connect to Logger32 via the WiFi using Logger32's TCP server, transferring QSO records individually in ADIF format as QSOs are logged on the S.A.T. To do this:

1. Within the S.A.T. dashboard, select the 'Location' tab.
2. At the bottom of the 'Location' page, set the 'QSO LOG TYPE' to ACLog from the drop-down list.
3. Set 'QSO LOG IP' to the IP address of your shack PC running Logger32. To find it, run IPCONFIG from a Windows command prompt (a.k.a. terminal) or:
   - Press [**Win+X**]
   - Click to open **Network connections** (or use the [**Alt+W**] shortcut)
   - Click the properties area top centre-ish ▼



   - Look through the properties for the **IPv4 address** to enter into the S.A.T. *e.g*. 192.168.1.1
4. Set the 'PORT' to Logger32's TCP server port number – by default 52001. Mouseover the TCP panel on the status bar at the bottom of Logger32's screen and read the pop-up ▶ message: if there is no pop-up, you'll need to open the TCP port first.



5. Leave 'DX COMMANDER PORT' set at 0 = disabled.
6. Save the S.A.T. settings and check it out. You can keep an eye on the TCP traffic through Logger32's TCP event viewer.

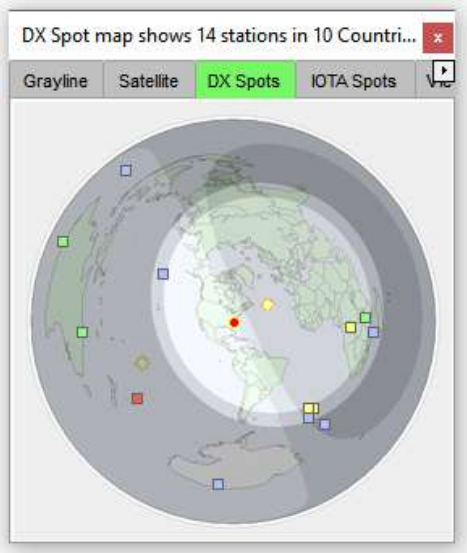
[Thanks to Keith G0FEA and the S.A.T. website for the info.]

Copyright © 2026 Robert C. Furzer K4CY

Q. Why are DX spots positioned incorrectly on the map?

A. There are two common reasons:

1. You haven't customized (updated, redrawn) the base maps for your current station location. The geographical locations of spots, satellites *etc*. are determined relative to your defined station QTH and then displayed appropriately on the overlays applicable to each map tab … but if the maps on the base layer of the images are wrongly positioned, the markers *appear* to be in the wrong places in the world.



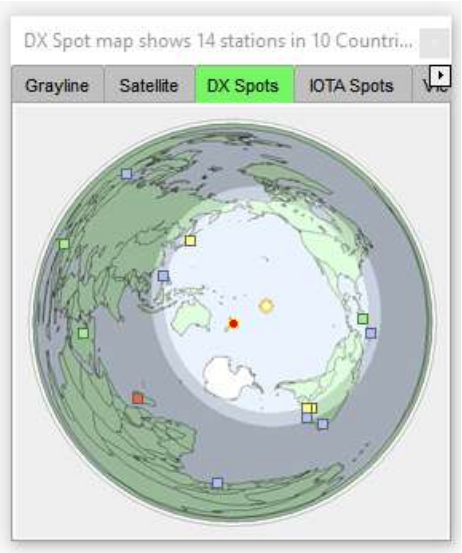
If you neglect to update any base map after installing Logger32 or moving station QTH, you will *hopefully* notice that the markers are not shown in the correct places. It's even more obvious if you elect to show distinctive markers for the sun and your home QTH.

◀ This is how the DX Spots map appeared soon after I installed Logger32 on a new PC.

The red dot in the center of the map shows my station QTH … which is in New Zealand, not North America as it appears. I took this screenshot mid-morning when the sun was to my North East, overhead somewhere in the central Pacific Ocean, not the central North Atlantic. Notice that the darkest gray shading shows that it was night for Europe … so clearly the sun was not where the marker indicates as it tends to shine only in the daytime, for some reason. Strange that.

I moused-over the blue DX spot marker near the bottom of the map to discover it was a 9G5 station in Ghana, West Africa, not one of the frozen Antarctic bases suggested by the position of the marker.

Having recalculated the base map, everything slotted neatly into place[329] ▶



Notice that the markers remain at the same positions in the image, whereas the world map changed underneath.

[329] Did you notice that the dark shaded night area changed from a kidney shape to a doughnut? That's simply a consequence of the few minutes that elapsed between screenshots, not the recalculation of the base map. Remember: the day/night shading is an *overlay*, a separate layer on top of the base map image.

Copyright © 2026 Robert C. Furzer K4CY

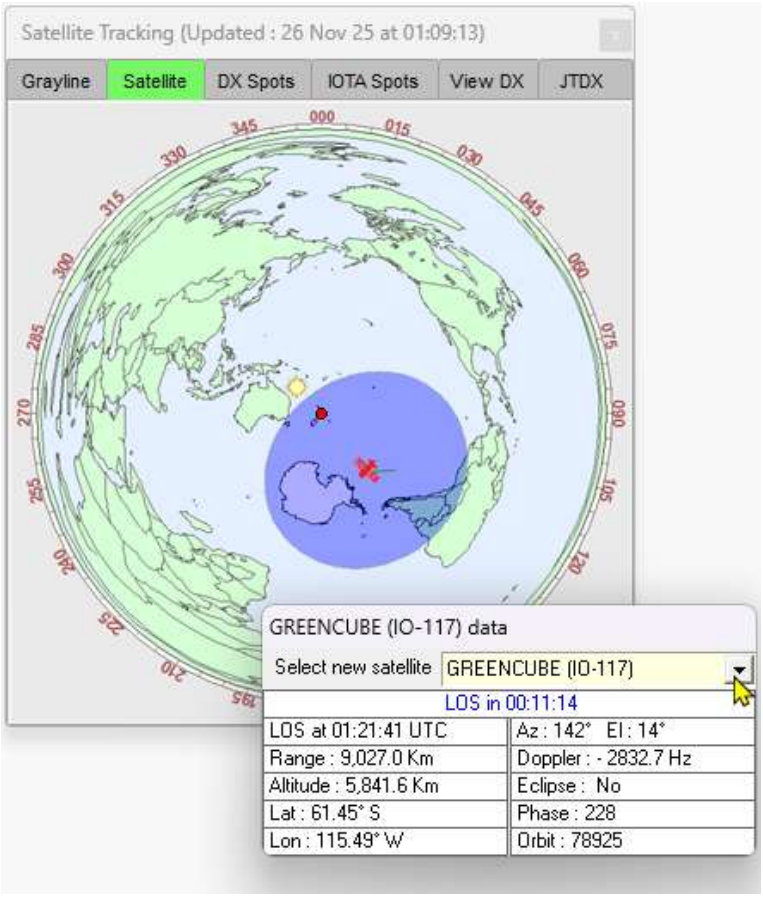
2. Locations gleaned from DX spots *etc*. are imprecise and sometimes wildly inaccurate (wrong, wrong, wrong!) due to:
   - Mis-sent or mis-copied callsigns *e.g*. YC4K and C4KY are dyslexic/garbled versions of K4CY, the prefixes indicating markedly different locations.
   - Misleading/ambiguous locational modifiers (*e.g*. /P5 or /LH) and prefixes (*e.g*. E5 *usually* indicates someone on the South Cook Islands but they *could* be on the remote North Cooks, a different DXCC entity).
   - Vagueness generally *e.g*. these days, a W5 may be almost anywhere in the US, not necessarily Texas, and GB special event stations are usually located in England but may be anywhere within the United Kingdom of Great Britain and Northern Ireland (comprising several DXCC entities).
   - Remote station ops using their home callsigns, giving no indication as to the actual location of the remote transmitter (which may be illegal, and is certainly naughty).
   - Operators' doubts about their true locations *e.g*. ham passengers on cruise or container ships, in aircraft or camper vans, drunks in charge of a microphone, bewildered Alzheimer's sufferers …
   - Mistakes with Maidenhead locators *e.g*. transposed letters or digits, corrupted FT# messages, pure guesswork by operators, confusion over the signs for latitude and longitude values …
   - Out of date locations *e.g*. you may have worked and correctly logged a W6 in San Francisco some years back, but she has now retired to be with her family in North Carolina using the same W6 callsign: if the previous QTH was logged and is carried forward to future QSOs using the QSO mask, it will be wrong if you work her again unless you notice and correct it.
   - Errors in online callsign databases.
   - Pirates ahoy!

Copyright © 2026 Robert C. Furzer K4CY

Q. Every time I start Logger32, the *What to do?* message window tells me "There are no Keplerian elements for TRITON-1 loaded in memory". I tried downloading Keps via the Internet but the message still appears. Is there any way to stop it? I'm into terrestrial comms with no interest in satellites.

A. It seems likely the Keps you downloaded did not cover TRITON-1 (a Dutch cubesat launched in 2013 to test an experimental **A**utomatic **I**dentification **S**ystem, since you asked) … so try configuring a different default satellite, one for which there *are* Keps:

1. Click to open the satellite tab of the tracking window. The currently-selected satellite data window opens.
2. Click to open the dropdown satellite list, then click to select a satellite – any one, it doesn't matter which ▶ All the satellites that appear in the list had Keps in the set you downloaded most recently.
3. Click to open the DX Spots tab of the tracking window.
4. Close Logger32, then re-start it to confirm that the annoying TRITON message is gone.



Q. Why are there multiple overlays on my map?

A. Under some circumstances, a little bug (a bugette?) hiding deep in the map code may display *two* day/night overlays on the same background map, as if the Earth has two suns. Thankfully, it doesn't, or 'global warming' would be 'global frying to a crisp', and 'climate change' would be 'underwear change'. To resolve this anomaly, simply right-click the dodgy map to open a menu, then click <**Create custom base maps**> to redraw the map, dispensing with the false sun. The plural in that menu option is a reminder that it's worth clicking the other tracking window tabs and checking the other maps too, as they may also need to be de-sunned in the same manner.

Copyright © 2026 Robert C. Furzer K4CY

Section C

Logger32 advanced topics

Copyright © 2026 Robert C. Furzer K4CY

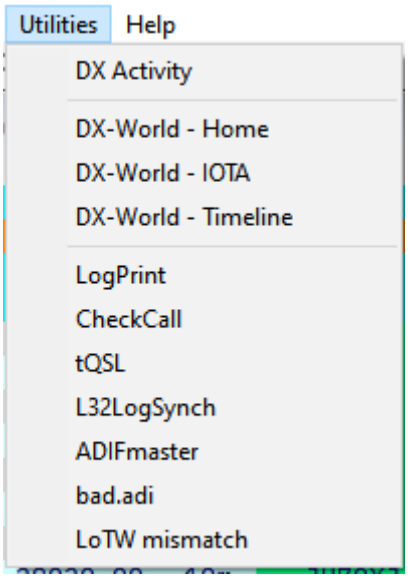
20 Add-on utilities

> "The difference between ordinary and extraordinary is that little extra"
>
> *Jimmy Johnson*

As if plain Logger32 wasn't already a sufficiently powerful multi-headed beast, additional utility functions can be added-on.

20.1 Pre-loaded utilities

Logger32 has four pre-defined/standard entries in two sections at the top of the <**Utilities**> menu[330], plus up to 20 user-defined entries beneath ▶



The top four options display information about past, present and future DXpeditions from DX websites and newsletters.

The remaining options towards the bottom of the menu (if any) are used to run utilities that *you* have configured.

Hinson tip: *your* <**Utilities**> menu will only resemble mine as shown here *if* you happen to have also manually configured entries for LogPrint, CheckCall, TQSL *etc*. as I have. Your utilities may vary.

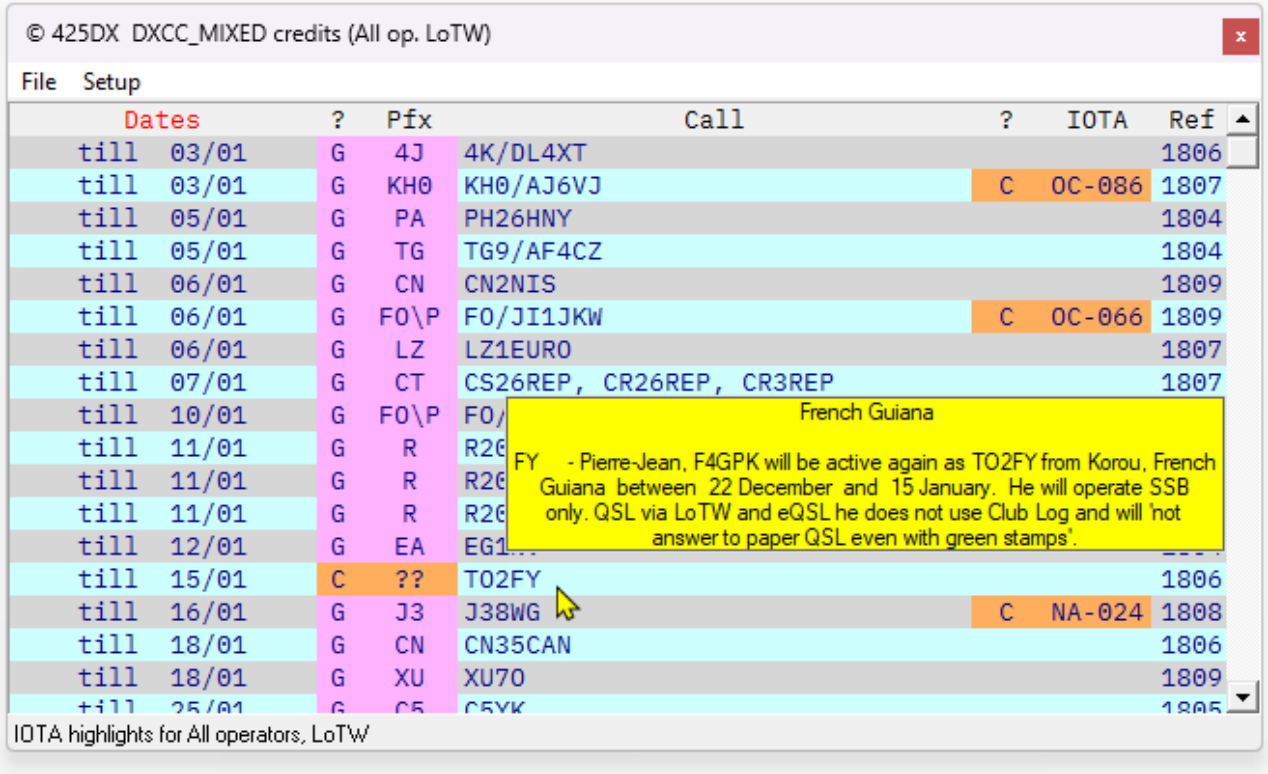
Read on for more …

[330] Some utilities are built-in and available by default) or add-ons (they are not integral parts of Logger32's program code). They are web-based information services, whose URLs are pre-installed.

Copyright © 2026 Robert C. Furzer K4CY

20.1.1 DX activity

Utilities ⇨ **DX Activity** displays a table of information drawn from the 425DX news bulletins ▼



The table is a representation of data downloaded from the 425 DX News Calendar.

Entries in the table are shown with Logger32's usual highlighting colors for 'new ones' (wanted for the configured DXCC award, or for IOTA).

The table can be sorted by clicking any column header, turning it red.

Mouse-over any callsign for a yellow popup with more information.

I have been following developments from Bob and everyone else for many many years of using Logger32 and a big thank you to all.

Paul GOWRE

Copyright © 2026 Robert C. Furzer K4CY

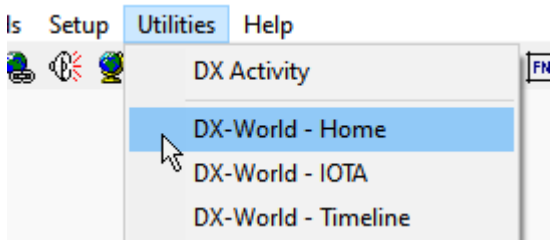
Right-click any bulletin Ref number to display it in your web browser ▼

```
13 December 2025                                        A.R.I. DX Bulletin
                                  No 1806

                          ============================
                          *** 4 2 5  D X  N E W S ***
                          ****  DX  INFORMATION  ****
                          ============================
                           Edited by  I1JQJ & IK1ADH
                         Direttore Responsabile  I2VGW

4K      - Jan, DL4XT will be active holiday style as 4K/DL4XT from Azerbaijan
          between 26 December and 3 January.  He will operate SSB, CW and FT8
          on 40 and 20-10 metres. QSL via LoTW, Club Log's OQRS and QRZ.
A7      - A71QND is the special callsign for the  Qatar Amateur Radio Society
          to celebrate Qatar National Day (18 December).
          Also taking part  in the celebrations  will be  the Bahrain Amateur
          Radio Society as A91QTR on 12-18 December, and the Emirates Amateur
          Radio Society as  A60QATAR on 18-22 December  (individual operators
          may append a serial number  to the special callsign),  and possibly
          the Kuwait Amateur Radio Society (9K9QATAR),  the  Royal Omani Ama-
          teur Radio Society as A43QAT,  and the  Saudi Amateur Radio Society
          (HZ1QATAR)
```

20.1.2 DX-World



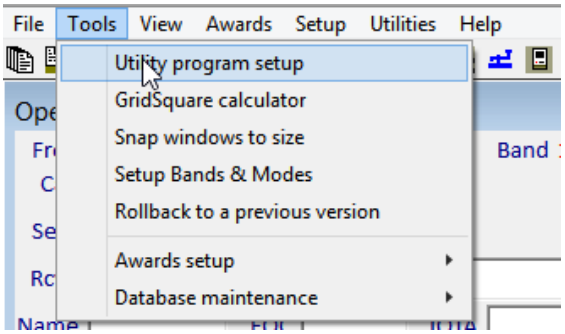
◀ The three DX-World items on the <**Utilities**> menu simply open pages from the DX-World website in your browser.

Browse the website for news about current and planned DXpeditions, special events, QSL routes *etc*.

20.2 Utility program setup

Up to twenty of your favorite PC utilities (typically radio-related programs such as L32 LogSync, CheckCall, CC, TQSL, ADIFmaster, LogPrint, ADIF2QSL, DXAtlas and others) can be listed and launched from Logger32's <**Utilities**> menu.

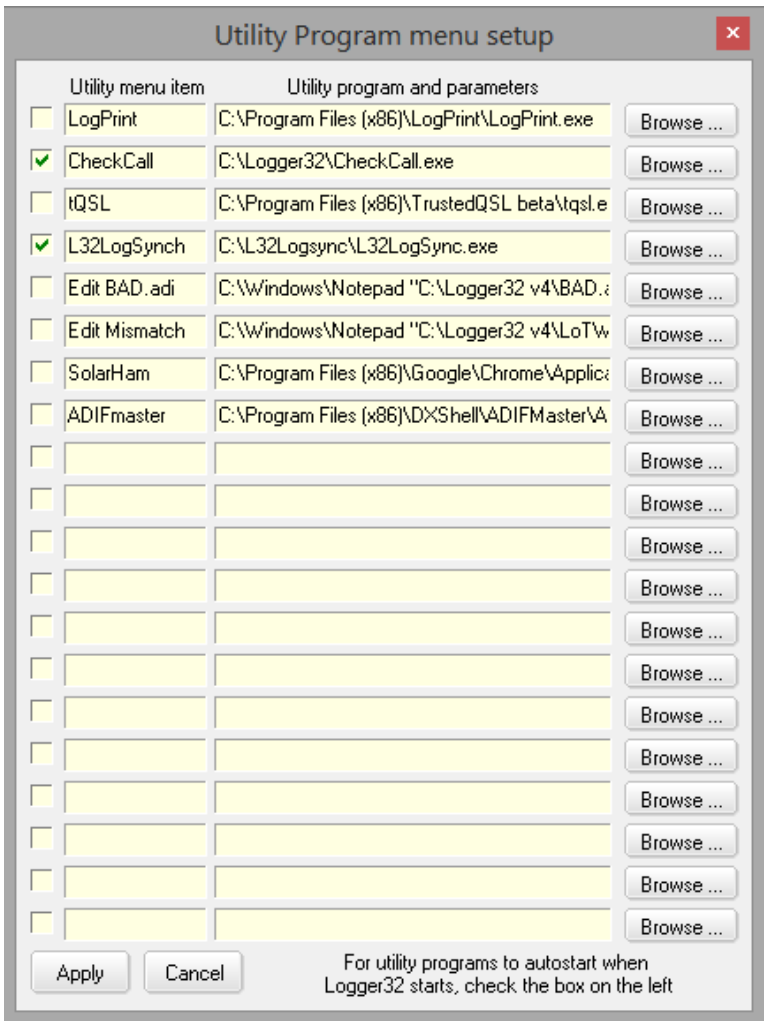
To configure them, first download, install and configure the programs, then open **Tools ⇨ Utility program setup** ▶



Copyright © 2026 Robert C. Furzer K4CY

The utility program setup form has 3 data-entry columns plus a column of <**Browse**> buttons ▼

- Click to tick the square box on the left to launch the program on that line automatically when Logger32 starts up.
- Give the utility a handy name in the <**Utility menu item**> column. This name appears on the <**Utilities**> menu.
- Specify the command line to launch the program, along with any command line parameters (enclosed with "speech marks" if the parameters include spaces).
- Optionally add the command line parameter /MIN to have the program start-up with its screen minimized, if you don't need to see or use it (*e.g.* with CheckCall).
- If you're not sure where the program is, click <**Browse**> to look for it on your system using File **E**xplorer.



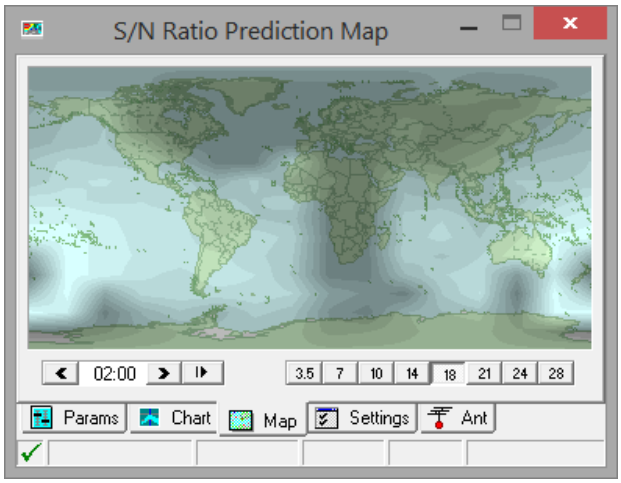
When Logger32 auto-starts an application it remembers the unique Windows Task ID number that was assigned to the application. When Logger32 closes, it looks at every window that is currently open and checks to see if the parent of this window has the task ID in question. If so, Logger32 sends a windows CLOSE message to the window.

Bob K4CY

Hinson tip: as illustrated by the DX-World entries, the Utilities menu can open arbitrary web pages in your browser. Simply put the path and filename for your web browser in the <**Utility program …**> column, followed by a space and then the URL you wish it to open. You can of course bookmark ham websites in your browser too, in the normal fashion. The IOTA website's map page, for instance, is one you might like to add to your Utilities menu: www.iota-world.org/iotamaps/

Copyright © 2026 Robert C. Furzer K4CY

20.3 Ham CAP propagation prediction



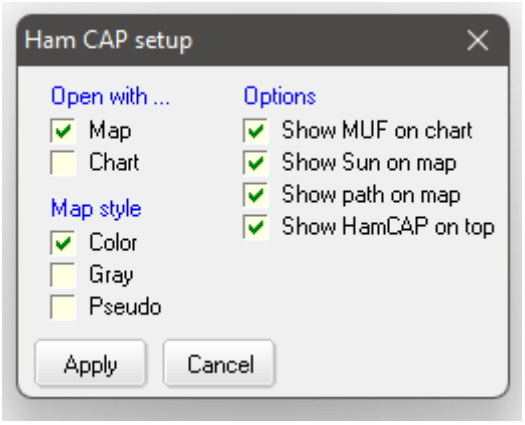
Logger32 supports Ham CAP, a freeware amateur propagation prediction application from Alex VE3NEA on the DX Atlas website ▶

It is based on the freeware "ITS HF" modeler originally developed by the US government as VOACAP to help shortwave listeners around the globe monitor the enthralling Voice Of America broadcasts. More specifically, Logger32 can update the solar and geomagnetic data in Ham CAP from WCY's messages circulating on the DX cluster network, and pass information about the DX stations you are chasing/logging into Ham CAP.

20.3.1 Install and configure Ham CAP

1. Download the latest **ITS HF** engine from the Greg-Hand website: look down the file list on the *www.greg-hand.com/versions/* webpage for the most recent *itshfbc_nnnnnnx.exe* file near the bottom – currently *itshfbc_180417a.exe* as of March 2026[331].
2. Run the downloaded *.exe* file, choosing the default installation folder *C:\itshf* or another one with a short name that fulfils the limitations spelled out by the installer.
3. Download **Ham CAP** as a *.zip* file from the DX Atlas website.
4. Un-zip the downloaded file to a temporary folder such as *C:\HamCAPtemp*[332]
5. Install Ham CAP by following the *readme.txt* instructions unzipped from the downloaded file. The defaults work for me.
6. Ham CAP comes with an *ancient* (2017!) *SSN.dat* file, saved by the installer in the folder *%appdata%\afreet\reference* so:
   - Download the updated solar/geomagnetic data file ***ssn.dat*** (covering 2022-2031) also from the DX Atlas website. This 2022 file is not quite as old.
   - Move the newly-downloaded *ssn.dat* file into the folder *%appdata%\afreet\reference*, overwriting the ancient file.
7. At this point, Ham CAP should run as a stand-alone program. Feel free to give it a whirl: put it through its paces and find out what it can do for you. If you find it useful, you can launch it from within Logger32 …
8. In Logger32, configure your preferences under **Setup ⇨ Ham CAP** and click <**Apply**> when done ▶
9. Launch Ham CAP using icon #17 ▼ on the toolbar.





[331] You won't be running this directly but Ham CAP uses it as its modelling engine, 'behind the screen'.

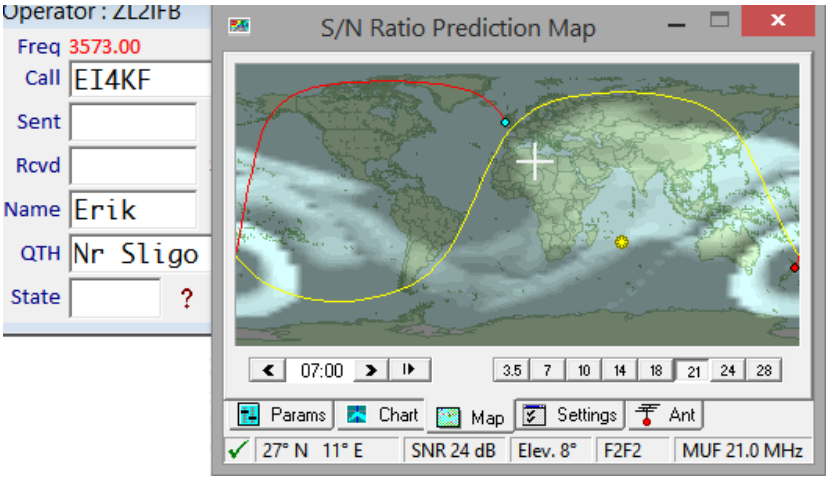
[332] You can delete it later: it is only used while installing Ham CAP.

Copyright © 2026 Robert C. Furzer K4CY

10. Open the <**Params**> tab in Ham CAP to configure your location, RF transmitter power, long or short path.
11. Configure your antennas for each band on the <**Ant**> tab. The program uses approximate antenna gain and take-off angles as part of its modeling. If your specific antennas are not shown, something vaguely similar will do fine.

20.3.2 Using Ham CAP

With a DX callsign in the log entry pane, Ham CAP uses the time, band, DX callsign and your location information to display its propagation predictions on a world map or chart ▶



Light areas on the color map predict where your signal can be received. Mouseover them to see (across the bottom) the predicted **S**ignal to **N**oise **R**atio and peak signal elevation (take-off) angle, plus the ionospheric propagation mode and **M**aximum **U**sable **F**requency.

Click any of the band buttons to check out the path on other bands. Is it worth QSYing to another band? Click to change the UTC time. Maybe conditions will improve later on?

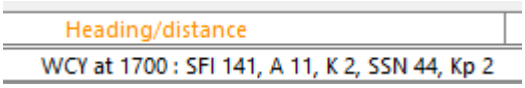
20.3.3 Update Ham CAP's solar and geomagnetic parameters

For its propagation predictions, Ham CAP uses the **S**un **S**pot **N**umber and smoothed **p**lanetary **K**-index (Kp) figures from the downloaded *SSN.DAT* data file, having been *estimated* years in advance. Predictions will be more accurate if the *estimates* are replaced with actual measured values. Both values are available from DK0WCY's magnetogram and are circulated every few hours on the DX Cluster network.

Logger32 can grab the latest figures and paste them into Ham CAP *provided* that:

1. You are using a DX Cluster node that routinely receives and forwards the WCY messages. Most do … but you may need to configure the cluster not to block the WCY messages.
2. You have been connected to the DX Cluster node long enough to receive a WCY message. If you recently started Logger32 or just logged-in to the cluster, you can send the cluster command **SH/WCY 1** to have the cluster send you just the most recent WCY message[333].

Having been received, the WCY data appear on the status line ▶

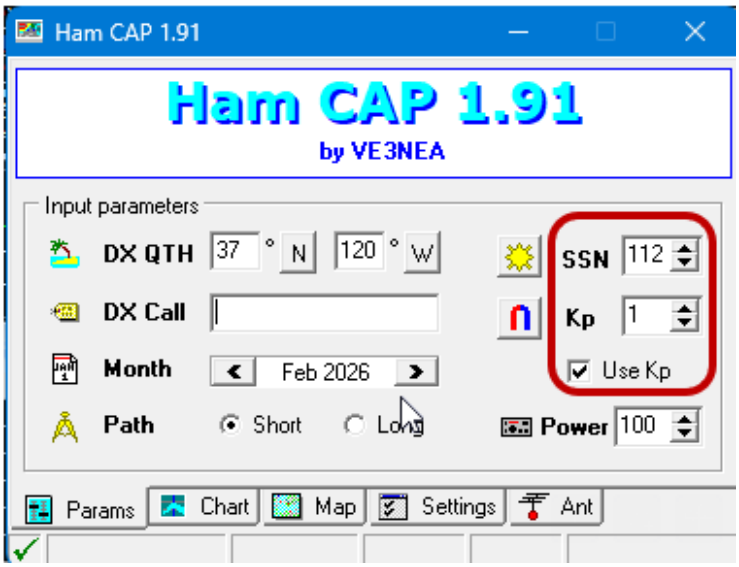


[333] As with WWV, if you omit the **<space>1** from the SH/WCY command, the cluster will probably send *several* recent WCY messages, starting with the most recent. Logger32 casually ignores the WCY message timestamps, processing them sequentially and ending with the *last* message it received, *not* the most recent values. To be honest, in practice, it makes little difference unless conditions are disturbed.

Copyright © 2026 Robert C. Furzer K4CY

Alternatively, visit DK0WCY's magnetogram web page in your browser for the current figures (look for "Sunspot Number" and "Kiel 3-hour k"), manually entering them into Ham CAP through the <**Params**> tab ▶

Aki's tip: if you change the month in Ham CAP (*e.g.* to predict HF conditions and plan to contact a forthcoming DXpedition), it pulls and uses the *predicted* SSN and Kp values from SSN.DAT file for the chosen month, rather than the current values.



20.4 NCDXF beacons

Beacon	Frequency
4U1UN	silent
VE8AT [2]	silent
W6WX	28.200
KH6RS [4]	24.930
ZL6B	21.150
VK6RBP	18.110
JA2IGY	14.100
RR9O	silent
VR2B	silent
4S7B [4]	silent
ZS6DN	silent
5Z4B [7]	silent
4X6TU	silent
OH2B	silent
CS3B	silent
LU4AA	silent
OA4B	silent
YV5B	silent

The **N**orthern **C**alifornia **DX** **F**oundation, in cooperation with the **I**nternational **A**mateur **R**adio **U**nion, funds, operates and maintains a time-synchronized (sequentially transmitting) worldwide network of 18 HF propagation beacons on 14100, 18110, 21150, 24930 and 28200 kHz. Read all about the beacons, including the operating schedule and current status, at www.ncdxf.org/pages/beacons.html.

◀ The beacons transmit in sequence[334]: according to the schedule, each beacon transmits a ~10 second[335] message on a given frequency once every three minutes, around the clock.

Each transmission consists of the callsign of the beacon sent in CW at 22 WPM followed by four one-second dashes. The callsign and the first dash are sent at 100 watts. The remaining dashes are sent at 10 watts, 1 watt and 100 milliwatts,

[334] The NCDXF beacon schedule has footnotes with news regarding the status of the beacons. The beacons are fairly reliable but sometime suffer equipment failures and other problems that take them off-air, sometimes for months and occasionally for *years*. Check out those little footnote references against beacon callsigns on the online schedule.

[335] The callsigns are different lengths, hence the messages are not all *precisely* 10 seconds long.

Copyright © 2026 Robert C. Furzer K4CY

respectively. If you can hear the QRP dashes, there is clearly a good path between you and the beacon on that band at that moment.

In principle, you can simply spend 3 minutes listening on a beacon frequency to figure out which part of the world has propagation to you right now … but, in practice, not everyone can copy CW at 22 WPM, especially if a beacon is very weak, surfing the waves of QSB.

However, because the beacons transmit on specific frequencies at known times, you can tentatively identify a beacon purely from the published schedule without necessarily copying its full callsign, *provided*:

- Your receiver is tuned to one of the beacon frequencies in CW or SSB mode.
- Your clock, and the beacon clocks[336], are reasonably accurate.
- The beacon frequency remains clear of interference … particularly CW transmissions about 10 seconds long.

Hinson tip: for the sake of patient DX listeners, *please* do not transmit on the beacon frequencies.



◀ Click the toolbar pie chart icon #21 to open the NCDXF function. The beacon scheduled on a given band and time is identified with a distinctive marker on the world map ▼

The beacon currently scheduled on the present beacon frequency is the yellow-highlighted row in the center of the table, the active time showing in the left-hand column and its frequency on the right.

Beacons that you have marked as inactive (see below) have a darker background in the table and different colored markers on the map – red in my case ▲





The beacons may be listed in turn on one frequency …

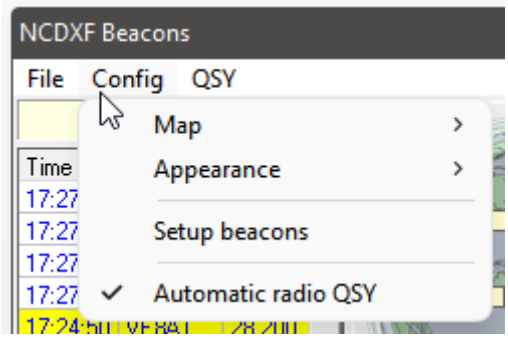
◀ … or a selected beacon can be shown with its sequence of transmission frequencies.

[336] The NCDXF beacons are GPS-synchronized, although there are sometimes technical issues with the beacon controllers and transmitters due to them operating unattended, continuously, in harsh environments. On rare occasions, the beacons have transmitted at the wrong times or frequencies, sent garbled messages, failed to change power levels or gone QRT. Donations to NCDXF's beacon fund are welcome.

Copyright © 2026 Robert C. Furzer K4CY

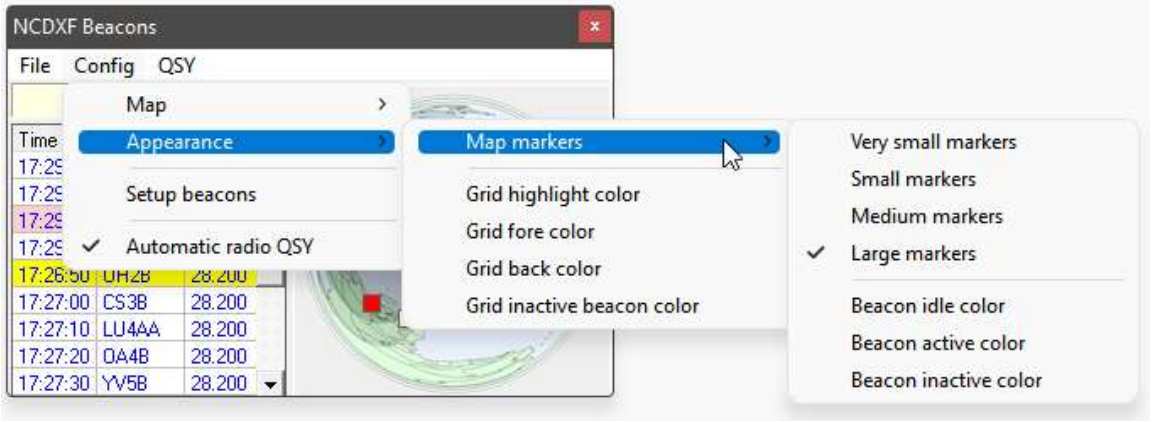
Hinson tip: Mouseover any beacon's marker on the map for a tooltip showing its callsign and location (if so configured). And yes, before you even ask, you *can* change the colors. For me, 'traffic light colors' are intuitively obvious. I find the great circle map useful, showing daytime, grayline and nighttime areas. See the next section for notes on configuring your installation.

20.4.1 NCDXF beacon menu

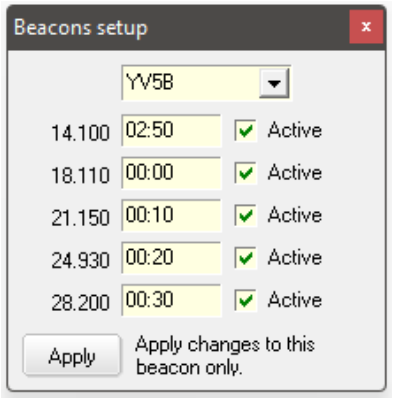


With the NC DXF beacons pane open:

- **File**: click <**Exit**> to close the NCDXF window, if clicking the corner X is too hard**.**
- **Config** is used to configure various settings ▶
  - **Map**: choose the map projection, switch the map display on/off, and optionally display the tooltips and a grayline overlay.
  - **Appearance**: opens a submenu to set your preferences for the tabular display, including the highlighting, grid and beacon marker colors ▼



  - **Setup beacons**: adjust time slots and activity ▶



    For the beacon selected at the top, click to tick the appropriate frequencies if the beacon is QRV, and confirm its scheduled time slots, then click <**Apply**>.

    Hinson tip: some beacons are in physically challenging locations, and may experience hardware failures. If you are copying other stations in the area but not the beacon, it may be QRT. For news about beacon problems, planned maintenance, unplanned outages *etc*., visit the NCDXF web site.

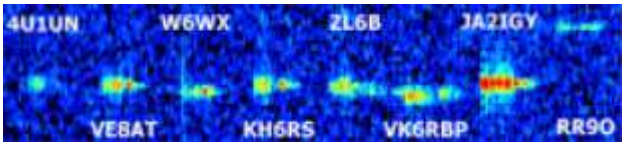
  - **Automatic radio QSY**: track a given beacon as it moves from band-to-band, your VFO following the scheduled sequence[337].
- **QSY**: when enabled (not grayed out), clicking this menu item immediately tunes your CAT-connected radio to the active beacon frequency.

[337] This only happens if the focus is on the NCDXF window (if not, click it!) with a beacon selected from the drop-down list.

Copyright © 2026 Robert C. Furzer K4CY

It is possible to use the spectrum or waterfall display in the Sound card data window to monitor audio received on the beacon frequency like a panadapter. The NCDXF transmitter VFOs are *not* highly stable and calibrated frequency standards, however, unlike WWV and the like.

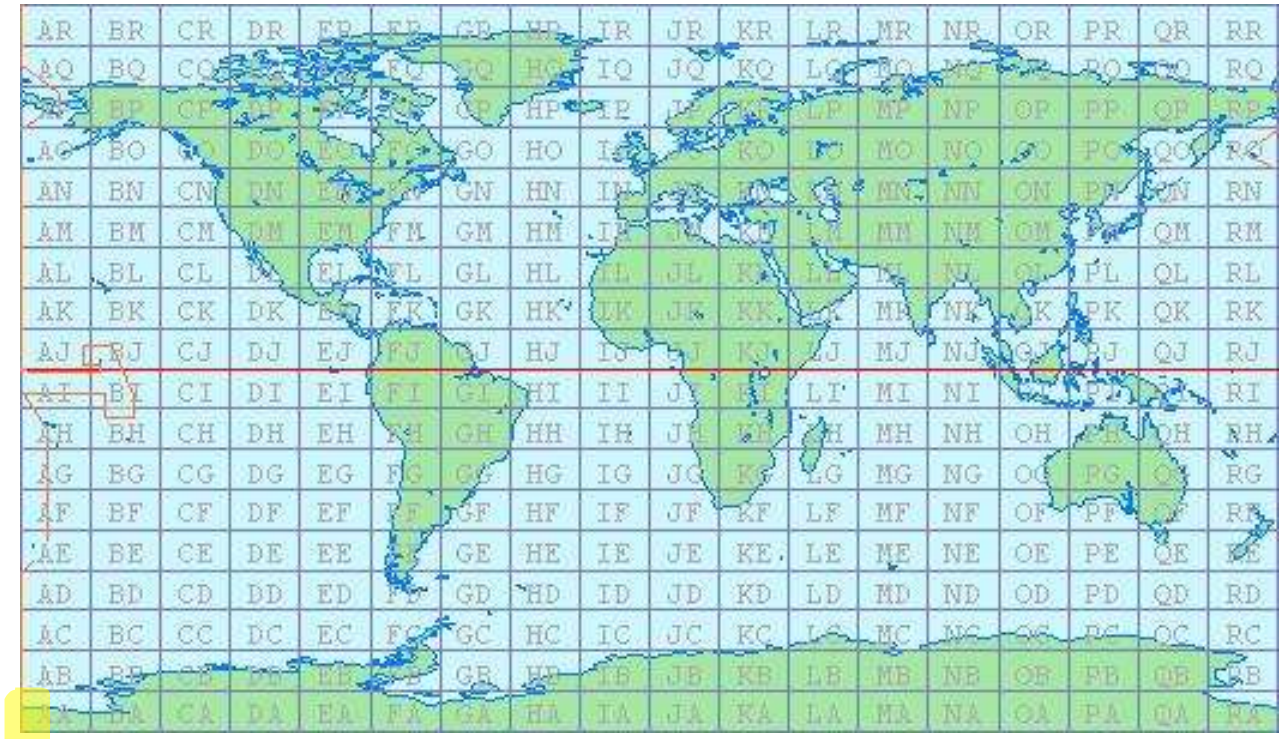
As of April 2024, RR9O was noticeably HF of the others on 10m, by a few hundred Hertz ▶



20.5 Grid squares (locators)

20.5.1 The Maidenhead locator system

The Earth's surface is divided into 18 x 18 = 324 **fields**, each one being 20° longitude (East ↔ West) by 10° latitude (North ↕ South). Fields are denoted by pairs of capital letters in the range AA-RR: the first letter denotes the longitude, the second the latitude ▼



The sequence always flows West → East and South ↑ North. The starting longitude is -180°, while latitude starts at the South pole (-90°). It is like an XY reference on a graph, with the origin at the bottom left corner of the world map above.

Each field is subdivided into 10 x 10 = 100 **squares** of 2° longitude by 1° latitude. Squares[338] are denoted by two decimal digits 00 - 99.

Each square is further divided into 24 x 24 = 576 **subsquares** of 5 minutes longitude by 2.5 minutes latitude. Subsquares are denoted by two lowercase letters aa - xx.

Therefore, a 6-character locator (for example FN43mj) specifies a location to within 2.5 **n**autical **mi**les North ↕ South and between 0 and 5nmi (depending on latitude) East ↔ West.

[338] Despite the name, they are not literally square in shape because the Earth is not, in fact, entirely flat, despite what some intellectually-challenged people claim.

Copyright © 2026 Robert C. Furzer K4CY

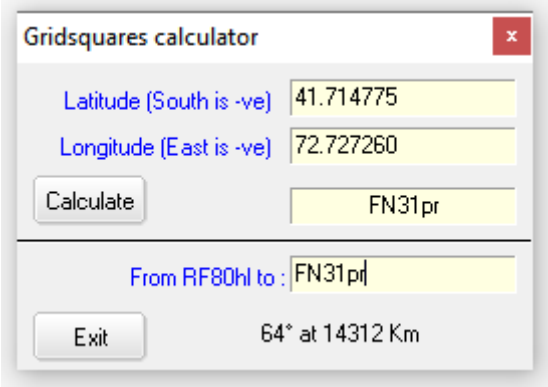
20.5.2 Grid square calculator

If you need to calculate a grid square reference ('locator') from latitude and longitude, or determine the distance and bearing from your station location to a DX grid square, a simple utility is provided in Logger32.



◀ Click the toolbar FN icon #18 to open the GridSquare Calculator **F**unction**N**.

The calculator has two sections ▶



- The upper section calculates the grid square for a specified latitude and longitude. Enter the latitude and longitude as decimal numbers (degrees and fractions of a degree), positives for North and West, or negatives for South and East, then click <**Calculate**> to display the locator.
- The lower section calculates the great circle bearing (azimuth) and short path distance from <**My QTH**> (configured by right-clicking the log entry pane and clicking **Setup ⇨ My QTH Lat/Long**) to the center of the DX grid square you have entered[339].

The same algorithm is used to determine a distance and bearing to a grid square reference[340] entered into the log entry pane (*e.g.* most stations send their 4-character grids in their CQ messages on FT#). The result can be displayed in the DX info bar just above the status bar[341] ▼



Hinson tip: for nearby grid squares that are very close to <**My QTH**>, the bearing and/or distance may *appear* incorrect. Logger32 is in fact calculating correctly: the *apparent* error is because in reality stations are seldom precisely at the centers of their respective grid squares. That discrepancy matters in the nearfield. Don't rely on grid square calculations to point your microwave dish, laser or mortar at a neighbor, or to confirm that you are 'socially distanced'.

[339] You have to type the grid square in to the lower box to calculate the bearing and distance, even if it is shown in the upper box.

[340] If there is no grid square, Logger32 uses the approximate geographical center of the DXCC entity instead, or a nearby city: the specific point is defined by latitude and longitude in the Country database.

[341] Yes, there are several bars in Logger32. We're still hunting for the jukebox and pool table, beer nuts, the snack menu and not least the Famous Grouse optic.

Copyright © 2026 Robert C. Furzer K4CY

As a new user of Logger32, I would like to thank the builders of this fine piece of software. Following some problems with a few of the features, which the community helped me out with, I now have the setup exactly as I would wish. The add-ons by N2AMG are also very welcome and useful.

Gwyn, G4FKH

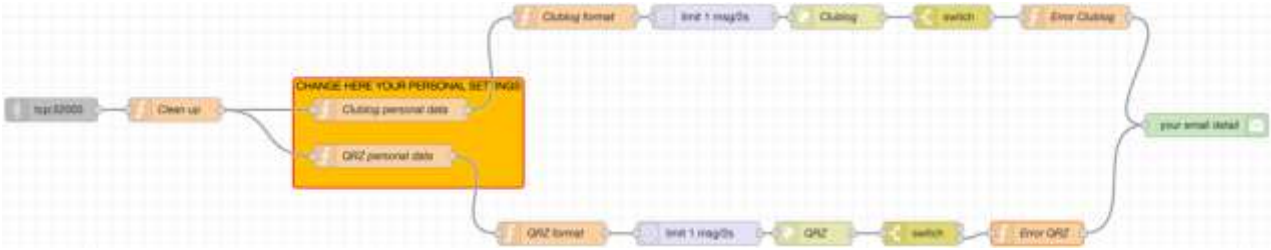
20.6 L32LogSync

L32LogSync is a free add-on utility application written by Rick N2AMG, enabling Logger32 users to automate the uploading of QSOs to, and where applicable downloading of confirmations from, online log/QSL services such as Club Log, LoTW, eQSL and HRDlog.net. It is possible to upload individual QSOs automatically as soon as you have logged them, or to send them in batches with just a few clicks.

For more information on L32 LogSync including the download link and the latest update, visit www.n2amg.com/software/l32-logsync/

20.7 Node-RED

Node-RED is an open-source visual tool intended for non-programmers to build automation applications. Node-RED applications can automate tasks such as monitoring and controlling hardware devices, passing data to network services, responding to events and visualizing data. Automation flows are constructed by dragging and dropping nodes in a browser-based editor.



For example, Andrea IZ2LSC has built and shared a workflow in Node-RED ▲ to receive QSOs from Logger32 via TCP as they are logged, then upload them to QRZ.com and Clublog, reporting any upload errors via email.

Hinson tip: Node-RED has further potential for shack automation and monitoring. Worth a look?

Copyright © 2026 Robert C. Furzer K4CY

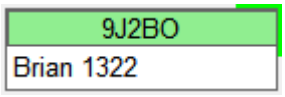
Bob, the information you provided was spot on. Thank you. I have not seen the manual for some time – it is quite the document, a testament the contributors. Wow the program has evolved so much from the old logger! It is a great program with a wealth of features. I have macros that help with some of the configuration tasks for various modes. Again, thank you.

Jay N3OW

20.8 CheckCall

CheckCall, a simpler version of the HamQTH or QRZ callsign lookup facility, is a free Logger32 add-on utility, one of 17 utils available from Aki JA1NLX at the start of 2026 ▶

CheckCall ver8.2
CheckCallMini ver7.2
CheckSpot ver4.02
Lookup4CNTY ver2.4
LookupMaster ver1.16
ModeConverter ver1.01
NixieClock ver 1.3
SummaryV2 ver2.04
SummaryAllTime ver2.0
SummaryThisMonth ver1.2
SummaryThisYear ver2.2
SummaryLast30Days ver2.2
SummaryIOTA ver2.72
UDP decoder ver1.04
USACA_MARAC ver1.2
utlStart ver1.3
UploadDownload ver4.2



◀ CheckCall displays a little pane showing information about the station you are currently logging. The information shown comes not from an online database but from a plain text data file on your disk, each line listing a callsign plus the associated text to display as you are logging it.

With CheckCall installed and running[342], tabbing or moving away from the Call field of the log entry pane triggers CheckCall to look up the callsign in the data file and, if an entry for that callsign is found, the window pops up showing the data line.

Right-clicking the top line caption opens a configuration menu with a few self-evident options.

The compact CheckCall window can be moved to any convenient space on your displays.

Hinson tip: I use CheckCall to show member numbers and names of my friends in FOC as I am working them. Other amateur radio club membership lists and so forth would work just as well, provided each line starts with a callsign.

Hinson tip: Aki kindly supports and maintains his utilities, making changes as he actively develops and refines them for his own purposes. If you find that a given version of a utility does exactly what you want, you might like to archive a copy of the utility program just in case you need to reload it after Aki drops or changes the utility in ways that don't suit your preferences or needs. Aki offers limited email support for older utilities but cannot work miracles.

[342] Add-on utilities such as CheckCall and L32LogSync can be launched *automatically* when Logger32 starts up by adding the programs and ticking the entries in **Tools ⇨ Utility program setup**.

Copyright © 2026 Robert C. Furzer K4CY

20.9 JTAlert

For instructions on integrating JTAlert with both Logger32 and JTDX, see the TCP Server chapter.

20.10 N1MM+ Bridge

Details of this add-on linking the N1MM+ contest logger with Logger32 in real time are also in the TCP Server chapter.

20.11 Add-on utilities FAQ

Q. I have a problem with [an add-on utility] …

A. Visit the relevant website and if necessary contact the author of the utility for assistance. In theory, we can only help you with Logger32 and its built-in utilities on the Logger32 reflector … but in practice we'll do what we can to assist.

Thanks to Bob and all the others that work so hard on this program. I used it 5 or 6 years ago and can't remember why I switched. Having used most all the major programs, now that I'm back I find Logger32 an amazing program. The manual is great also. Thanks so much for giving to the amateur community this way!

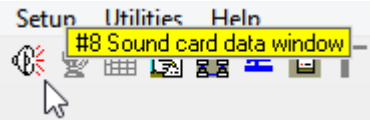
Jim N4KH

Copyright © 2026 Robert C. Furzer K4CY

21 Sound card digital modes

> "Digital technology is not a barrier, but a bridge that connects us to the world"
>
> *ChatGPT*

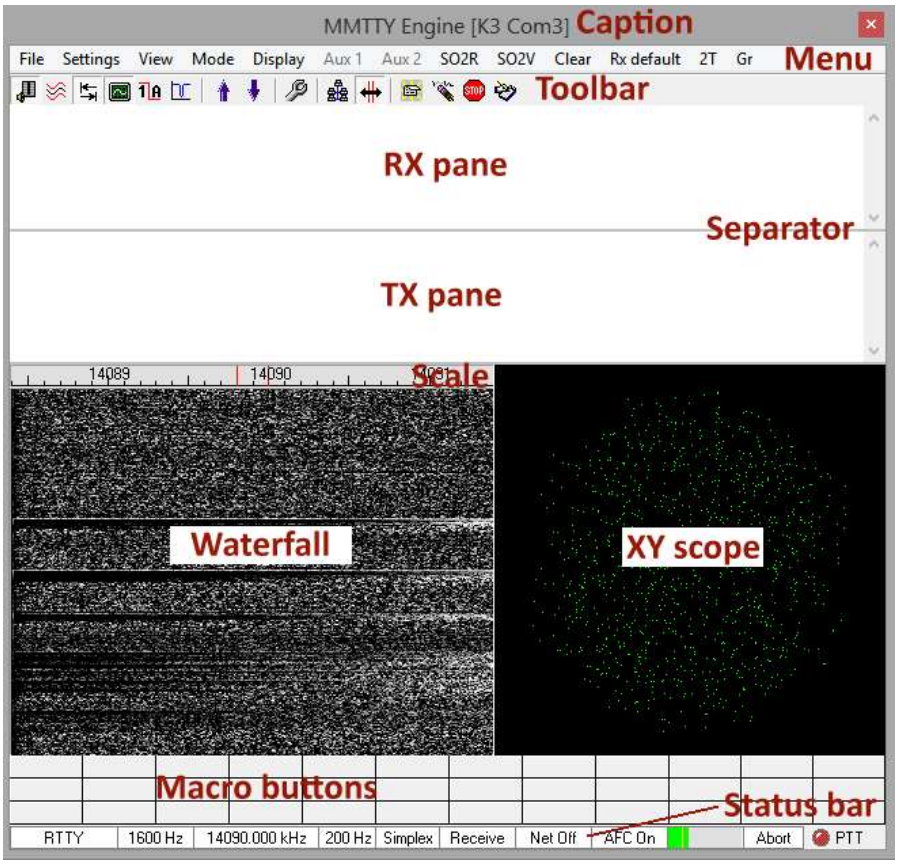
Logger32's **Sound card data window** supports PSK and RTTY modes[343] using a computer sound card and (ideally) a CAT-connected transceiver.



◀ Open it by clicking the speaker icon #8 on the toolbar, or from the <**View**> menu.

21.1 Sound card data window panes

The window is divided into several panes and panels ▶



They are the same in MMTTY and MMVARI except where noted below.

- **Caption**: shows which soft modem and radio you are currently using.
- **Menu**: offers very similar configuration options for MMTTY and MMVARI. More below.
- **Toolbar** (MMTTY only): buttons give ready access to various MMTTY functions. More below.

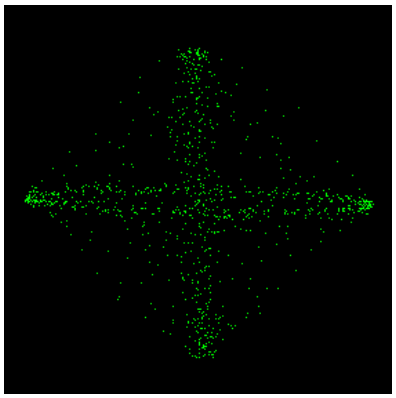
[343] Logger32 also supports more modern digital modes such as FT# through digimode software such as JTDX and WSJT-X. Although they are also sound card digimodes, they are covered in the UDP BandMap chapter, not here.

Copyright © 2026 Robert C. Furzer K4CY

- **RX|TX pane**: received and decoded signals are shown on the upper RX pane. Type the text you want to transmit into the lower TX pane.
- **Separator**: while the RX and TX panes share an area, the dividing line between them can be dragged up or down with the mouse to vary the proportions for each[344]. When the mouse cursor is positioned on the line, it changes to an up/down pointer ▶

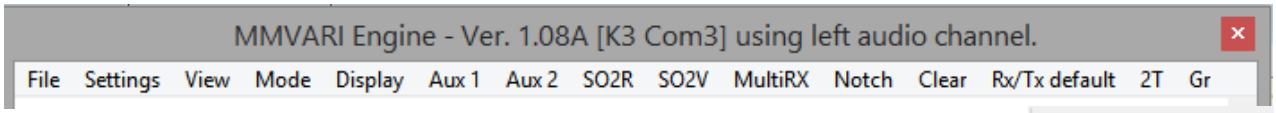
  The same thing applies to the upper edge of the frequency scale and XY scope (if shown in MMTTY).
- **Scale**: the frequency scale represents the range of audio frequencies arriving at your sound card's line or microphone input. It can display either audio frequencies or the corresponding radio frequencies (*e.g.* with the VFO set to 14080 kHz, a 1 kHz audio signal will be shown at 14081 kHz on the RF scale).
- **Waterfall** or **spectrum** tuning display: either of two display formats, both showing the strengths of signals across the received frequency range – see below. MMTTY's display is monochrome. MMVARI's has more color.
- **XY scope** [MMTTY only]: this digital simulation of a classical analog tuning aide displays a perfectly-tuned RTTY signal as a pair of ovals, one horizontal and the other vertical ▶

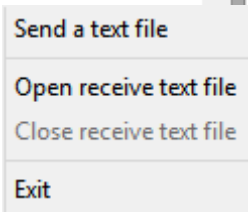
  The original setup involved feeding the output of the mark audio filter to the X plates of an oscilloscope and the space signal to the Y plates. Although essentially redundant since MMTTY's **A**utomatic **F**requency **C**ontrol does a good job of tuning RTTY, the XY scope is strangely captivating[345].
- **Macro buttons**: up to 4 rows of 12 buttons each can trigger macros consisting of commands and/or message text to transmit.
- **Status bar**: ten panels plus an "LED" indicator across the bottom of the window are informative and mostly clickable ▼ See below for more.



21.2 Sound card data window menu



- **File**: has options to send and receive text files over the air ▶

  You can prepare one or more "brag files" ready to send. These are text files containing content such as your name, QTH and station details that you would normally type out during a QSO. Invoke them from the <**File**> menu to have the text sent faultlessly without all that frantic hunting-and-pecking at the keyboard.

[344] Space for 3 or 4 lines of TX text is sufficient for me, maximizing the amount of space remaining for RX decodes.

[345] The XY scope takes a fair amount of processing: disable it to conserve computing resources for more important tasks – such as RTTY decoding – if your machine is seriously under-powered.

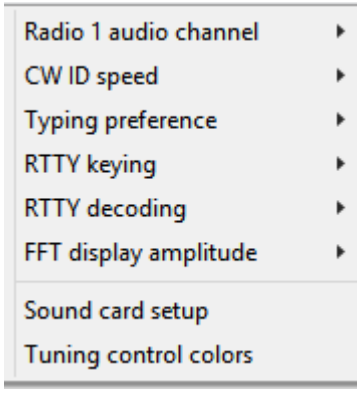
Copyright © 2026 Robert C. Furzer K4CY

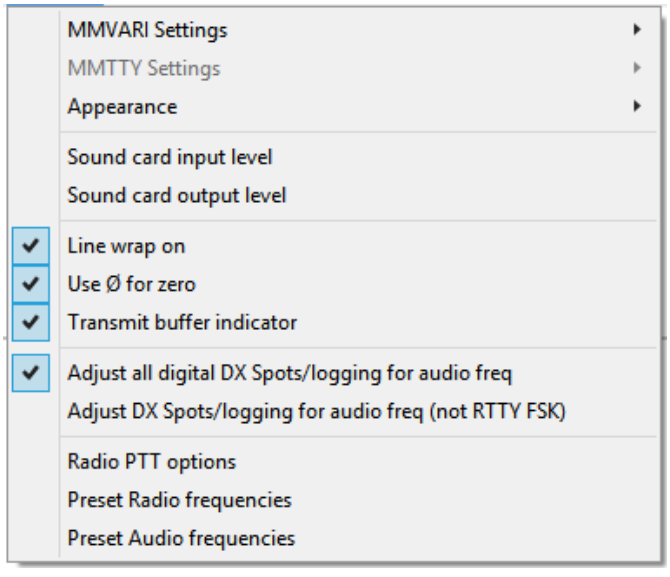
- **Settings**: configuration options – see below.
- **View**: see below.
- **Mode**: does dual service. You can select from either the MMTTY or MMVARI engines, and then choose from the various data communications modes offered by each program – see below.
- **Display**: choose between the spectrum and waterfall format tuning displays – see below.
- **Aux 1|2**: run supplemental decoders to monitor 1 or 2 other signals in their own little windows, in addition to the main one.
- **SO2R**: opens another MMVARI window for Radio 2 in an SO2R setup.
- **SO2V**: opens another MMVARI window for the sub-receiver on Radio 1. Typically, the main decoder takes its audio from one channel of the stereo output from the SO2V radio corresponding to the main receiver (VFO A), while VFO B/the sub-receiver's audio passes over the other stereo channel to another decoder in a separate window.
- **MultiRX** (MMVARI only): opens a multi-channel decoder in its own window – see below.
- **Notch** (MMVARI only): toggles a sharp audio notch filter at a frequency indicated by ▶ 

an N with a yellow background on the frequency scale. Click and drag the notch indicator left or right to null-out an interfering signal: you will see a 'hole' open up in the waterfall.
- **Clear**: wipes the RX and/or TX panes of text, more efficiently than a wet squeegee.
- **Rx/Tx default**: determines where on the audio spectrum your RX and TX are positioned. If you use GRITTY, you might like to add a default setting of 2285 Hz (GRITTY's fixed frequency).
- **2T**: launches the 2Tone RTTY encoder/decoder.
- **Gr**: launches the GRITTY RTTY decoder.

21.2.1 Sound card data window *Settings* menu ▼

- **MMVARI|MMTTY Settings**: the engines share some settings while others differ.





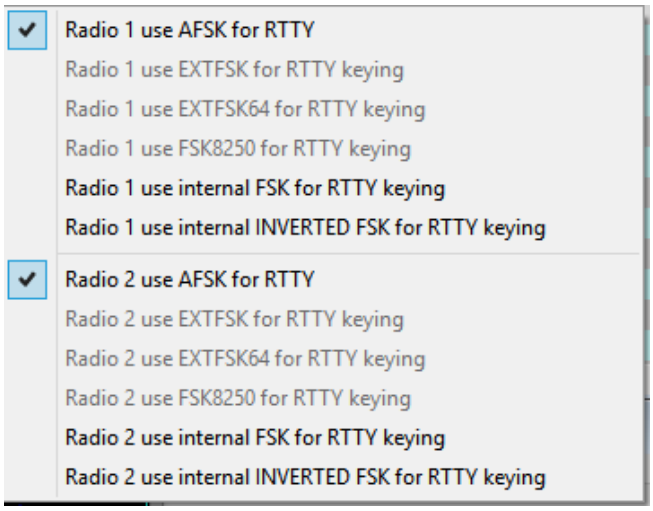
▲ The **MMVARI settings** submenus[346] determine whether:

  - Radio 1's audio arrives at the PC on the left or right stereo channel (and *vice versa* for Radio 2 in an SO2R setup).
  - CW IDs should be sent at 9, 12, 19 or 37 WPM, if at all.

[346] With MMVARI running, the MMVARI settings are active, but not the MMTTY settings, and *vice versa* when MMTTY is running.

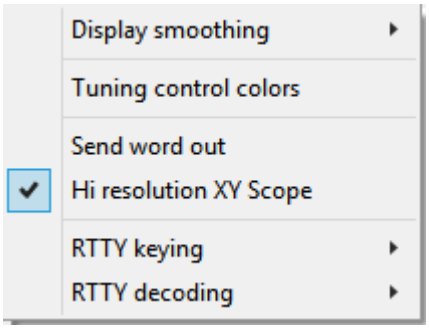
Copyright © 2026 Robert C. Furzer K4CY

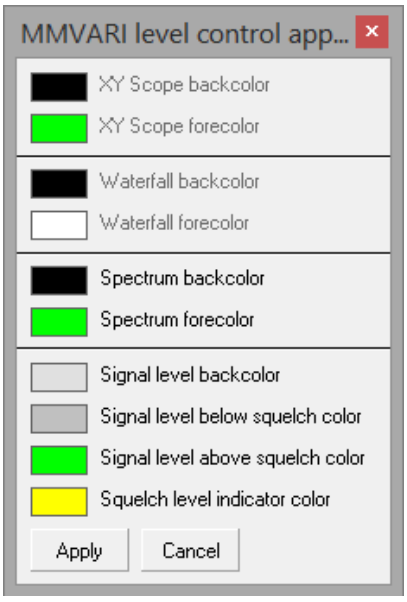
- On PSK, text should be sent in mixed case (as typed) or all CAPITALs or all lower-case.
- RTTY should be generated using AFSK, true FSK (normal or inverted[347]) or simulated FSK using optional drivers (EXTFSK, EXTFSK64 or FSK8250), on Radio 1|2 ▶
- To use **I**nfinite **I**mpulse **R**esponse or **F**ast **F**ourier **T**ransform decoding for RTTY. FFT is more effective but also more compute-intensive, so IIR may work better on under-powered computers.
- To use squared or logarithmic scaling of the display amplitude with FFT.



<**Sound card setup**> lets you choose which sound card/s to use for input and output, for Radio 1 and 2, and set the audio sampling rate. Faster is generally better provided the sound card and computer can cope with all those calculations.

<**Tuning control colors**> gives you the chance to fiddle with the colors of the XY scope and waterfall (only in MMTTY), plus the spectrum, signal strength bar and squelch line ▼





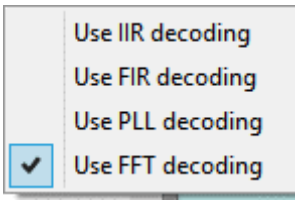
▲ The **MMTTY settings** submenus are:

- **Display smoothing**: smoothing successive samples removes outlying values and makes the display more representative of the 'average' signal levels but slows the update rate – so choose no, medium or heavy smoothing as you wish.
- **Tuning control colors**: offers the same color menu as for MMVARI ▲ except that now the XY Scope and Waterfall options are enabled.
- **Send word out**: slow, error-prone typists may like this option. Text is transmitted a word at a time when you type the following space. If you spot a typo *before* hitting the space, you can backspace and retype the word to correct it. Meanwhile, your RTTY transmission idles while everyone watching waits with bated breath for the next juicy morsel to emerge from their decoder.

[347] The normal/reverse icon on the MMTTY toolbar only affects the *received* signal polarity if you are transmitting with FSK, since the radio fixes the transmit frequencies. The FSK reverse option under RTTY keying inverts the keying signal sent to the transmitter.

Copyright © 2026 Robert C. Furzer K4CY

- **Hi resolution XY Scope**: if enabled, this paints more dots on the XY scope … slightly increasing the CPU load compared to the low resolution or completely disabled scope.
- **RTTY keying:** offers the same AFSK and FSK choices as for MMVARI - see above.
- **RTTY decoding**: offers 2 RTTY filtering/decoding algorithms ▶ in addition to MMVARI's IIR or FFT *i.e*. **F**inite **I**mpulse **R**esponse and **P**hase-**L**ocked **L**oop. It might be worth experimenting with this setting to find the best decoder for a given signal under specific operating conditions … but the conditions are constantly varying, and PCs differ in performance, so there is no 'ideal' decoder. You may not even notice any difference.



- **Appearance**: configure fonts, colors *etc*. See below.
- **Sound card input|output level**: these options open the Windows functions to configure the audio playback and recording devices. Click to select whichever device is connected to or embedded within your radio, then click <**Properties**>, then click to open the <**Levels**> tab. Aim to set the Windows and corresponding radio levels in the middle range if possible, avoiding the extremes.
- **Line wrap on**: Line wrap works the same way for PSK31 and RTTY.
  - **RX window line wrap**: Logger32 displays received text in the RX window exactly as it is received. The only exception is that when text reaches the right-hand edge, it wraps to the next line. There is no special line wrap: the next character (70) will appear at the left. It is up to the sending station to include carriage return/line feeds (CR/LF); otherwise long words will be split at character 70.
  - **TX window line wrap**: to transmit with line wrap, click Settings, Line wrap on. For RTTY using MMTTY, also check Settings, Typing Preference, RTTY - send word out. A check mark to the left of this choice means that it is on. On RTTY, if you do not check Send word out, Logger32 sends letter by letter.

    When you are typing in the TX window, text is wrapped between 63 and 69 characters. When there are 63 characters in a transmitted line, Logger32 waits for you to type a space in the text. If a space occurs between character 63 and 69, Logger32 replaces it with a carriage return/line feed (CR/LF). Remember, for RTTY mode, enable "word out".

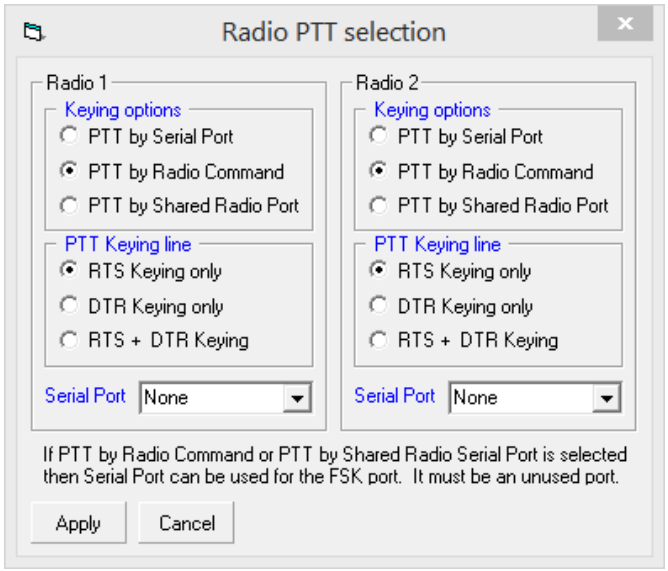
    If there is no space between character 63 and 69, Logger32 inserts a CR/LF after character 69 and puts what would have been the 70th character on the next line as the first character. To avoid long words being split across lines, avoid placing them at the end of a line by hitting <**Enter**> before typing them, or turn line wrap off in the <**Settings**> menu.
- **Use Ø for zero**: display zero as Ø rather than plain 0 to avoid confusing it with the letter O.
- **Transmit buffer indicator**: if you type-ahead text to be transmitted, this option colors the unsent and sent characters differently so you can see how far ahead you are. Change the colors in **Settings ⇨ Appearance ⇨ TX window buffer indicator color**.
- **Adjust all digital DX Spots/logging for audio freq**: automatically applies the appropriate audio and VFO offsets for RTTY[348] and PSK DX spots when they are clicked in the DX Spots pane, BandMaps or Tracking window.

[348] Logger32 presumes that a RTTY station's DX spot states his *mark* RF frequency.

Copyright © 2026 Robert C. Furzer K4CY

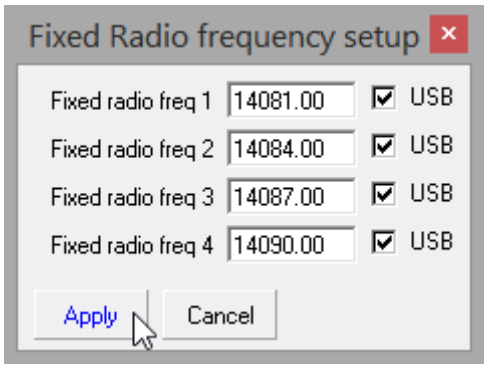
When you click a DX spot, Logger32 makes a series of checks. It looks up the spot frequency on your bandplan. If the sound card is open and the spot is determined to be RTTY or PSK, it sends the CAT-connected radio's VFO to the spot frequency plus or minus the appropriate offset to position the DX signal at the user-defined default frequency in MMVARI (for PSK or RTTY spots) or MMTTY (for RTTY only). This function is described further in the DX spots chapter.

- **Adjust DX Spots/logging for audio freq (not RTTY FSK)**: if you use FSK to transmit RTTY, there is no transmitted audio tone to adjust since the transmitter determines the mark and space frequencies directly.
- **Radio PTT options**: specify how to put your radio/s into transmit to send messages ▶



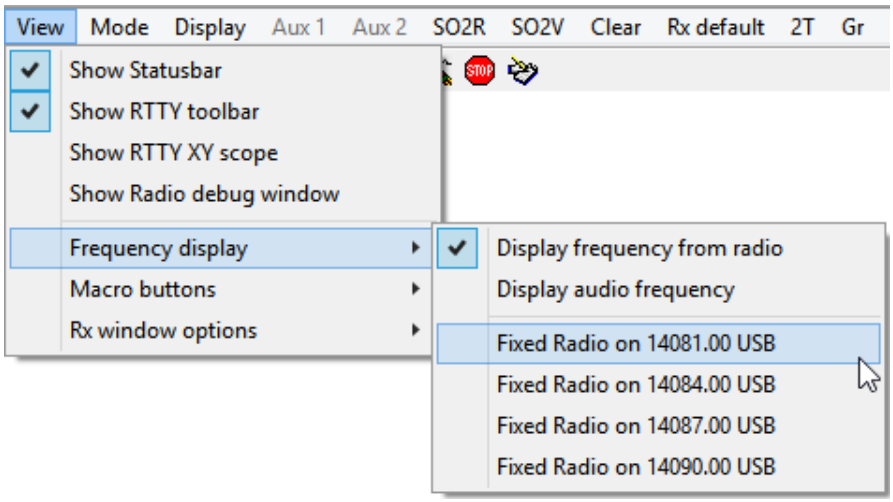
  Your options are, for each radio:
  - Send the appropriate radio command through the CAT interface. This is the preferred approach, if the radio has CAT, requiring no additional connections.
  - Control the RTS and/or DTS lines on a *dedicated* serial (COM) port which grounds the radio's PTT line through an appropriate PC-radio interface. Select the port from the drop-down list beneath.
  - Control the RTS and/or DTS lines through PC-radio interface on a *shared* serial port which is also used for FSK keying using the Tx data line.

  Once you have chosen the settings, click <**Apply**> to put them into effect and close the form.



- **Preset Radio frequencies**: if you are *not* using a CAT-connected radio, you can manually define 4 *static* VFO frequencies to which you manually tune your radio ▶

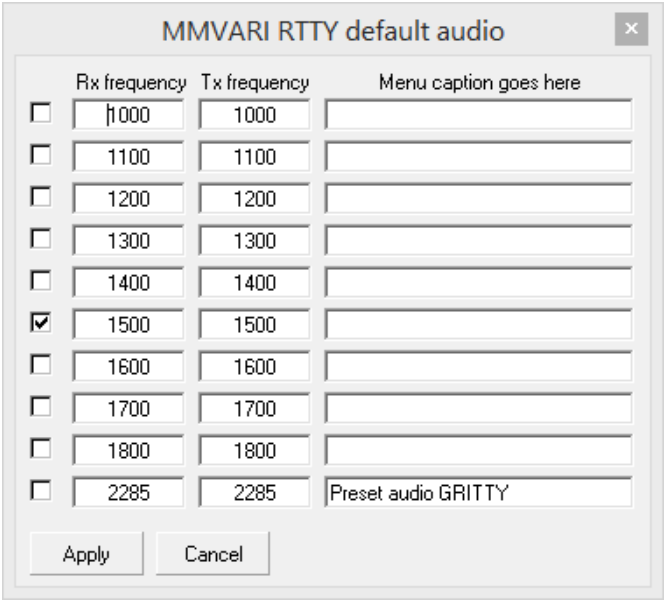
  These are then listed as scale options for the frequency display under the <**View**> menu ▶



  Having done so, the following macros work properly:
  $**RadioAndTone**$
  $**RadioFreq**$
  $**UpperOrlLower**$
  $**RXToneFreq**$

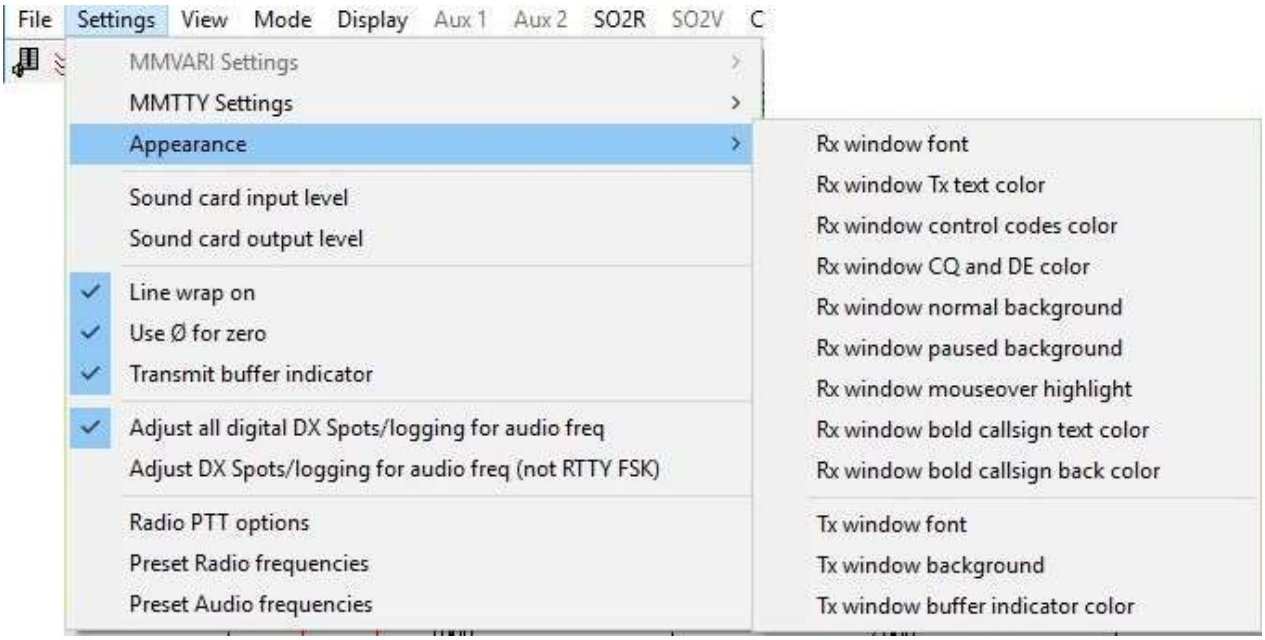
Copyright © 2026 Robert C. Furzer K4CY

- **Preset Audio frequencies**: configure and name up to ten pairs of RX and TX audio frequencies, and tick one pair as your default setting ▶



21.2.2 Settings ⇨ Appearance submenu

Settings ⇨ Appearance lets you get the RX and TX windows looking just right ▼

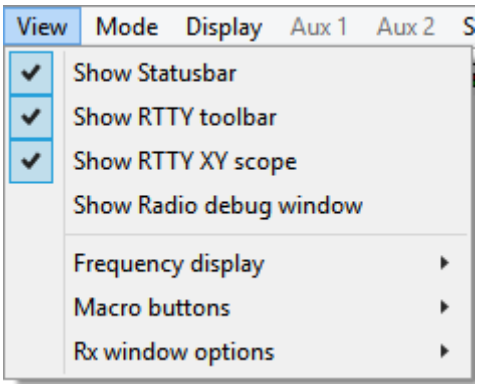


- **Rx window font**: sets print style and color of received text.
- **Rx window Tx text color**: sets the color for echoed text *i.e*. transmitted text shown in the Rx window. Whether transmitted text is echoed or not can be enabled/disabled using **View ⇨ RX Window Options**.
- **Rx window control codes color**: control codes are enabled/disabled with View ⇨ Show Control Codes.
- **Rx window normal|paused background**: you can change the background color when the RX window is paused (frozen) with the <**Insert**> key.
- **Rx window mouseover highlight**: the text changes color when you move the mouse cursor across it. This is useful when you are selecting text to right-click for the comment field of Logger32, or when selecting a callsign or RST to save into the log.

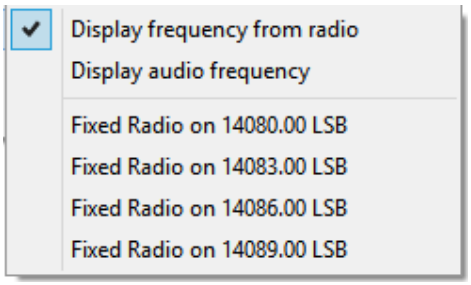
Copyright © 2026 Robert C. Furzer K4CY

- **Rx window bold callsign text|back color**: callsigns can be made to really stand out from the rest of the decoded text by the judicious choice of colors.
- **Tx window font**: sets print style and color of transmitted text in the TX pane.
- **Tx window background**: let us make the TX pane more distinctive.
- **Tx window buffer indicator color**: colors the moving bar showing text that has been transmitted so far.

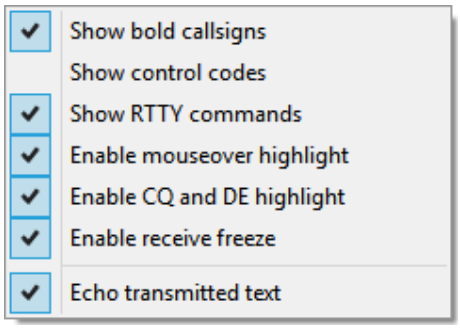
21.2.3 View menu ▶



- **Show Statusbar**: shows or hides the status bar across the bottom of the Sound card data window – see below.
- **Show RTTY toolbar** [MMTTY only]: the toolbar is a row of icons to toggle MMTTY functions.
- **Show RTTY XY scope** [MMTTY only]: displays the RTTY tuning scope.
- **Show Radio debug window**: displays the CAT communications to help you fault-find any related issues. See radio debug window.
- **Frequency display**: the submenu offers essentially 3 options for the numbers on the frequency scale above the waterfall/spectrum ▶



  - **Display frequency from radio**: is the obvious choice if you have a CAT-connected radio.
  - **Display audio frequency**: ignores the RF frequency, showing just the audio tones being received and generated by the sound card.
  - **Fixed Radio on …**: uses one of the 4 pre-defined RF frequencies defined under **Settings** ⇨ **Preset Radio frequencies**, on the basis that you will manually tune your non-CAT radio's VFO to the same setting, hence the display shows the radio frequencies.
- **Macro buttons**: choose to display 0, 1, 2, 3 or 4 rows of 12 programmable macro buttons.
- **Rx window options**: choose how to display and format information in the RX pane ▶



  - **Show bold callsigns**: the software is pretty good at picking out amateur callsigns in the decoded text, making it easier for you to notice and if wanted click/double-click them.
  - **Show control codes**: displays <CR> for new line carriage return codes received *etc*., as well as interpreting them.
  - **Show RTTY commands**: displays RTTY control commands that you issue from the toolbar or macros.
  - **Enable mouseover highlight**: shows the selection as you click-hold and mouseover the text, with the background color defined under **Settings** ⇨ **Appearance**. When you release the button, you are invited to choose what log entry fields (if any) to populate with the highlighted text. Otherwise, it stays in the Windows paste buffer until overwritten.

Copyright © 2026 Robert C. Furzer K4CY

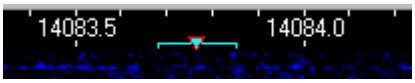
- **Enable CQ and DE highlight**: highlights “CQ” and “DE” in the decodes since CQs are generally the most interesting messages, and “DE” tells you that the sender’s callsign follows.
- **Enable receive freeze**: with the mouse cursor in the RX pane, rolling the mouse wheel can temporarily buffer incoming decodes while you scroll through previous decodes.
- **Echo transmitted text**: useful to show your outbound message text in red in the context of the received messages. It is also shown in isolation on the TX pane.

21.2.4 Tuning displays



◀ From the <**Display**> menu, two formats are available:

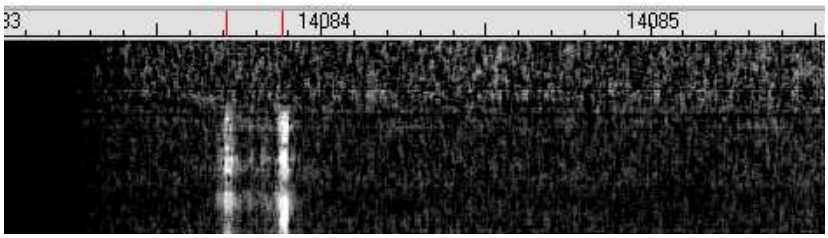
- **Spectrum**: this is a real-time instantaneous representation of the audio signal amplitude versus frequency.
- **Waterfall**: at the point of sampling, audio signals ranging from about 300 to 3000 Hz (depending on the audio bandwidth of your receiver) modulate the brightness of a horizontal line on the screen according to their amplitude. When next sampled less than a second later, the horizontal lines are shifted down to make way for a new line at the top, giving an effect similar to a s l o w waterfall showing signals across the receiver range for about the last half minute or so.



◀ In MMVARI, there is a horizontal cyan bar connecting the mark and space RTTY markers (separated by the shift value) and a red-outlined triangle showing the notional center point.

There is also a magenta triangle and subtle blue bar for your transmit frequency, although with Net **On** this is normally hidden beneath the receive markers ▶





◀ Clicking the waterfall or spectrum display near[349] the middle of the RTTY “tramlines” places the decoder there, its mark and space filters spaced equally either side.

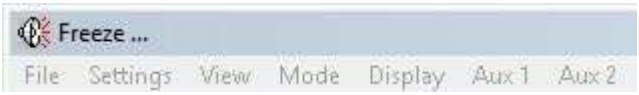
With MMTTY’s monochrome waterfall seen here, the space and mark markers are those thin red lines on the frequency scale.

[349] ‘Near the middle’ is usually close enough: if enabled, AFC fine-tunes the decoder to align with the signal.

Copyright © 2026 Robert C. Furzer K4CY

21.2.5 RX (receive) pane

If the RX pane fills with decoded text so some has scrolled out of sight, you can pause updates by placing the mouse cursor anywhere in the RX window and rotating the scroll wheel, with **View ⇨ Rx Window options ⇨ Enable receive freeze** selected. This buffers any incoming decodes without updating the display while you scroll to read the existing text. "Freeze…" appears in the ◀ caption as a clue about why the inbound decodes have stopped flowing. The display thaws out and resumes updating automatically after a while if no further scrolling occurs. If the cursor is moved away from the receive pane, freeze mode is immediately canceled and normal updating is resumed.



21.2.6 TX (transmit) pane

The TX pane shows text that is due to be transmitted, is currently being transmitted, or has already been sent. You can 'type ahead' here, even while still receiving, preparing the text to send when it is your turn to transmit. The text is transmitted in the order it is displayed in the window. This is also true for macro buttons: if you have text in the buffer when you click a macro button or press the appropriate function key, most macros take effect *after* the text already in the buffer when you clicked the macro button has been sent. The <**Clear TX**> and <**Abort**> functions however operate immediately, for obvious reasons.

Font and color options for the TX pane are in the MMVARI **Settings ⇨ Appearance** menu (see above).

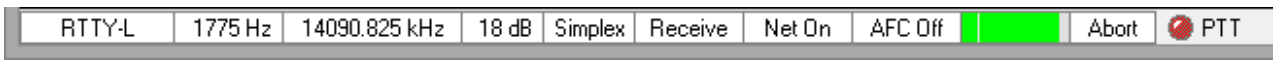
The TX pane has a scrollbar on the right-hand side when the pane fills with text. It fills from top to bottom. When the TX pane is full, text scrolls off the top, out of view unless you scroll back down using the scroll bar slider.

Transmitted text also appears in the RX pane as it is transmitted. This lets you monitor both the other station's transmission and your own, in context. You can change the color of the transmitted text by using the MMVARI menu **Settings ⇨ Appearance ⇨ Rx window ⇨ Tx text color** making it easier to distinguish between received and transmitted text.

When the TX pane is empty, if you are still transmitting, an idle signal is transmitted. If you are asking another station for an IMD report, send a few seconds of PSK idle.

21.3 Sound card data window status bar

The status bar at the bottom of the Sound card data window is the command and control center, providing access to information and commands that you will use actively during digital operations.

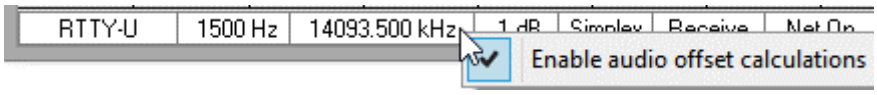


Fields in the status bar not only give information, but those with captions in blue or red are controls that act as switches to change modes or perform actions (for example, to switch between receive and transmit).

- **Mode**: the first panel indicates the digital mode (datacommunications protocol) currently selected. Clicking this panel cycles through the available modes. Right-clicking steps back through the cycle. Mode can also be set directly from <**Mode**> on the Sound card data window menu.

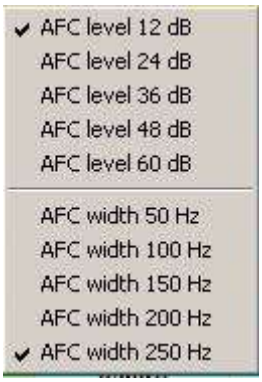
Copyright © 2026 Robert C. Furzer K4CY

- **Audio frequency:** the second panel indicates the audio frequency at the tuned position of the main window.
- **Operating frequency**: the third panel shows your actual operating frequency *i.e*. the radio frequency plus (USB) or minus (LSB) the receive audio frequency, assuming you have a CAT-connected radio. From the Sound card data window menu, select **View ⇨ Display frequency from radio.** Logger32 may or may not reflect the RIT or clarifier on the radio according to whether the radio reports it through CAT. To display the VFO dial frequency on the scale *without* applying the audio offset, right-click the radio frequency panel then click to deselect (untick) <**Enable audio offset calculations**> ▶

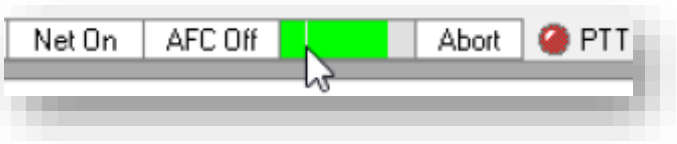


- **IMD**: the fourth panel on the status bar has two different uses, depending on which digital engine is being used.
  - In MMVARI, the fourth panel indicates the relative Signal to Noise level of the signal being decoded. A good quality signal is (by definition) a combination of factors such as: quality and linearity of the original generated and transmitted signal; propagation path attenuation, distortion, fading *etc*.; natural and man-made interference; receiver distortion, noise, linearity *etc*. Signal strength is only *part* of the equation.
  - In MMTTY, the fourth panel shows the RTTY frequency-shift being used.
- **Simplex/Split**: clicking the fifth panel shows simplex or split operation. With a CAT-connected radio, Logger32 can operate split frequency by moving the VFO as the radio goes between receive and transmit. The panel tells you if you are in simplex or split mode, and allows you to set the direction and amount of transmit offset relative to your receive frequency.
- **Receive/Transmit**: clicking the sixth panel toggles the transceiver between receive and transmit. Right-click this panel to activate the "Tune" function. See Sound Card Transmitter Audio Setup for detailed advice.
- **Net On/Off**: clicking the seventh panel enables and disables the Net function.
- **AFC On/Off**: another toggle, the eight panel enables or disables Automatic Frequency Control. PSK signals are quite difficult to tune-in manually. You will usually want the AFC on, at least to tune-in a station.

  You can adjust the MMVARI AFC capture level and width for both the Sound card data window and the SO2R/SO2V MMVARI Window. Right-click the AFC pane on the status bar of the window ▶



- **Signal strength and squelch**: the green[350] bar in the ninth panel indicates signal levels, with a thin yellow line indicating the squelch threshold ▶



  To change the squelch setting, position the cursor on the yellow line, click and drag the line left or right. The program will only attempt to decode signals above the squelch threshold, ignoring weaker signals. The signal strength

[350] These colors are configurable. Your squelch panel may look different.

Copyright © 2026 Robert C. Furzer K4CY

rises from left to right, so placing the squelch line at the left-edge provides maximum sensitivity with no squelch action … and a tendency to print gibberish caused by band noise. With the indicator near the middle, strong signals to the right of the line will be decoded but weaker signals to the left will not. Put the line too far to the right and nothing breaks the squelch.

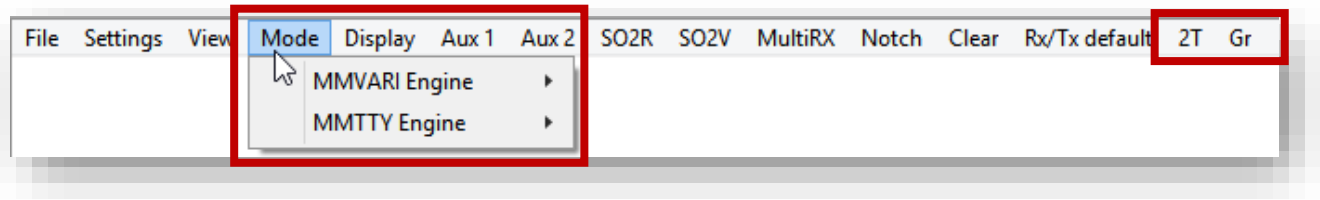
- **Abort**: clears the transmit buffers. If transmitting, the radio reverts to receive almost instantly.
- **PTT**: a red "LED" indicates that the PTT is released (floating: the radio is receiving). It turns green when PTT is grounded (asserted: you are transmitting).

21.4 Digital communications modes

Both MMTTY and MMVARI support conventional ham **R**adio **T**ele**TY**pe at 45.45 baud with 170 Hz shift. Occasionally you may find 50 baud RTTY but that and other variants are rare on the amateur bands. Commercial RTTY stations use a variety of speeds and shifts. RTTY uses ***F**requency*-**S**hift **K**eying.

MMVARI also supports ***P**hase*-**S**hift **K**eying plus other modulation types and speeds:

- BPSK – **B**inary **PSK** uses 2 phases. The most common form, [B]PSK31, operates at 31.25 bauds (equivalent to about 50 WPM), with faster variants PSK63 and PSK125 used occasionally, although all are now overshadowed by the slower but more reliable **F**ranke **T**aylor modes FT#.
- MFSK – **M**ultiple **F**requency-**S**hift **K**eying with between 4 and 64 tones.
- QPSK – **Q**uadrature **PSK** uses 4 phases.
- MBCS – MMVARI's modes listed with "(MBCS)" are **M**ulti-**B**yte **C**haracter **S**et implementations for complex languages such as Japanese with far more than the minimalist 26 letter English alphabet.



21.5 Selecting the soft modem

Two sound card digital mode engines (soft[ware] modems) are integrated into Logger32 and can be opened from the Sound card data window's <**Mode**> menu. Two more can be installed separately and invoked from Logger32:

- **MMTTY**, written by Makoto Mori JE3HHT, supports RTTY with various baud rates, shift widths, and mark/space selections.
- **MMVARI**, also by JE3HHT, offers PSK, MFSK, QFSK *and* plain 50 baud 170 Hz shift ham RTTY.
- **2Tone** by David Wicks G3YYD only supports RTTY but is capable of decoding weaker signals than MMTTY. Download the latest version from here. Un-zip the files into your *C:\Logger32* folder. Click <**2T**> on the Sound card data window menu to run **2T**one.

Copyright © 2026 Robert C. Furzer K4CY

- **GRITTY** by Alex Shovkoplyas VE3NEA is another RTTY-only engine that is more sensitive than MMTTY. Download it from here. Install it by double-clicking *Setup.exe* in the downloaded file. Click <**Gr**> to run **GR**ITTY.

21.6 MMTTY

21.6.1 MMTTY toolbar



Most of the MMTTY toolbar buttons are toggles. Look closely for the subtle button shadows to see whether the buttons are depressed[351] or not.

- **Squelch**: enable or disable squelch that only attempts to decode signals above a strength threshold set on the status bar signal strength/squelch pane.
- **BPF**: enable or disable an audio **B**and-**P**ass **F**ilter.
- **Normal/rev**: try switching to **rev**erse if you are only receiving gibberish from a reasonably strong station. If that doesn't help, toggle back to Normal and check that you have tuned the station properly.
- **XY scope**: show the RTTY tuning aide.
- **UOS**: **U**nshift **O**n **S**pace automatically selects the letters character set following receipt of a space character, regardless of whether the transmitting station sent a letters-shift (on the presumption that it may have been transmitted but not received *e.g.* due to QRM or QSB).
- **Notch**: enable or disable a narrow audio notch filter.
- **Figures**: change to the numbers character set (until the next letters-shift is received, or the letters-shift icon is clicked, or a space is received with UOS).
- **Letters**: change to the letters character set (until the next numbers-shift is received, or the numbers-shift icon is clicked).
- **Settings**: a submenu of configuration options.
- **Net**: enable or disable the Net function that generates the same frequency tones for transmission as those to which the decoder is currently tuned.
- **AFC**: enable or disable Automatic Frequency Control that fine-tunes the decoder frequency to align with the signal currently being decoded, or if none a signal nearby.
- **HAM**: adopt the usual amateur defaults for RTTY *i.e.* 45.45 baud, 170 Hz shift …
- **Reset**: revert to the currently-selected profile settings, negating any settings changes made since the profile was last saved.
- **Abort!**: stop transmitting immediately, and clear the TX buffer of any text due to be sent.
- **Profile**: choose one of the stored MMTTY profiles (sets of parameters).

[351] As in pushed in, not down in the dumps. <**XY scope**> and <**Net**> are quite happily depressed in the screenshot shown.

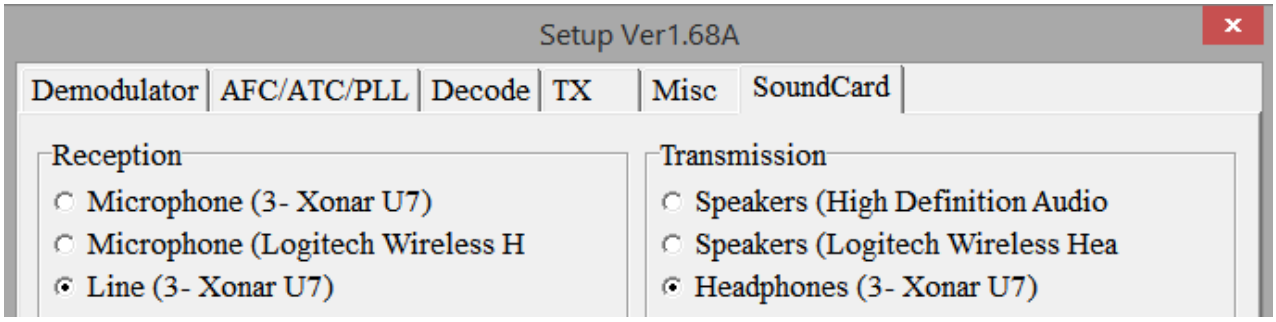
Copyright © 2026 Robert C. Furzer K4CY

21.6.2 MMTTY sound card selection

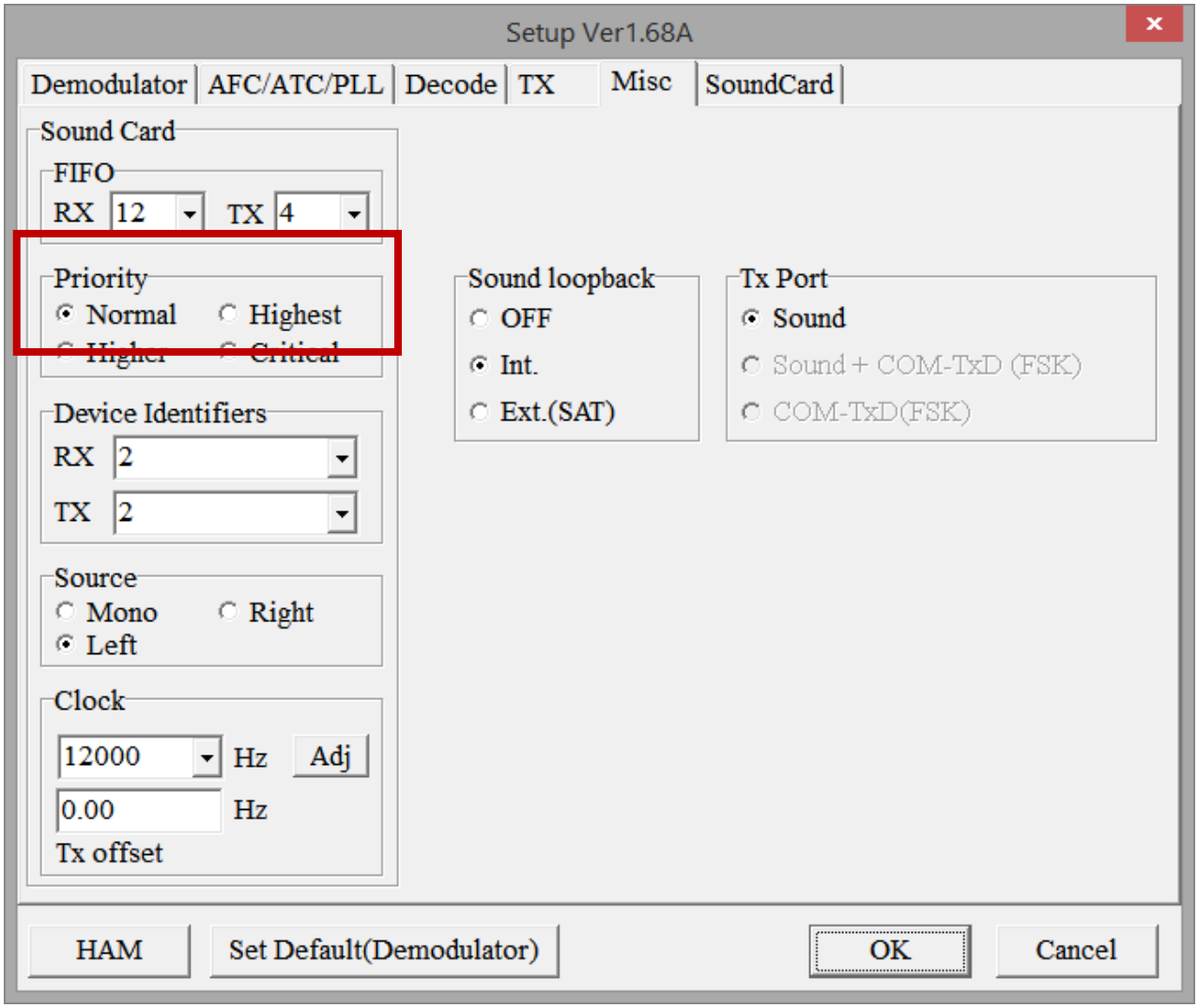
Start MMTTY through the Sound card data window's <**Mode**> menu, then click the wrench icon on the toolbar ▶



Click the <**SoundCard**> tab and under Source select the sound cards/devices connected to – or perhaps embedded within – your radio ▼



Then click the <**Misc**> tab and select the mono/stereo audio input channel to be used ▼

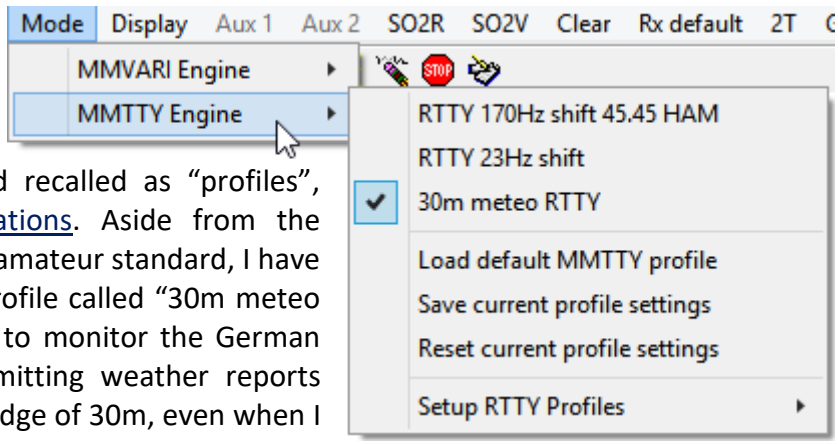


Click <**OK**> at the bottom of the setup window to make the changes … and then **save your settings** for the next time using **Mode ⇨ MMTTY Engine ⇨ Save current profile settings**.

Copyright © 2026 Robert C. Furzer K4CY

If you have an SO2R or SO2V setup, from the Sound card data window SO2R or SO2V menus open a second instance of MMTTY. Repeat the above sound card setup process there as well.

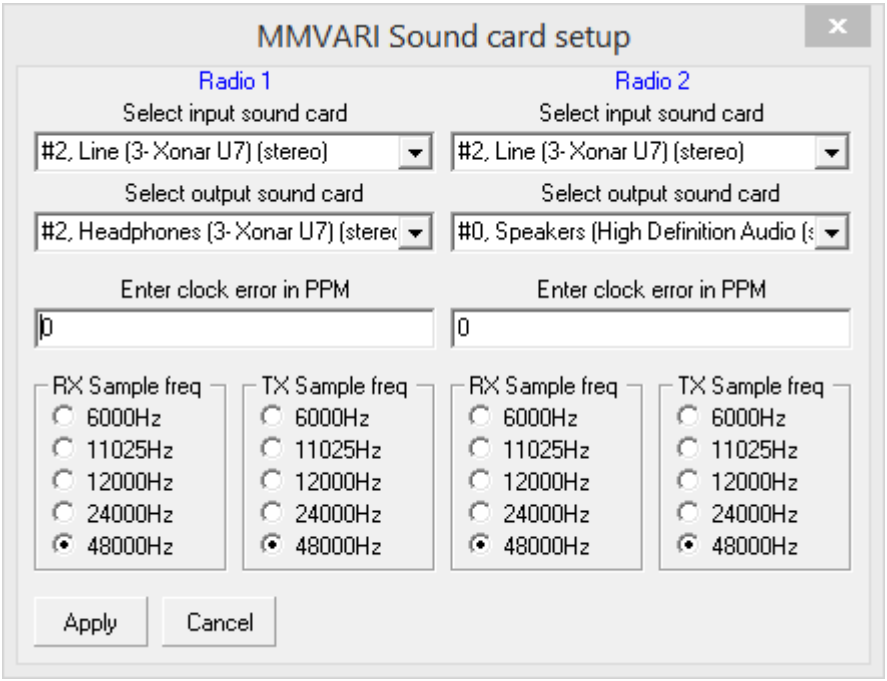
21.6.3 MMTTY profiles



Although MMTTY only supports RTTY, it can be configured for various shifts and speeds … and those setups can be saved and recalled as "profiles", similar to Logger32's configurations. Aside from the conventional 45.45 baud 170 Hz amateur standard, I have created and saved an MMTTY profile called "30m meteo RTTY" for 50 baud 425 Hz shift to monitor the German commercial RTTY station transmitting weather reports around the clock at the bottom edge of 30m, even when I can find no ham RTTY.

21.7 MMVARI

21.7.1 MMVARI sound card selection



Open MMVARI through the Sound card data window's <**Mode**> menu.

Open **Settings ⇨ MMVARI settings ⇨ Sound Card setup**.

◀ Select your sound card/s for input and output of audio signals.

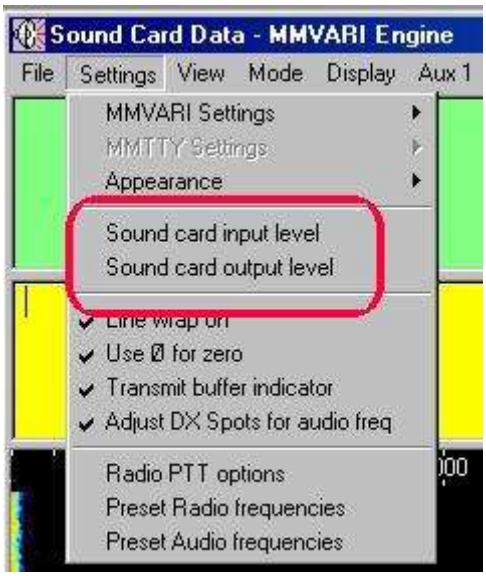
◀ Select the RX and TX sample rates, then click <**Apply**>.

Now open **Settings ⇨ MMVARI settings ⇨ SO2R/SO2V audio channel** and select the audio channel for Radio #1. If you are already set up for SO2R or SO2V, the other audio channel will be automatically selected when the Radio #2 or sub-receiver is active. If you setup different sound cards for each radio, the system will select the appropriate sound card when you switch radios.

Copyright © 2026 Robert C. Furzer K4CY

The MMVARI selections are automatically saved when you exit the respective setup child window.

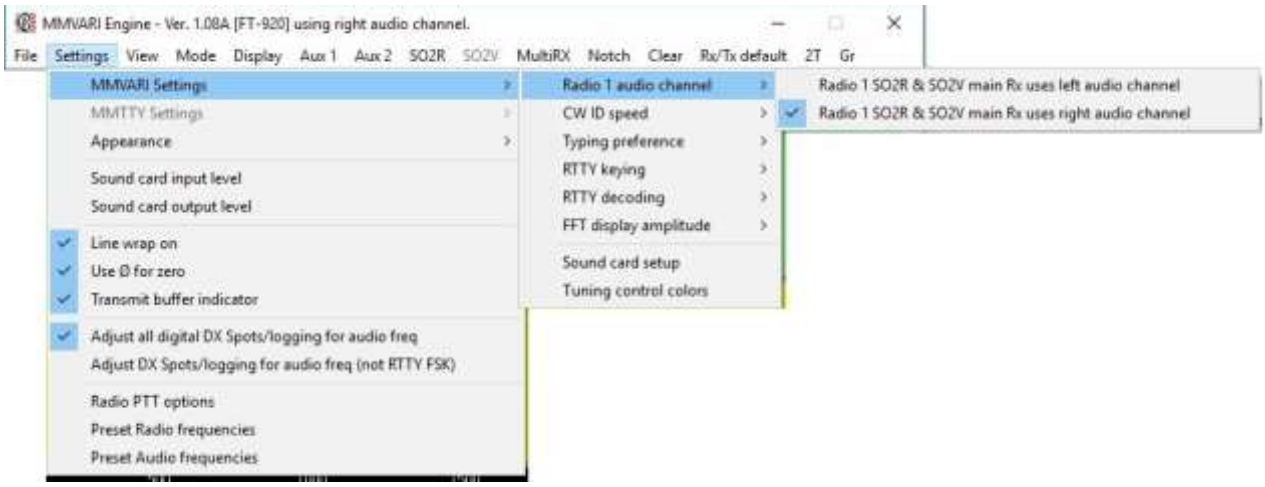
Finally, set the volume controls in either MMVARI or MMTTY.



From the Sound card data window menu, open **Settings ⇨ Sound card input level** to open the appropriate volume controls ▶

To help avoid distortion caused by over-modulation, the 50% level is a sensible starting point.

While operating MMVARI, find out which sound card is currently in use by mousing-over the audio box in the lower status line. The selected sound card is shown in the tooltip in either Receive or Transmit modes.



The Sound card data window enables:

- PSK and RTTY communication with minimal hardware.
- One-click entry of callsigns for use in logging and in macros.
- Display of frequency information (requires a CAT-connected radio).
- Automatic entry of data from the received data screen to Logger32.

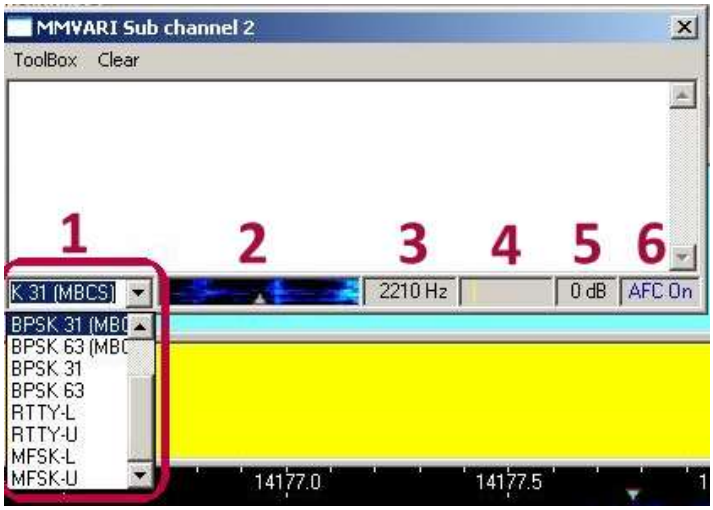
There are several ways of using Logger32 and the Sound card data window. The choice will determine what features are available:

- Communication may be set up between a CAT-capable radio and the computer. This lets Logger32 control the radio, display the radio frequency, plus a number of special functions.
- For RTTY mode, it is possible to operate AFSK or FSK. FSK has certain technical advantages, but some advanced operating features are only available using AFSK.

Most program settings made from the Sound card data window menu remain the same in all modes. However, this is not true for the macro buttons: the buttons are the same for PSK31, PSK63 and PSK125 modes, but you can have an entirely different set of buttons for RTTY.

Copyright © 2026 Robert C. Furzer K4CY

21.7.2 MMVARI AUX



The AUX window's status bar has six ▶ horizontal panels:

- Panel 1 shows the current mode, with a pull-down mode selector
- Panel 2 is a miniature spectrum display.
- Panel 3 reveals the audio tone offset in **H**ert**z**.
- Panel 4 is an indicator showing signal strength and squelch level.
- Panel 5 gives the signal to noise ratio in dB.
- Panel 6 shows MMVARI's **A**utomatic **F**requency **C**ontrol status.

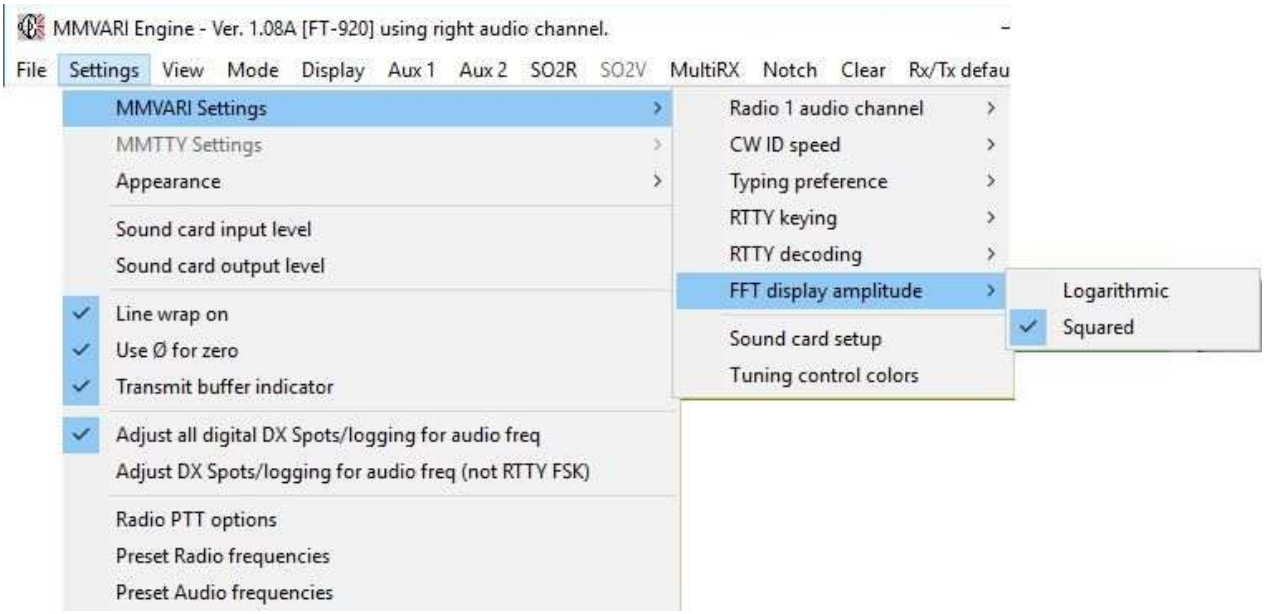
To close the AUX or Sub channel, click the X in the upper right corner or use <**Alt**+**F4**> with the focus on the window you wish to close.

21.7.3 MMVARI settings menu

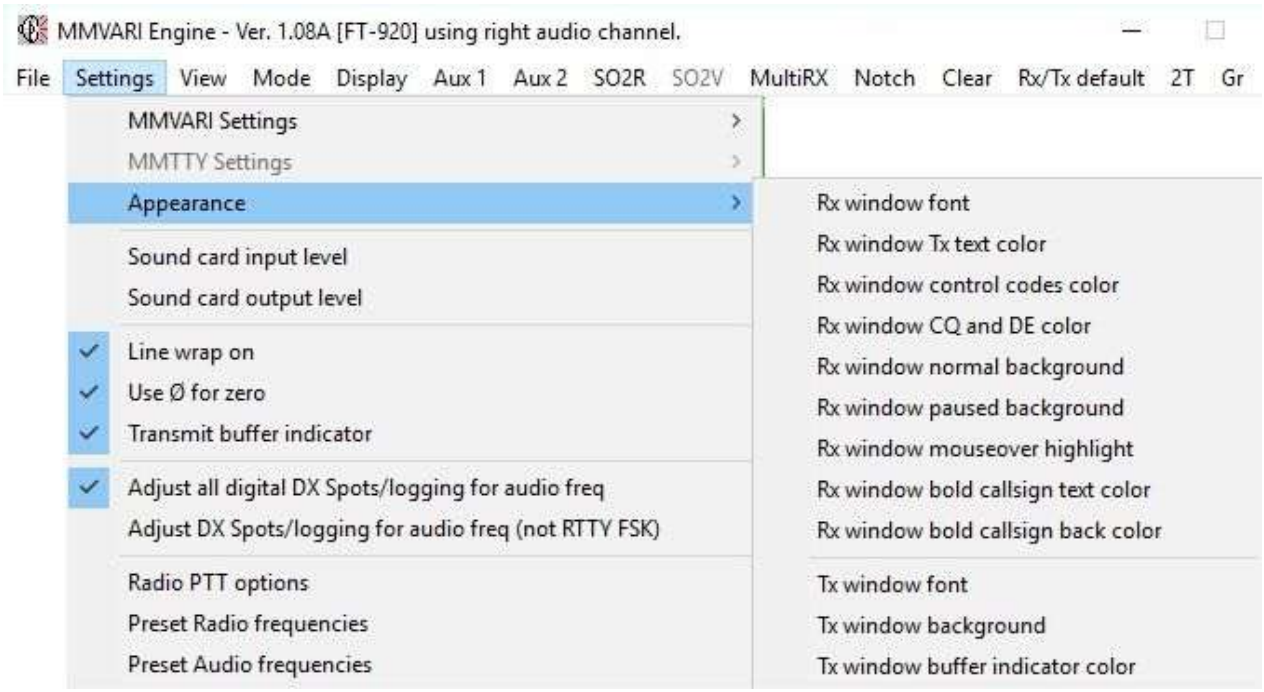


Copyright © 2026 Robert C. Furzer K4CY

Here are your options for the MMVARI window ▼



These are the settings listed under **Settings ⇨ Appearance** ▼



- **Rx window font**: sets print style and text color.
- **Rx window Tx text color**: sets color for echoed text. Echoed text is enabled/disabled under **View ⇨ RX Window Options**.
- Rx window control codes color: control codes are enabled/disabled with **View ⇨ Show Control Codes**.
- **Rx window CQ and DE color**: picks out those message elements in the chosen color.
- **Rx window normal background**: sets the default background color.
- **Rx window paused background**: sets the background color when all RX windows are paused using the <**Insert**> key.

Copyright © 2026 Robert C. Furzer K4CY

- **Rx window mouseover highlight**: sets color for text change when the cursor is over it. This is useful when you are selecting text to right-click for the comment field of Logger32, or when selecting a callsign or RST for a log.
- **Rx window bold callsign text color**: sets the text color for callsigns which are automatically shown in bold.
- **Rx window bold callsign back color**: colors the background behind callsigns.
- **Tx window font**: sets print style and text color.
- **Tx window background**: sets the background color for the transmit window.
- **Tx window buffer indicator color**: change the color showing transmitted text.

21.7.4 Reading a fast-filling RX window

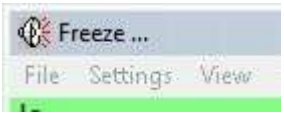
The main receive window is used to display received PSK31 text, echo transmitted text as it is sent, and provide an area for you to grad details such as callsigns, signal reports and names. If it fills too fast for you to read …

- **RX window pause**: the RX window sports a scrollbar on the right-hand side when the window fills with text. Text scrolls from the top of the window downward until the window is full, at which point text scrolls off the top, out of view. See this text by:
  1. Press the <**Insert**> key on the keyboard (once). The RX window background goes white[352] (or whatever color you selected in **Settings** ⇨ **Appearance**) and the displayed text is frozen (incoming text continues being decoded and buffered).
  2. Click the scroll bar or drag the slider to scroll the buffered text through the RX window.
  3. Press <**Insert**> again to resume the display updates.

  This technique works during transmit, as well as in receive. You can freeze and scroll through the RX window while the text you have already entered is being transmitted by Logger32.

  Logger32 continues to receive data while the RX window is paused, and the text that arrived while the display was paused appears at the bottom of the RX window when you cancel the pause.
- **Mouse scrolling**: if the RX window is so full of text that some has scrolled out of sight, or if it is updating too quickly for your tired eyes to read the text, you can pause the RX window by placing the mouse cursor anywhere in the RX window and turning the scroll wheel to buffer incoming data, letting you scroll the existing text without interruption by display updates.

  "Freeze…" appears on the caption ▶



[352] When paused, if the background turns the same color as the text, you won't be able to read the letters. Bear this in mind when choosing the text and background colors.

Copyright © 2026 Robert C. Furzer K4CY

Enable “Freeze” by opening the MMVARI **View ⇨ Rx window options** and selecting **<Enable receive freeze>** ▶



Decoding continues as normal in the background and is buffered while you review the received text, uninterrupted by the display of newly-decoded content.

The freeze times-out and thaws (self-cancels) if no further scrolling is detected for a while. If the cursor is moved away from the receive pane, freeze mode is immediately canceled. Either way, any text that was decoded in the background now appears, followed by any further decodes as they arrive.

21.7.5 RX frequency controls

Here are three ways to control your receive frequency:

1. Click somewhere in the Logger32 receive range.
2. Use the $**QSY**$ macro.
3. Use the $**Up**$ or $**Down**$ macros.

The actual transmit/receive frequency is your USB radio’s VFO dial frequency *plus* the audio frequency displayed in MMVARI/MMTTY. On LSB, *subtract* the audio frequency from your radio’s dial frequency.

Logger32 shows the RIT offset for radios that report it via CAT.

21.7.6 Using receiver filters

Logger32 uses audio DSP to select one signal from many that are in the receive range. If you set your receiver selectivity to wide, you can still copy a single signal, as long as no signal is very strong. However, DSP in Logger32 works only on the audio coming into the sound card so if the transceiver receive passband is wide, noise and interference come through too. If some of the interfering signals are very strong, they may affect the receiver AGC, which reduces the amplitude of a weaker signal that you are trying to copy.

Audio DSP in your radio is of little value, especially the early versions. A narrower audio signal may *sound* nicer in the speaker/headphones but copy is unlikely to improve – and in fact may be degraded due to digital artifacts – compared to MMVARI or MMTTY alone. They have more sophisticated DSP using the full power of your PC’s sound card and CPU.

IF DSP, on the other hand, may provide the filtering you need to prevent distortion from occurring in the IF stages of the receiver.

Copyright © 2026 Robert C. Furzer K4CY

Since RTTY and PSK31 are narrow-bandwidth modes, the best results in terms of removing interference and maximizing signal-to-noise ratio (S/N) are where the IF selectivity is as narrow as the signal (about 250 Hz for RTTY and about 50 Hz for PSK) . Even if you do not have narrow crystal filters, you may be able to adjust the passband using IF-shift, or move an interfering signal outside the radio's passband while leaving the desired signal within it. If you radio has DSP, IF notch filters and noise reduction may help (*provided* the DSP has been well designed, implemented and configured to minimize digital artifacts which can interfere with digital signal decoding).

One tactic is to set your radio's selectivity wide while tuning the band so you can see many signals.

Once you have found a signal and you want to make a QSO, narrow your radio bandwidth. If your radio uses **p**ass**b**and **t**uning, IF-shift, or some other passband-modification scheme, you may be able to adjust it to reduce interference. With Logger32's spectrum display, you can see the effect of making filter and other IF changes at your radio. Sometimes, all the filters are not exactly centered at the same frequency, and you may have to adjust the passband tuning control to ensure that you have the signal of interest in the passband. With the $**HexCommand**$ and $**Command**$ macros, you might define macro buttons to control your bandwidth right from the Logger32 screen.

It is also possible to use your radio's manual IF notch filter to remove interference. Again, you will see the effect of this filter visually by watching the waterfall display. If your radio has an effective manual notch filter, tune it slowly and you will see a line of black (no signal) move across the display. However, if this notch is an audio filter, the warning above about audio filtering applies. With an audio notch filter, any distortion caused by the interfering signal will already be present by the time you apply the audio notch filter in an attempt to remove it.

An automatic notch filter is unlikely to work because it will probably remove the wanted signal along with the unwanted ones if it cannot discriminate between them.

21.7.7 MultiRX

MultiRX is a multi-channel decoder capability within MMVARI. Whereas normally you are only interested in decoding the signal of somebody you are in contact with, the MultiRX window splits the received audio band into up to 24 channels in which up to 24 signals are decoded simultaneously. This can be useful in a DX pileup to figure out which station the DX is working and hence where he is listening at the moment – or if you *are* the DX, you can pick out callers from your pileup. In a contest, MultiRX helps you search and pounce on new stations.

To open it, click <**MultiRX**> on the MMVARI toolbar[353].

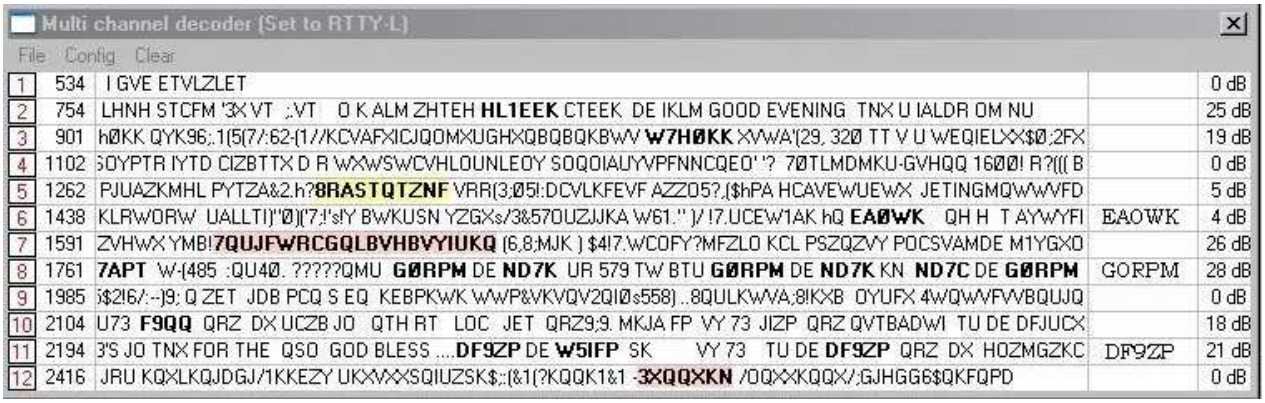
The audio spectrum is divided equally between subchannels like this:

- The lower limit is set to 500 Hz and the upper limit is set at 2500 Hz, so the available spectrum is 2000 Hz.
- This gives you 6 adjacent subchannels of 333.3 Hz each or 12 subchannels of 166.6 Hz *etc*. Given that amateur RTTY signals normally have a shift of 170 Hz, up to 11 RTTY signals could theoretically share the 2000 Hz bandwidth without overlapping … but in practice on a busy band, overlaps are common and netting is imprecise, hence the value of additional subchannels.

[353] When the MultiRX window is open, the AUX channels are disabled.

Copyright © 2026 Robert C. Furzer K4CY

Each decoder starts out in the middle of its allocated subchannel, sweeping either side to search for and perhaps lock-on to a signal within its range. Decoded text[354] appears on a row in the MultiRX pane's table ▼



There are five columns in the table:

1. **Subchannel number**: subchannels are also indicated with little markers on the MMVARI spectrum frequency scale.
2. **Audio frequency:** of the channel, in Hertz.
3. **Decoded text**: scrolls to the left once the column width is filled - a tickertape effect.
4. **Bookmarked callsigns**: callsigns can be picked out in bold in the decoded text streams. You can right-click them to save them in this column, so you know which station is active in that channel even after the decoded text containing their callsign scrolls off to the left, out of sight.
5. **Signal-to-noise level**: a measure of the strength of the signal above the noise, and hence the quality/reliability of the decoding.

The MultiRX window's caption shows the mode in use, and beneath that is a menu with three items:

- **File**: offers just the one option, <**Exit**> which does the same as clicking the corner X.
- **Config**: use this submenu to configure MultiRX – see below.
- **Clear**: wipes the MultiRX pane as if you had just closed and reopened it.

21.7.8 MultiRX config menu

- **AFC settings**: AFC level sets the signal-to-noise level required for the AFC to lock on to and attempt to decode a signal. AFC width sets the bandwidth within which the AFC sweeps for decodable signals[355] ▶



[354] MultiRX decoders share the same squelch setting as MMVARI's main decoder.

[355] You can also drag a subchannel indicator left or right on the waterfall frequency scale and drop it onto a particular signal if the AFC function seems reluctant to lock-on. If the signal drops out, AFC action resumes.

Copyright © 2026 Robert C. Furzer K4CY

- **# Channels**: click to tick the number of receive channels to display ▶

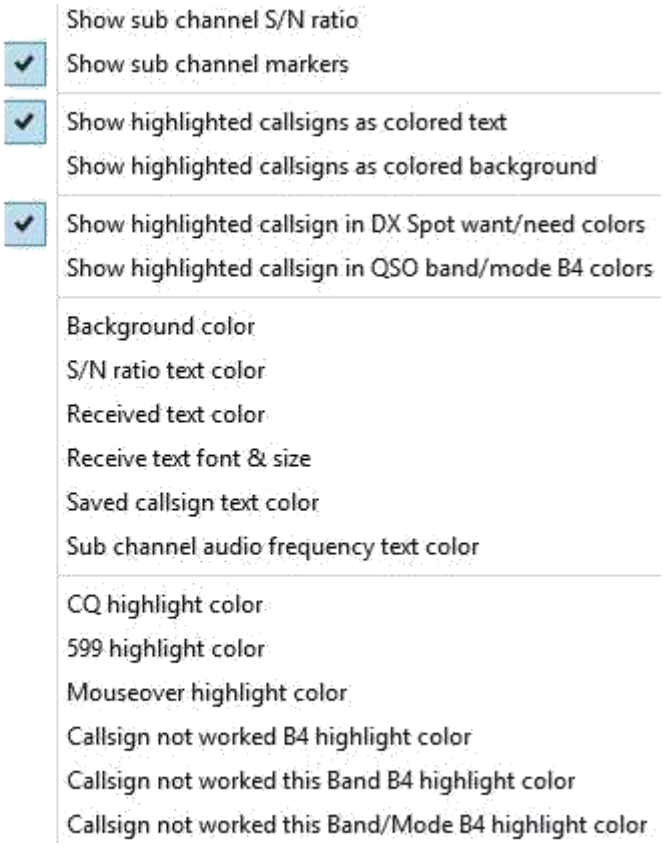


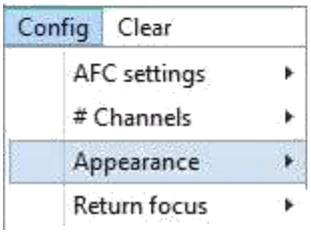
A low-specification PC *may* lack the power to decode all 24 channels simultaneously, so if it appears to be struggling and stuttering, try reducing the number accordingly.

- **Appearance**: gives you Logger32's usual range of options to customize the text, background and highlighting colors, font and font size *etc*. ▼▶

  <**Show highlighted callsign in DX spot want/need colors**> applies the same highlighting to 'new ones' as is used elsewhere in Logger32 … which is easier and less confusing than setting up custom colors just for MultiRX.

  <**Show highlighted callsign in QSO band/mode B4 colors**> applies the colors from your worked/confirmed table.





- **Return focus**: choose whether to focus (*i.e*. leave the cursor) on the log entry pane or the Sound card data window after you click things in MultiRX.

21.7.9 MultiRX operation

Right-clicking a callsign highlighted in the receive column puts the call in the righthand column for future use, bookmarking it without changing the log entry pane or Sound card data window.

Clicking a highlighted callsign in the received text, or a bookmarked callsign in the righthand column, places the callsign into the log entry pane and shifts the Sound card data window's decoder to that signal.



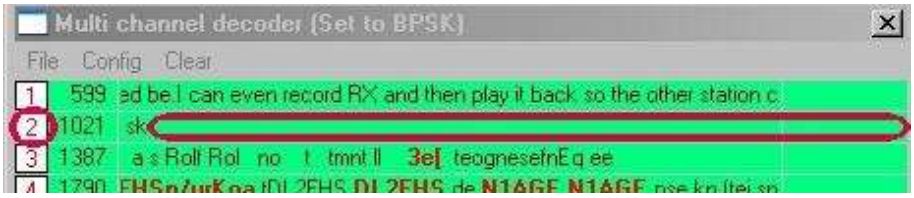
Each channel is allowed to float within its assigned frequency segment. The AFC function detects and locks-on to a signal within its range. You can also manually position a channel marker over a signal to enhance this process. Place the mouse pointer on the respective channel number in the Spectrum display and drag it over the desired signal. Release the pointer to lock on to and start decoding the selected signal. If the signal drops out, AFC springs back to life, automatically tuning within the range until it detects and locks-on to another signal.

Copyright © 2026 Robert C. Furzer K4CY

21.7.10 MultiRX clear

On the MultiRX window menu, <**Clear**> empties the receive subchannels and any bookmarked callsigns – as does QSYing the radio.

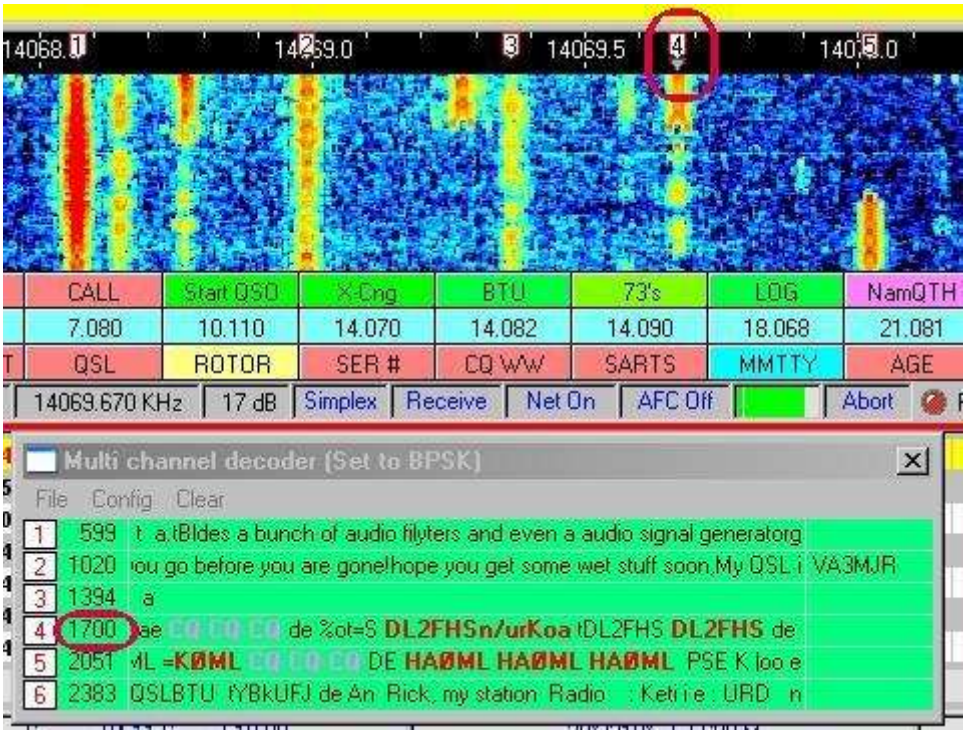
Clicking a subchannel number clears just that row ▶



21.7.11 MultiRX for split operating

When using MultiRX to bust a pileup where the DX station is working split, it is necessary to change the TX audio frequency without changing the callsign in the log entry pane's Call field.

To use the MultiRX channels exclusively, the split frequency must be 2 kHz or less. If the DX station is working a wider pileup, this feature is not helpful.



Scan the MultiRX channels for a calling station to tail-end, or select a relatively quiet transmit frequency. Click the audio frequency in the second column of MultiRX to place your main receive window and transmit frequency at the desired audio frequency. You can now transmit at the split frequency while continuing to decode the DX station in one of the MultiRX channels. When using AFSK, Net should generally be on.

21.7.12 Aux 1 and Aux 2 windows

There are three optional receive windows in MMVARI:

- **Aux 1** & **Aux 2** are auxiliary (additional) receive-only decoders, supplementing the main RX window. You can thus decode 2 further signals on the display spectrum/waterfall.
- **MultiRX** is a separate multi-channel receive-only facility described elsewhere.

Click <**Aux 1|2**> on MMVARI's toolbar to open the Aux windows, then size and position each on your screen using the mouse.

Copyright © 2026 Robert C. Furzer K4CY

The Aux windows offer:

- An additional window to receive a PSK31 signal (possibly ready for a change to QPSK).
- A way to monitor a crowded band or look for your next contact while in QSO.
- The means to work split frequency directly from Logger32 using audio split (as opposed to transceiver/VFO split).

From the Aux1 and Aux2 windows, a callsign can be captured by double-clicking it. However you can only transmit from the main TX pane, since the Aux windows are receive-only.

If you saw a DX station on Aux 1 signing P5DX, you might:

1. Quickly double-click P5DX's signal in the Aux RX window, putting his callsign in the log entry pane and filling the $**Call**$ macro.
2. Call him! Your TX frequency is unlikely to be on his frequency, so you are probably calling him split.

- By default, the Aux windows have the same font and background colors as the Main RX window, but you can change them through <**ToolBox**> on the Aux menu. <**Clear**> on the menu empties the Aux window of received text.

The two Aux windows each have:

- A caption.
- A small menu.
- An RX pane showing decoded text.
- A minimalist status bar.





To tune-in signals on Aux, <**Alt**+**click**> on the waterfall to position Aux 1 or <**Ctrl**+**click**> for Aux 2.

21.8 Using MMVARI/MMTTY with CAT

Here are various functions that are enabled with CAT:

- Display and logging of frequency from the radio.
- Frequency control of the radio using the $**Align**$ macro.
- Full split frequency operation (on any mode).
- Transmit/receive switching by CAT commands.

First, configure and get your CAT connection working as explained in the CAT chapter.

Copyright © 2026 Robert C. Furzer K4CY

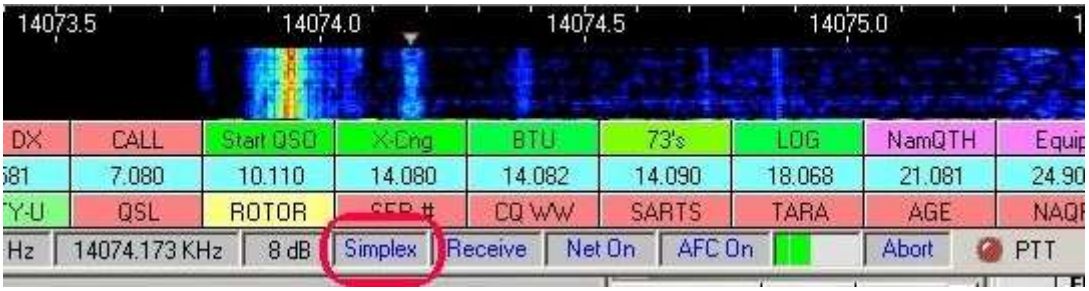
21.8.1 Operating split frequency with CAT

There are two ways that Logger32 can operate split frequency with MMVARI/MMTTY:

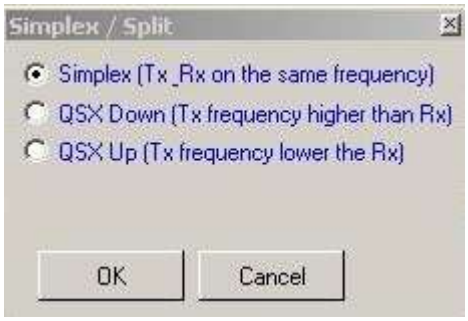
- Using different audio tones for transmit and receive. This method only works in PSK, not RTTY because (particularly with FSK) the RTTY tones are fixed.
- Changing your VFO frequency between transmit and receive using CAT.

To operate split with computer control, for PSK or RTTY:

- On the status bar in the <**Simplex**> panel, the Simplex/Split status is displayed. It probably says Simplex …



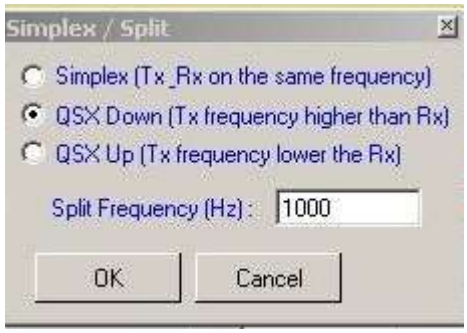
- Click the <**Simplex**> panel to open a form ▶



  Depending on where the other station is listening:
  - Click <**QSX Down**> (listen down) to transmit *above* your receive frequency
  - Click <**QSX Up**> to transmit *below* your receive frequency.

  … or stay on simplex if you want to transmit on the DX frequency, potentially causing QRM and wrath from other DXers.

- As soon as you click <**QSX Down|Up**>, a field opens for you to specify the split frequency offset *e.g.* 1000 Hz ▶



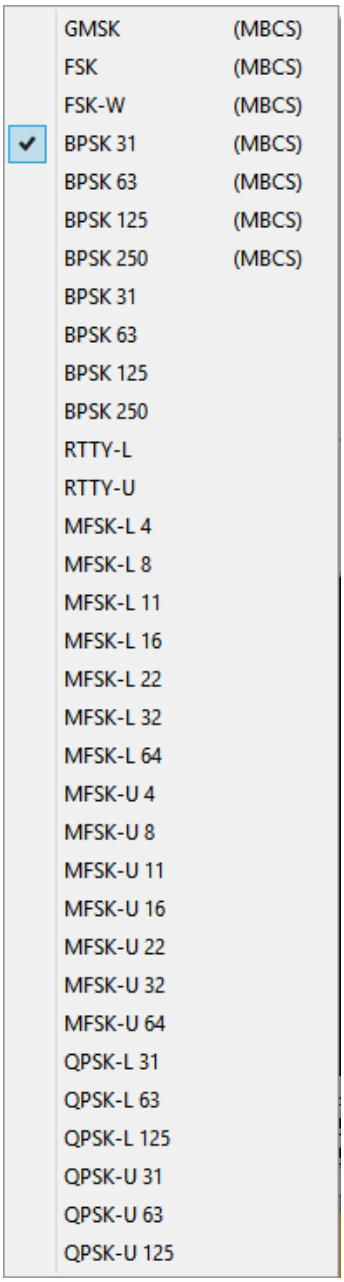
- Click <**OK**>.
- The status panel now reads “Split” instead of “Simplex”.
- Make your calls, and, we hope, work the rare DX station who is listening split!
- When you are finished, click the Simplex panel, select <**Simplex**> and <**OK**> to resume normal operation.

You can also use a macro to do this, but it is no easier unless you integrate it with some other commands. The $**QSX**$ macros is described in the macros chapter. There is a $**Simplex**$ macro as well, to undo $**QSX**$.

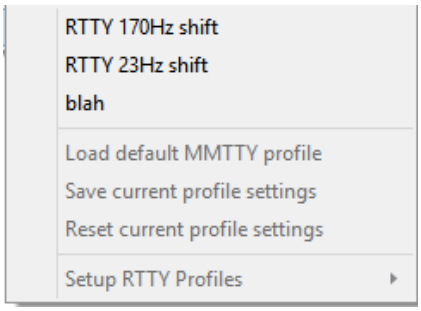
Copyright © 2026 Robert C. Furzer K4CY

21.9 Receiving RTTY and PSK signals

21.9.1 Pick a mode, any mode

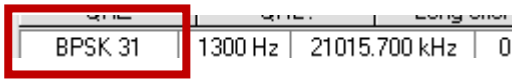


◀ MMVARI MMTTY ▶



The modes of operation (more accurately, digital communications protocols) supported by MMVARI and MMTTY are found under the Sound card data window's <**Mode**> menu.

The mode that is currently active is ticked on the <**Mode**> menu and shown on the leftmost panel of the Sound card data window's status bar ▼



21.9.2 Switching to receive

- While you are transmitting, click **Transmit** (the 6th panel on the status bar). Logger32 first finishes sending the remaining text in the TX buffer, *then* releases PTT so the radio returns to receive. The 6th panel changes from **Transmit** to **Receive** at the end of the process.
- The <**Pause/Break**> key also toggles TX ⇄ RX, the same as clicking the Transmit/Receive panel.
- Click **Abort** (9th panel on the status bar) or tap the <**Esc**> key. Logger32 clears the TX buffer and releases the PTT *immediately*. It "drops the mike".
- Run the $**Receive**$ macro *e.g.*

Copyright © 2026 Robert C. Furzer K4CY

21.9.3 Align function

There are situations in which you might want to retune your radio after you have begun to copy a station, without interrupting copy and maybe losing the station *e.g.*:

- You want to transmit a high tone to reduce the chance that you will transmit distortion, but the station you want to receive is low in your receive/transmit range, at the current radio frequency setting (see Digital signal quality).
- You have a narrow receive filter but the station you want to receive is not quite at the center or peak of the filter.
- You want to quickly move the station you are receiving nearer the edge of the receive range, so you can use your radio's fancy DSP filters (high cut or low cut) to cut down adjacent QRM.

The align function helps with both PSK and RTTY.

Provided you have a CAT-connected radio, it can be retuned with a mouse-click.

To use the align (frequency change) function:

- Click **Settings ⇨ Default main Rx Frequency** and set it to an appropriate value *e.g.* a high tone or the center of the narrow filter passband (ideally both *i.e*. use the passband tuning control to center it on the high tone frequency).
- Click **View ⇨ Frequency Display ⇨ Display Frequency from Radio**.

Here is how the align operation works:

- Leave the AFC on so your system can retune following the frequency change.
- Click a signal to receive it in the main RX window.
- Right-click the same signal to invoke the align function, so that signal will be received at the preset Default Main RX Frequency.

As well as right-clicking, you can also assign the $**Align**$ macro to a programmable button. When clicked, Logger32 moves the main passband to the default RX frequency *and* adjusts your VFO frequency such that the signal is now at the default RX frequency. Bingo![356]

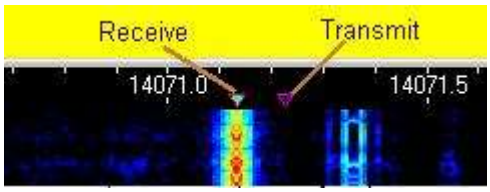
Here is an example:

- Tune your radio to 14070 kHz in USB, RTTY or DATA mode as appropriate.
- Set your default RX frequency to 2000 Hz, towards the top of your transmit filter passband such that on transmit, if you were to accidentally overmodulate on that frequency, any harmonics would be attenuated by the filter.
- Click to receive someone at 14070.5 kHz. If you were to respond on their frequency, you would normally be generating a 500 Hz tone, which is quite low and more likely to generate audio harmonics if overdriven.
- Use the align function (right-click or macro).
- Logger32 moves the VFO down to 14068 kHz, while moving the receive and transmit tones up in frequency to compensate, such that you are still receiving and transmitting at 14070.5 kHz

[356] Using this feature, it is possible to tune your radio up or down simply by clicking: click anywhere in the display to have Logger32 shift that point to the Default Main RX Frequency. If you then click to the right of (higher in frequency than) your Default Main RX Frequency, your radio tunes HF; if you click to the left (lower frequency), the radio dutifully tunes LF. Try it!

Copyright © 2026 Robert C. Furzer K4CY

… but now your transmit audio is near the top of the transmit bandpass filter, attenuating any audio harmonics if you accidentally overdrive the radio.

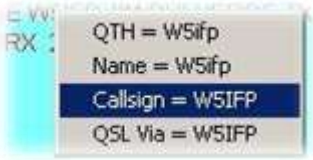


It can be difficult to right-click at precisely the right place on the received signal. The $**Align**$ macro takes care of that problem. Instead of retuning where you click, it moves the current signal that is being received to the default RX frequency. Simply dedicate a programmable button to trigger the $**Align**$ macro. When you click the button, it is as if you had right-clicked in the correct place.

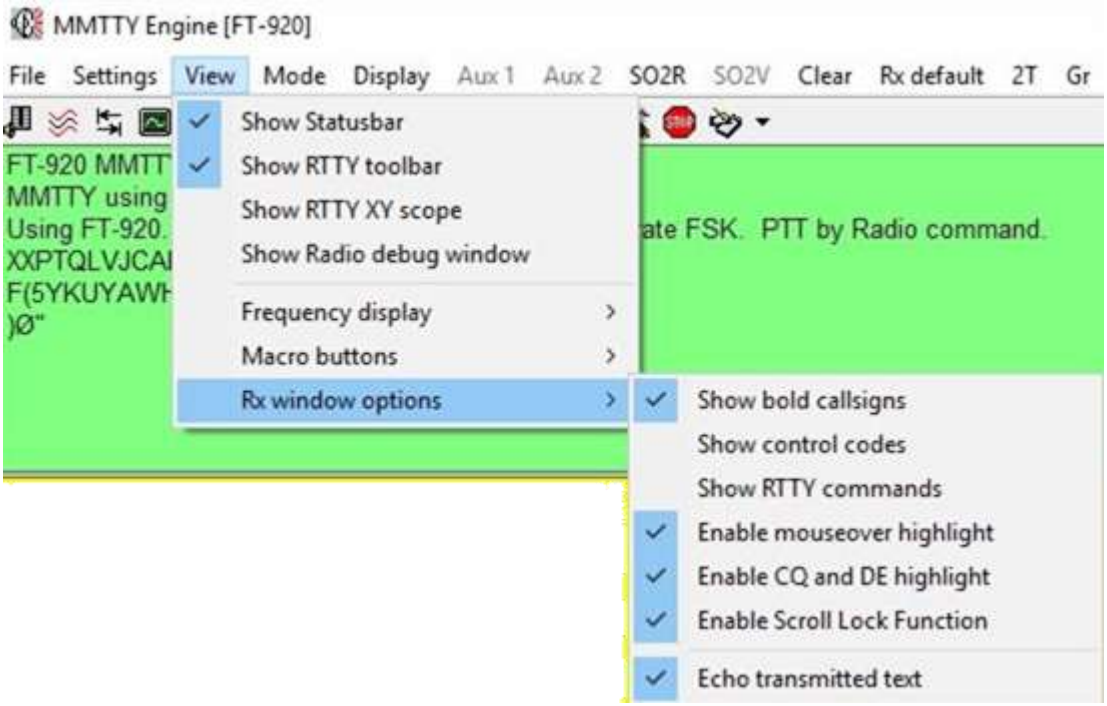
21.9.4 Logging PSK and RTTY QSOs

Logger32 can capture most of the QSO information pertaining to an RTTY or PSK QSO *directly* from the RX window. Data can be transferred to the log entry pane simply by clicking the appropriate words or numbers in the received text window.

In order for this to work, right-click the log entry pane, click <**Setup>** and then click to select each field you want to display and populate:



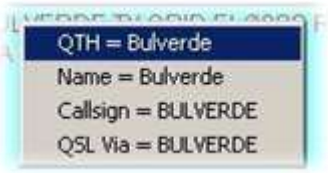
- **Callsign**: click the callsign of the station you are in contact with or desire to contact. This is a predefined field that does not require any configuration.
    - To make it easier to pick out callsigns, they can be highlighted in the received text in bold. Open the Sound card data window menu **View ⇨ Rx window options** and check (tick) <**Show bold callsigns**> ▼



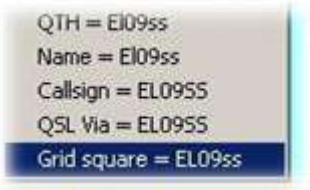
Copyright © 2026 Robert C. Furzer K4CY

- **Name**: right-click the person's name in the received text to open a little window ▶ then click <**Name**> to transfer it to the Name field in the log entry pane. This is a predefined field.

- **QTH**: right-click the person's city, region or whatever location they sent, then click <**QTH=**> to grab it. This is a predefined field.



- **Grid square**: right-click a grid square and click <**Grid square =**>. This field must be configured to match the ADIF definition of gridsquare *i.e*. a 2, 4, 6 or 8 character Maidenhead locator.



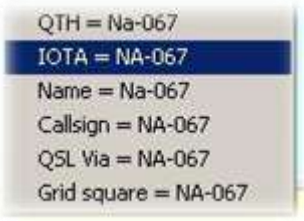
- **RST**: right-click a numeric **R**eadability-**S**ignal strength-**T**one signal report then click <**RST =**>. This is a predefined field.



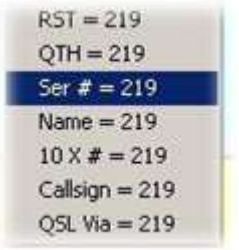
- **Ten-Ten club number**: right-click the person's Ten-Ten member number, then click <**10 X # =**>. This field must be configured to match the ADIF definition of Ten-Ten.



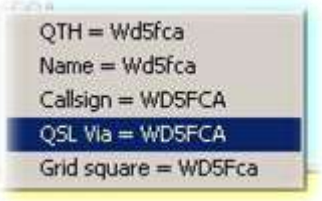
- **IOTA island identifier**: right-click the IOTA island reference then click <**IOTA =**>. This field has to be configured to match the ADIF definition of IOTA.



- **Contest exchange** (*e.g.* QSO serial number): right-click the contest exchange and click <**Ser # =**>. This field has to be configured to match the ADIF definition of SRX.



- **QSL manager**: right-click the QSL manager's callsign, then click <**QSL Via =**>. This field has to be configured to match the ADIF definition of QSL_VIA.



- **User fields**: you can also transfer data for any of the user-defined fields (USER_1, USER_2 …). These will appear in the drop-down menu listing using whatever

Copyright © 2026 Robert C. Furzer K4CY

name you have given them. For example, *provided USER_1 has already been setup in the log entry pane as the "SOTA" field*, right-click a **S**ummits **O**n **T**he **A**ir reference then click <**SOTA =>**.

21.9.5 Grabbing text with the mouse

Mousing-over the received text lets you grab content. The system is clever enough to recognize some types of data (*e.g.* callsigns and reports), giving you the option to drop the selected text into the respective fields in the log entry pane, and hence your logbook.

To capture a section of received text, position the pointer at one end of the section, click-and-drag the mouse pointer across, marking the desired text as it goes. When you release the mouse button, the highlighted text is automatically placed in the clipboard, ready to paste somewhere else by putting the cursor where you want it to go, then simply pressing <**Ctrl**+**V**> or right-clicking then clicking <**Paste**>. This is a little awkward, but gets easier with practice.

The color of the text highlighting can be adjusted using **Settings ⇨ Appearance ⇨ Receive window mouseover highlight**.

You can copy and paste text from the Sound card data window into other Logger32 windows, a browser, or a word processor via the Windows Clipboard:

- Click-and-drag the mouse across some text in the RX window, momentarily highlighting the text, then release the mouse button.
- Right-click and select <**Copy**> or <**Cut**>. Don't forget this step!
- Mouse to the window where you want to paste it, right-click and select <**Paste**>.

21.9.6 Saving received text to a file

You can save received text to a file for later reference (for instance in a contest, where you may want to double-check callsigns and reports), or to print all or part of it, using an editor such as MS Notepad, WordPad or TED Notepad:

1. Click **File ⇨ Open receive text file**.
2. Pick an existing file or name a new one to receive the saved text.
3. Receive and save the file!
4. When the activity period is completed, click **File ⇨ Close receive text file**.[357]

21.9.7 RX window pause with <Insert>

The receive window sprouts a scrollbar on the right-hand side when the window fills with text. Text scrolls down from the top of the window until the window is full. At that point, older text is scrolled off the top, out of view. You can see this text again by tapping the <**Insert**> key, once. The RX window background changes to white (or whatever color you have selected in the <**Toolbox**> menu) and the text is frozen. Click the scrollbar to scroll the buffered text through the RX window. Tap <**Insert**> again to start receiving text: the buffered RX text flows through rapidly.

[357] There is no indication that the text is being saved to file, except as viewed in the <**File**> menu. If you neglect to **Close receive text file**, you will continue saving the received text until you close the Sound card data window, close Logger32, shut down the PC … or run out of disk space!

Copyright © 2026 Robert C. Furzer K4CY

This technique works during transmit as well as receive. You can even freeze and scroll the RX window while the text you have previously entered is being transmitted by Logger32.

21.10 Transmitting RTTY and PSK signals

21.10.1 Transmitting with MMVARI

The receive and transmit frequencies are identified on the waterfall with a red diamond filled with light blue (receive) and dark blue (transmit).

The $**Align**$ macro works with Net **On** or **Off**. When you execute $**Align**$ or right-click the waterfall, both receive and transmit frequencies change to the default audio frequency. Your CAT-connected radio QSYs to the new frequency.

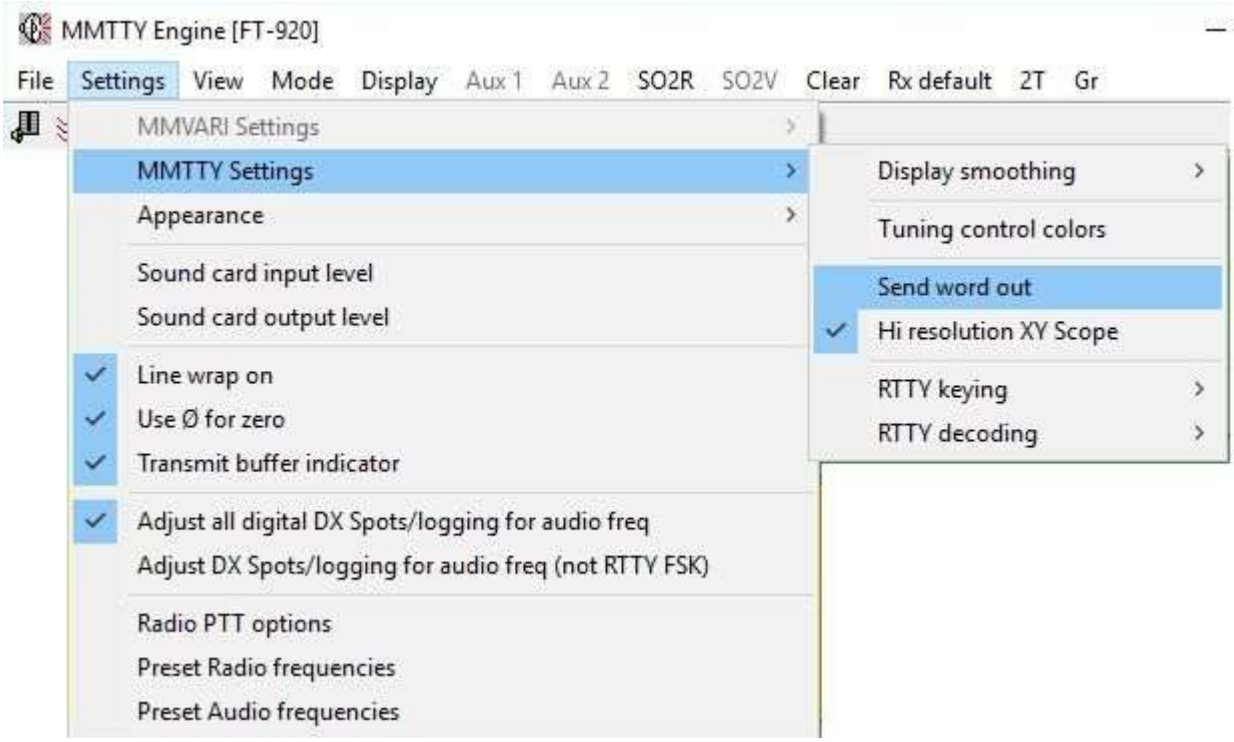
- **Transmitted characters**: RTTY has its own limited alphabet, the Baudot code, which differs from, say, Varicode used for PSK31 or ASCII. Transmitted letters are always in UPPERCASE, and there is a limited character set *i.e.*:

  A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 1 2 3 4 5 6 7 8 9 0 . , " ( ) / ? & - ! :

  If you type invalid characters that cannot be transmitted (*e.g.* $, @, #), they do not change color like the surrounding text during transmission since they cannot be sent[358].
- **Backspacing**: using RTTY, you can backspace to correct typing errors but only *before* the character has been transmitted. There is no Baudot code for backspace, so once a letter is sent, there is no way to tell the receiving station to back up and erase that character. The easiest way to benefit from the backspace feature in RTTY mode is to set Settings, Typing preference to word out …
- **Word Out vs. Character Out**: Logger32 in RTTY mode allows the user to choose either word out or character out. Click **Settings ⇨ Typing Preference ⇨ RTTY- Send word out** to select it (transmit the whole word when you press the spacebar or send a macro containing a space), or de-select it to transmit each individual character as soon as possible.

[358] Email addresses can't be sent directly via RTTY, but *name AT domain.com* would be fine. Email wasn't around when Baudot and RTTY were invented, *long* before Bill Gates was born.

Copyright © 2026 Robert C. Furzer K4CY



- **U**nshift **O**n **S**pace: elsewhere we discuss setting UOS for receive. You can also use UOS on transmit. The default for RTTY is to transmit like this: "[FIGS]599 599 [LTRS]GOOD SIG". However, if the other person has UOS set, that would print as "599 TOO GOOD SIG". Using UOS for transmit results in you transmitting and extra figures-shift character "[FIGS]599 **[FIGS]**599 [LTRS]GOOD SIG", resulting in "599 599 GOOD SIG" at the far end. To set the RTTY engine to transmit this way, use the RTTY toolbar <**Setup**>, open the <**TX**> tab. In the middle of the form is the UOS option to transmit a [FIGS]-shift after letters or a space preceding numbers.
- **Setting Audio Levels:** as with other AFSK digital modes, it is important that your transmitted audio tones are not overdriven and distorted. Here is a quick adjustment procedure:

  1. Switch your radio to display ALC.
  2. In transmit mode, adjust your transmitted signal to the point where the ALC setting *just* begins to show a slight deflection. You can do this with slight changes in your Mic gain and/or your sound card volume controls.

  Unless you are using true FSK, set your power output to around 50% of the designed maximum output.
- **Normal and Reverse Polarity:** there is one more important step to ensure that your transmitted RTTY signal is 'the right way up.' That depends on two things: the sideband selected and the normal/inverted setting. If you have chosen to transmit using lower sideband, have your RTTY set to NORMAL (default). If you transmit RTTY using upper sideband set the RTTY engine to reverse (inverse or inverted) polarity using the RTTY toolbar Normal/Reverse button.
- **Switching modes using a hotkey**: Logger32 provides hotkeys (shortcuts) that can be programmed to trigger macros in RTTY mode for this purpose. $**RTTYReverse**$ is the macro to invert the signal, and $**RTTYNormal**$ changes back to normal. You can also use the macros $**RTTY**$ or $**RTTY-i**$ to switch from any mode to RTTY, normal or inverted respectively.

  To run RTTY and set up normal or inverted mode, use the RTTY Setup button on the toolbar. On the Demodulator tab at the extreme lower left is the Reverse option. Tick the box to operate

Copyright © 2026 Robert C. Furzer K4CY

inverse polarity. MMTTY always receives and transmits with the same polarity, so when this box is checked, you will transmit and receive in inverse polarity.

- **Transmitting with macros**: when changed to RTTY, Logger32 clears any PSK macros and starts afresh with RTTY macros. Many macros work with both PSK and RTTY (the basic ones like $**Transmit**$ and $**Receive**$ are the same), but those with "RTTY" in them are specifically for RTTY mode. To copy macros from PSK to RTTY:
  1. Open the macro for editing in PSK mode (right-click the macro button). Leave it open for editing.
  2. Switch modes to RTTY.
  3. Click <**OK**> to save the macro as an RTTY macro.
- **Example macros**: here are a few examples for you to cut and paste into the available RTTY macro buttons, editing according to your situation and preferences.
  - Responding to a CQ:

    $**Call**$ de $**MyCall**$ $**MyCall**$ KN
  - Starting a conversation:

    FB $**Name**$, The name here is Andy, QTH is Fredonia, New York. Located about 45 miles southwest of Buffalo, NY.
  - Over:

    So $**Name**$, BTU. $**Call**$ de $**MyCall**$ K
  - Brag file (equipment):

    Station at this end is a Kenwood T2000 running 30 watts into two stacked 7 element Yagis at 150 feet. Software is Logger32 $**Version**$ plus MMTTY.
  - 73 and log the QSO:

    So $**Name**$, thanks for the nice QSO, will say 73 now. QSL via LoTW. $**Call**$ de $**MyCall**$ SK $**Log**$

21.10.2 Select the transmit frequency

There is no place to enter your frequency within the Sound card data window. Frequency selection is accomplished by clicking in the display window, by tuning your radio, or by using the $**QSY**$ and $**QSX**$ macros. You presumably know how to tune your radio, so we'll just tell you about Logger32 here:

- To transmit a CQ message or tune up (briefly!), click inside the display area in an open area not presently being used by another station; or
- To respond to another station, click inside the display area on top of the other station's received signal.

Either way, your frequency is displayed on the 2nd panel of the status bar.

21.10.3 Switching to transmit and back to receive

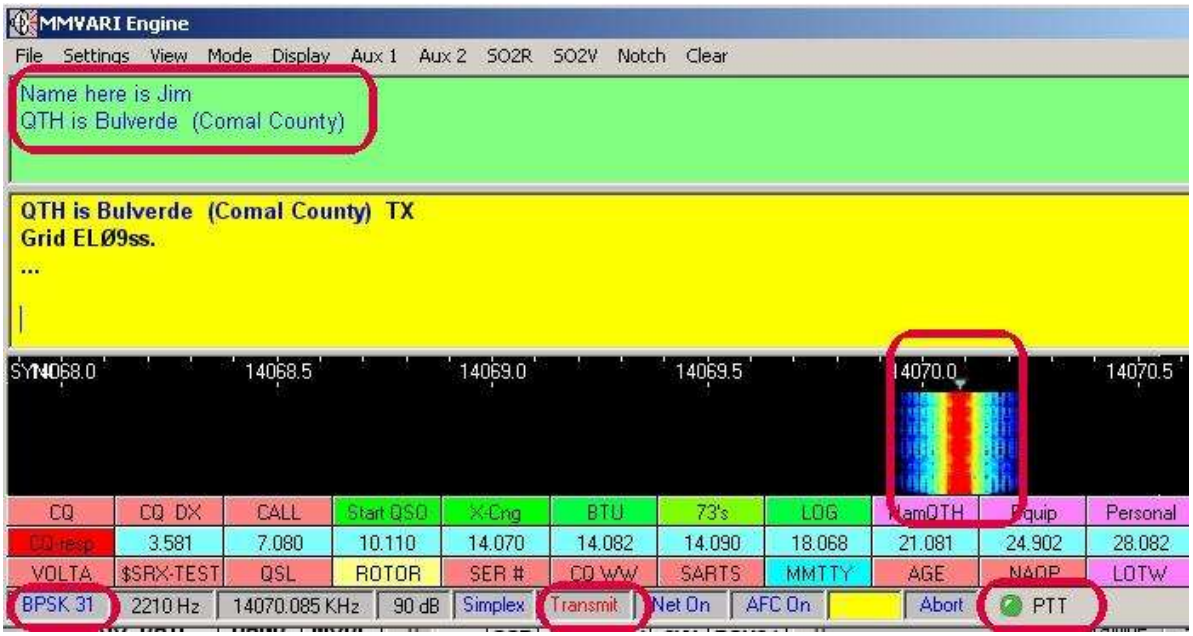
- Use the <**Pause/Break**> key.
- Use the <**Esc**> key to drop out of transmit in a hurry.

Copyright © 2026 Robert C. Furzer K4CY

- Click Receive (the 6th panel on the status bar), changing it from Receive to Transmit by grounding the radio's PTT line or sending the requisite CAT command.
- Use the $**Transmit**$ macro. See macros.

21.10.4 Indications that you are transmitting PSK

Aside from any indicators on the radio and sidetone, there are clues that Logger32 is transmitting:



- The 6th panel on the status bar indicates transmit.
- The PTT "LED" is green.
- The display shows only the one strong PSK signal at your transmit frequency (*NB* the signal shown is simply a software-generated image showing the frequencies your audio system was commanded to generate. It is not actually monitoring the generated audio nor your radio transmissions. It says *nothing* about the quality of your RF transmissions).
- The Main RX/TX Tuning window shows the PSK signal.
- Transmitted text appears in the RX window, character-by-character as the message is sent.

21.10.5 Methods to enter text in the TX window

- Type text here using the keyboard.
- Paste text from Logger32 or another application using standard Windows copy/cut and paste (mouse right-click or <**Ctrl**+**C**>/<**Ctrl**+**X**> and <**Ctrl**+**V**>).
- Insert text previously saved to macros by clicking the macro buttons or pressing the defined hotkeys.
- Load a text file from disk, using **File ⇨ Send a text file** on the MMVARI menu. This is known as a brag file, referring to overly lengthy and frankly boring equipment descriptions tediously sent by some egocentric operators. Yours isn't though, is it? Oh no.
- If an incorrect letter has not been transmitted, backspace to erase and then correct it. The corrected text is transmitted. If you are fast enough with the corrections, the incorrect text is never transmitted: nobody except you (and anyone in the shack watching over your shoulder!) knows there were errors.

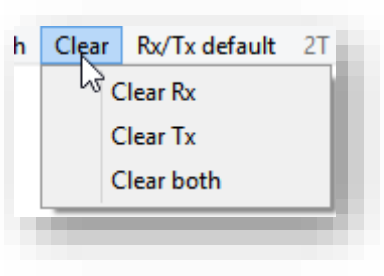
Copyright © 2026 Robert C. Furzer K4CY

- If incorrect letters have already been sent, the receiving station may see the incorrect text, then watch as letters are erased and corrected, thanks to PSK's backspace character (not available in the Baudot code used for RTTY).
- A correction can only be made to the last letter in the TX buffer. It is not possible to use the back arrow to correct text in the middle of the buffer. Erase backwards from the last letter in the TX buffer to the error, erase it, correct it, and re-type the remaining erased text.

21.10.6 Transmitting upper and lower case

PSK can transmit all the letters on a computer keyboard, both upper (capital letters) and lower case, and the special characters. Most of us type in mixed case as usual. However, uppercase letters take a little longer to transmit in PSK than lower case letters. You can force Logger32 to transmit only upper- or only lower-case letters from the MMVARI **Settings ⇨ MMVARI Settings ⇨ Typing preference** menu item.

21.10.7 Clear the TX window



Click **Clear ⇨ Clear Tx** on the menu to wipe the TX buffer ▶

The $**ClearTXBuffer**$ macro does the same thing.

Your radio remains in transmit, though. To stop sending *and* switch to receive, click <**Abort**> on the status bar. It works like an emergency stop button ▼



21.10.8 Net function

With Net turned **On**, you will transmit "simplex" *i.e.* on the other station's frequency. With Net **Off**, you will transmit where you last transmitted, probably "split".



It is possible to work split-frequency by using different audio tones for transmit and receive. Net is turned **Off** and the transmit frequency is set separately.

<**Net On**> forces Logger32 to transmit on the current receive frequency indicated in the display, and shown on the status bar frequency pane.

Copyright © 2026 Robert C. Furzer K4CY

If either or both stations have Net **On**, they share the same frequency, alternating transmissions for each over[359].

If you click the panel to turn Net **Off**, you transmit at the last-used TX audio frequency. The transmit waterfall display will be at the audio frequency at which you are transmitting. If you have made a mistake, you can abort your transmission and turn Net **On**, or, if you decide to stay where you are, you can click your own transmit waterfall during while you are transmitting to move your receive passband to your transmit frequency, so you can copy someone who calls you your transmit frequency (zero-beat).

21.10.9 Operating split through the main window

We will assume that you know where the station you will call is operating, and have that station in the receive range.

- Turn Net **On** to transmit wherever you are receiving.
  - Put the cursor on the spot where you want to transmit.
  - Make a very brief transmission at that location (click transmit on and off) to move the transmit frequency to that location.
- Turn Net **Off** to leave the transmit frequency where you last transmitted.



Click the station you are trying to work so you can receive him. The transmit diamond remains at 2210 Hz.



At this point, you will transmit where you last transmitted because Net is off. You will receive where you click.

You will note that the transmit audio frequency is 1179.9 Hz and the receive audio frequency is 825.9 Hz. You are now ready to give a call:

> "CQ Split Listening +/- 350 Hz down de W5IFP"

[359] Rule of thumb: the station that calls CQ or initiates a contact should leave Net **Off**, his transmitter staying put (assuming it doesn't drift!). Callers should have Net **On** to transmit on and track the CQer's frequency.

Copyright © 2026 Robert C. Furzer K4CY

You can program a macro to repeat this CQ with the following text:

$**Transmit**$ CQ Split CQ Split listening +/- 350 Hz down de $**MyCall**$ $**MycCall**$

$**Transmit**$ CQ Split CQ Split listening +/- 350 Hz down de $**MyCall**$ $**MyCall**$ pse K

$**Receive**$

The audio frequency is still showing 1811 Hz. The actual transmitted signal will be 2210 Hz, which was the TX frequency where you turned Net **Off** ▼



21.10.10 Operating split with Aux

Say you come upon a DX station operating split and calling "CQ DX CQ DX de P5DX listening down 500 Hz". Here is a way to follow his instructions:

1. Turn Net **Off** (**this is crucial!**). From now on, your transmit frequency will *not* follow your receive frequency.
2. Open the Aux 1 RX window (blue passband). Look for activity down 500 Hz (*e.g.* if he was calling on 1400 Hz, you should look down around 900 Hz).
3. If you see some activity close to 900 Hz, hold <**Alt**> and click the signal in the display, to bring the Aux 1 RX window to that area (blue passband).
4. Click the audio frequency panel (second from left) on the Aux 1 window status bar. This moves your transmitter VFO to that frequency. When you transmit, your signal now will be at the blue passband. On receive, you are decoding the station working split under the red passband Main RX Window.

You can use either Aux 1 or Aux 2 to set the transmit frequency when working split. Receive the station in the Main RX window.

When you are done, reset your transmit frequency to the receive frequency by clicking <**Reset**> on the status bar of the main receive window. Remember to turn Net **On** to keep your TX and RX frequencies aligned automatically.

Copyright © 2026 Robert C. Furzer K4CY

The following example shows receiving the station in the Aux window at 2195 Hz. The transmitted signal in this example would be at 1706 Hz in the Main RX window ▼



21.10.11 Operating split with Aux reversed

1. Turn Net **On**.
2. Receive the station you are trying to work with Aux 1 or Aux 2.
3. Click the cursor where you want to transmit, sending the Main (red) passband indicator there.
4. At this point, you will transmit where the red indicator (main passband) is, and you receive in the Aux window.

21.10.12 Using split mode on the radio

If your radio has two VFOs, enable split on the radio and set the transmit VFO to operate where you want. This is the same method you would use for SSB or CW split. Leave Net **On** in Logger32 and you will be shifted an amount exactly equal to the difference between your two VFOs.

21.10.13 FSK split operation

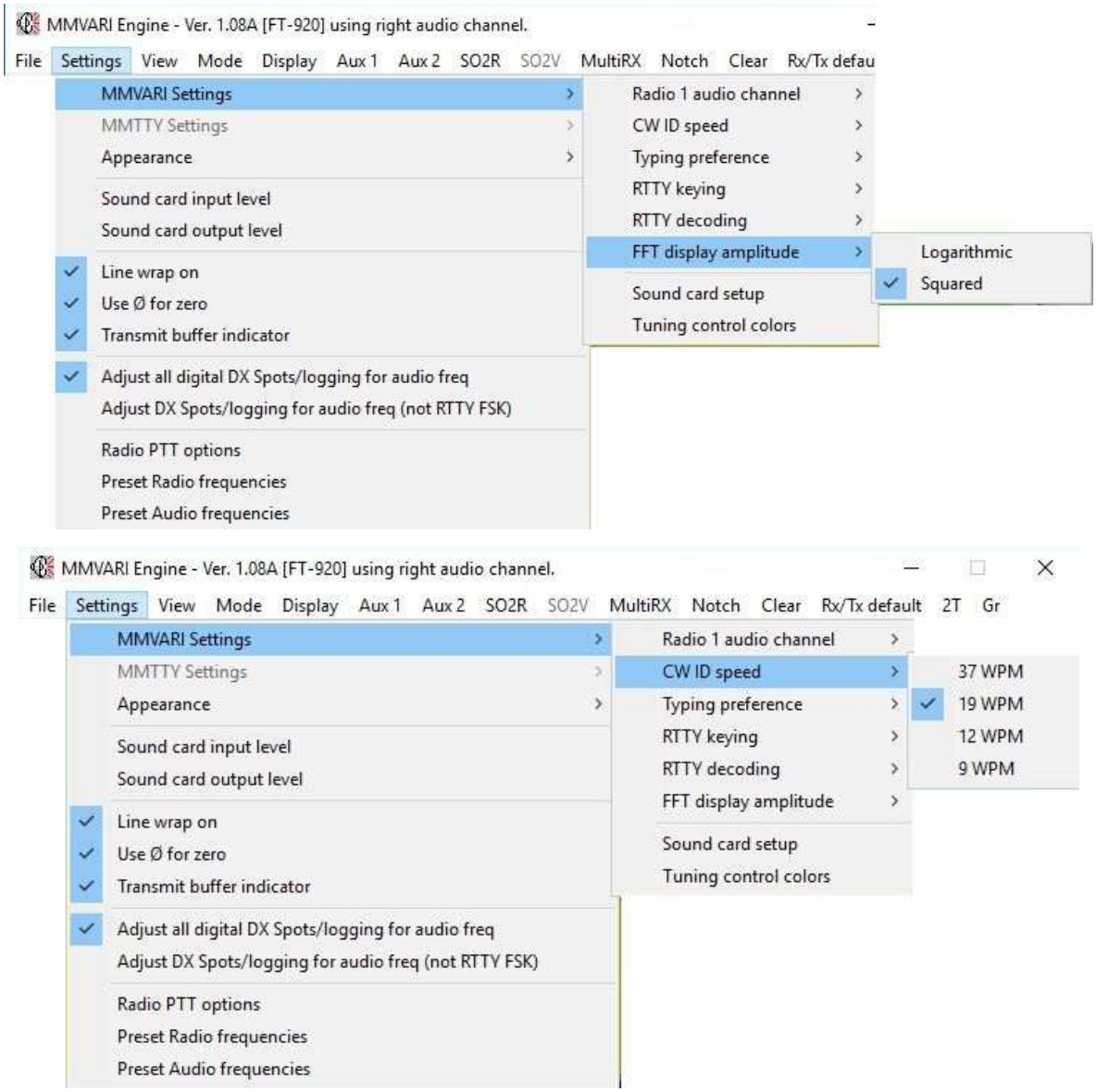
Due to the fixed transmit offset associated with FSK operation, it is more practical to use conventional methods of split operation with dual VFOs. The MultiRX window can still be useful to find a clear spot to transmit or find the DX station's listening frequency.

21.10.14 TX/RX tuning window

When MMVARI transmits, a histogram display in the small TX/RX tuning window represents the *audio* that is being sent from the sound card to the radio. The TX/RX tuning window is not a display of your radio's RF output, hence it does not show overdrive or other downstream issues.

Copyright © 2026 Robert C. Furzer K4CY

The following are user-selected options for the MMVARI window. The highlighted menu options affect the amplitude of the spectrum/waterfall display ▼



21.10.15 Operating SO2V

Put the DX station (your listening frequency) in the SO2V window. Find a suitable transmit frequency by monitoring the MultiRX or the main RX window. Make sure your transmitter is using the VFO displayed in the main window.

21.11 Communicating with RTTY

21.11.1 Mark and space

An RTTY transmitter sends out a continuous carrier that shifts back and forth between two distinct frequencies *i.e*. a form of frequency modulation. There is no amplitude modulation, only a pure carrier similar to CW with the addition of a frequency-shift. The lower RF frequency is known as the space frequency and the upper RF frequency is known as the mark frequency. The difference

Copyright © 2026 Robert C. Furzer K4CY

between the two is known as the shift. For amateur radio, the shift has been standardized at 170 Hz. It is customary to refer to the mark frequency as the frequency on which you are operating. For example, if you say you are transmitting on 14080 kHz, your mark frequency is 14080.00 kHz and your space frequency is 170 Hz lower at 14079.83 kHz.

21.11.2 Figures-Shift and Letters-Shift

RTTY uses the Baudot code, invented before radio even existed, and still widely used throughout the world. The Baudot code uses data bits to represent letters, numbers and punctuation, much like your computer does. Unlike your computer, which uses eight bits for each character, the Baudot code uses only five, plus a start bit and stop bit. Using fewer bits is good because it speeds up transmission and reduces the chance of errors, but there is a complication. Five data bits can only represent 32 different characters. Since there are 26 letters in the English alphabet plus ten numbers, plus some punctuation, 32 different characters is not enough, even if you only use capital letters, which Baudot does.

Mr. Baudot could have chosen to use six data bits or even more, but he found a better solution. He reasoned that most of what would be sent would be letters rather than numbers or punctuation, so he assigned all the letters to the basic 32. He then had six characters left over and he did a very clever thing with two of them. He made one of them a Figures-Shift and another a Letters-Shift. When sending one of the basic 32 characters, nothing special happens but when a number or punctuation is to be sent, a Figures-Shift character is sent first (it's a non-printing character - you won't see it on your screen). Whatever follows will still be one of the basic 32 characters, but the receiver will interpret it differently. For example the letter Q uses the same five data bits as the number 1, but when the receiver gets a Figures-Shift first, it prints the next character as a 1, not a Q. This continues until a Letters-Shift character is received, at which time the receiver goes back to "normal" printing. All of this shifting is done by the system - there is no key marked Letters-Shift or Figures-Shift. It's all automatic and you will scarcely notice it happening.

In fact, the only reason to mention it at all is because we are using radio instead of wires, and radio is susceptible to interference from various sources such as lightning static, man-made noise, *etc*. If a burst of static should happen to wipe out a Letters-Shift or Figures-Shift character, the characters following will not print correctly until another Letters-Shift or Figures-Shift is received. For example, suppose you are sending a signal report of 599, but the Figures-Shift character gets wiped out by a burst of static. Instead of printing 599, the other fellow's computer will print TOO. TOO is exactly the same as 599, without the Figures-Shift. So how can he read what you intended to send? It's easy if he knows the secret and here it is, look at the top row of letter keys on your keyboard - QWERTYUIOP. Now look just above each key and to the left. Each of those number keys is the same as the letter key below and to the right, plus the Figures-Shift. In our example, TOO = 599. Likewise, the word PIPE, if the Letters-Shift were missed, would print as 0803. If 0803 lost its Figures-Shift, it would print as PIPE.

When the bands are nearly empty, you can use practically any receiver bandwidth with good success. Your SSB filters are probably between 2.1 and 3.0 kHz wide and as long as no other stations are nearby, copy will be fine. For optimum performance however, less bandwidth is better, in fact *much* better. 170 Hz shift RTTY only needs about 250 Hz for proper copy. If you don't have a 250 Hz filter, 500 Hz will do pretty well, but anything wider than that will not be satisfactory in the long run.

Copyright © 2026 Robert C. Furzer K4CY

You may wonder why, if the shift is 170 Hz, do you need a 250 Hz filter? Why not 170 Hz? The reason is that shifting the frequency generates sidebands adjacent to the actual signal and if the sidebands are attenuated, the signal will be degraded. RTTY is actually a form of FM, and if you'd like to understand more FM theory, there are a large number of books available. For amateurs, the ARRL Handbook is a good source.

Depending on your transceiver, you may or may not be able to use a narrow filter for RTTY. Some transceivers do not have an FSK mode may be unable to select a narrow filter for SSB. Some improvement can be made by using an outboard audio filter between the speaker output and the sound card input but, unfortunately, that will not prevent a strong adjacent signal from causing the receiver's AGC circuit to reduce gain. The best solution is to upgrade to a transceiver that has an FSK mode built in, and which allows you to select a narrow filter while in that mode.

21.11.3 Advanced RTTY receiving

Now that you are receiving, here are ways to fine-tune the received signal. First, we will look at the RTTY toolbar at the top of the RTTY receive window.



- **MMTTY**: in RTTY mode, an additional iconic toolbar offers the following controls:
  - **Squelch**: set the level by dragging the vertical bar on the Tuning window.
  - **Bandpass Filter**: set your transceiver to a wide passband and turn this on and off to see its action. This audio DSP filter remains centered on the receive frequency.
  - **Normal/Reverse:** swaps the "sense" (mark/space ⇄ space/mark).
  - **XY display**: shows or hides the tuning scope display.
  - **UOS**: sets or clears **U**nshift **o**n **S**pace – see below.
  - **FIGS-shift:** see below.
  - **LTRS-shift**: see below.
  - **Setup**: configure many receive and transmit parameters.
  - **Net**: transmit on your receive frequency. Disable this for audio split operation.
  - **AFC**: enables **A**utomatic **F**requency **C**ontrol to track the received signal - see below.
  - **Reset**: does the same as on the status bar, namely resets to the preset transmit and receive frequencies and shifts.
  - **Abort**: stop transmitting ***right now!***
- **Squelch in MMVARI**: only when the signal is stronger than the squelch level is decoding attempted. Squelch can decrease the number of mis-copied characters but setting the squelch level too high can lead to otherwise sound characters being missed. Change the squelch level by placing the cursor over the 9th panel in the lower status bar and clicking the mouse where you want the decoding to start/stop. Click to the left for an open squelch (print everything) or to the right for a tight squelch (print only strong signals). The red line shows where the squelch is set, while the dancing green bar represents the current audio signal strength.
- **Bandpass filter**: the bandpass filter narrows the receive bandpass, using audio DSP. This filter remains centered on the RTTY passband when the passband moves.

Copyright © 2026 Robert C. Furzer K4CY

- **Normal/reverse sense**: this inverts (swaps over) the mark and space tones. Conventionally, the radio is set to LSB with the switch set to Normal, so the mark signal is higher in radio frequency than the space signal. To communicate with a station that is “upside-down”, “inverted” or “backwards” in sense, press this button to reverse your mark and space tones. Your transmissions will change to the same sense as for receive. Don’t forget to change back afterwards!
- **U**nshift **O**n **S**pace (UOS): enable UOS unless you have a reason not to do so. See below for a way to set your transmissions to help others who may not be using UOS.
- **FIGS/LTRS shift**: if it looks as if you have missed a LTRS- or FIGS-shift you can use these two buttons to force Logger32 to make the shift on receive. This can only be done to characters that have not been decoded. Once a character is printed, the shift cannot be changed retrospectively.
- **AFC**: you have a choice of four AFC modes on the Setup (toolbar), AFC/ATC/PLL tab:
  - **Free**: allow both mark and space to vary to copy the signal best. Transmit with the receive mark/space.
  - **Fixed**: mark can change, but shift is fixed at the default value.
  - **HAM**: the mark can change, but the shift can only take the values of 170/200/220/240 Hz, whichever works for most letters copied. The shift only changes on receive, remaining at the HAM default for transmit.
  - **FSK**: the center frequency cannot change and the shift can only take the values of 170/200/220/240 Hz, whichever works for most letters decoded.

  You can set the mark and shift by clicking **Setup ⇨ Demodulator** tab and setting them at the left. Click <**OK**> to leave this display and save the mark/shift values. If you set the AFC at Fixed, the mark frequency can change to try to copy a signal, but the shift will stay as you set it on the Demodulator tab.

  In Free AFC mode, the shift changes to match the measured shift of the received signal when you turn the AFC on. You can return to the standard HAM values by pressing Reset on the RTTY toolbar or the Status bar. The HAM values can be set on the Demodulator Tab. MMTTY comes with these preset at mark = 2125 Hz, shift = 170 Hz.

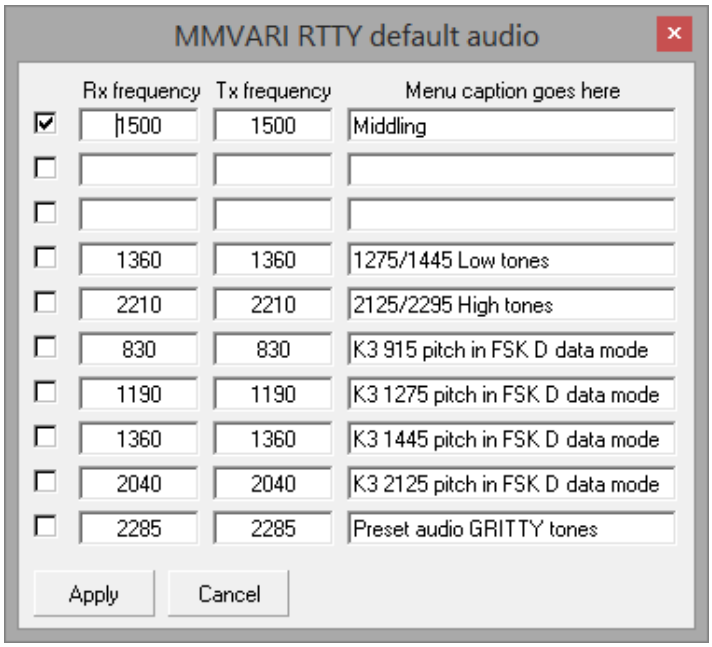
  When you set the AFC at Free and turn AFC on, Reset returns you to the original mark/shift values for a moment, then Logger32 looks around for RTTY signals and varies these parameters in an attempt to copy something.

  Click to Reset Shift: when you are in an AFC mode that allows the shift to change (such as Free), Logger32 returns the shift to the HAM default every time you click in the waterfall. This means that Logger32 always begins to try to decode a new signal at the default shift.
- **Reset**: the <**Reset**> button sets the mark and shift frequencies back to the values set on the Demodulator tab. RTTY and PSK modes differ in this: PSK reverts to defaults in the Settings menu.
- **Show RTTY commands**: Logger32 can echo in the receive window the RTTY control commands that you issue from the toolbar or macro. Click **View ⇨ RX Window Options ⇨ Show RTTY Commands** and they will appear in the receive window within brackets < >.
- **Show control codes**: you can view the control codes that are sent and received by Logger32. Control codes are carriage returns. Click **View ⇨ Window Options ⇨ Show Control Codes**.

Copyright © 2026 Robert C. Furzer K4CY

- **Macros**: macros for RTTY are explained in the macros chapter. You can use Logger32 macros that are relevant to RTTY or to general Logger32 operation (*e.g.* $**Transmit**$), and you can use some MMTTY macros if you are familiar with them, as long as they apply to RTTY operation in Logger32, and do not require parts of MMTTY that are not integrated into Logger32. For instance, you can send %d for Logger32 to send the date, but you cannot use %c to send the other station's callsign: you need Logger32's $**Call**$ macro because of the way that Logger32 handles callsigns.
- **Notch filter**: MMTTY provides one or two audio DSP notch filters for use during reception. To enable the notch filter(s), go to RTTY Setup (toolbar), Demodulator tab. Now look to the right and you will see two smaller tabs, one is LMS/Notch. Click the LMS/Notch tab. Set the tabs at 256 to start, this gives a relatively narrow, deep notch, and click the box next to Notch. You can also invoke two notches, but they will be at the same frequency, resulting in a deeper notch. Click <**OK**>. Turn the notch on by clicking in the graphic frequency display. A bar appears at the notch frequency. Right-click in the frequency display and the notch turns off. If you click the waterfall, you will move the receive passband: click the *frequency scale* to operate the notch.
- **Display smoothing**: the spectrum display can be smoothed, as for PSK. Click **Settings** ⇨ **MMTTY Settings** ⇨ **Display smoothing** and select (tick) your preference:
  - **Fast display (no smoothing)**: signals are displayed 'raw' in near real time with no damping. You'll see noise spikes and the signals varying rapidly up and down in strength.
  - **Medium and slow display (medium or heavy smoothing)**: displayed signals are averaged over a period of time. The effect of these settings is to reduce the noise spikes, lower the overall background noise, and prevent the signal from jumping up and down on the display. The effect of the smoothing is easily seen on the display. It is easy to experiment on received signals and determine which best suits the band conditions.
- **Align**: when you transmit, there is less likelihood of your transmitting harmonic distortion products if you use high tones. Logger32 has an automated method for transmitting with high tones on a CAT-connected radio *i.e*. the $**Align**$ macro or right-clicking in the display will:
  1. Move the decoder to a preset audio frequency.
  2. Send a QSY command to your radio such that whatever signal you had originally selected and been decoding ends up at the preset audio frequency.

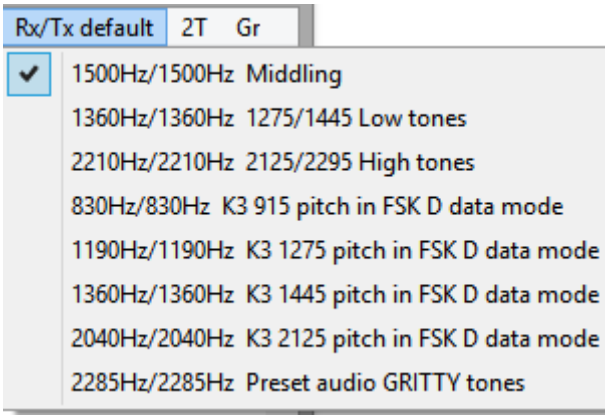
  Here is how to align in MMVARI:
  - First, define a default audio frequency under **Settings** ⇨ **Preset Audio Frequencies** *i.e*. click to tick the relevant row then <**Apply**>.
  - If the desired frequency is not listed, edit the Tx & Rx frequency box to the preferred operating tones.



Copyright © 2026 Robert C. Furzer K4CY

- For FSK operation, the default should place the receiver and transmitter on the same frequency. If the transceiver uses high tones (2125/2295 Hz), the Tx and Rx default frequency should be set to 2210 Hz (midway between the two[360]). Likewise if your transceiver uses the low RTTY tone pair (1275/1445 Hz), the default should be set to 1360 Hz.

The tones may also be setup to match the passband of the transceiver bandpass filters. Once established, the tones can simply be selected from the Rx/Tx default menu ▶



The $**Align**$ configuration is similar for RTTY in MMTTY, except that it is only necessary to set an RX audio default. In MMTTY with AFSK, the TX frequency is either the HAM default or, if Net is on, the RX frequency. In MMTTY with FSK, the RX audio default should be set to coincide with the transceiver's fixed-frequency FSK RTTY tones, using an RX setting midway between the FSK mark and space frequencies.

21.11.4 Turn AFC on

To use AFC, tune in an RTTY signal somewhere on the receive range away from the default RX audio frequency. Right-click near the middle of the receive passband lines. After a second or two determining the frequencies, Logger32 sends a frequency-shift command ($**QSYxxxx**$) to your radio so that the signal appears at the default preset audio frequency. At the same time, the decoder is moved to that same area then fine-tuned.

You can also use the $**Align**$ macro by defining the command on a macro button.

21.11.5 RTTY profiles

Logger32 can save and recall groups of RTTY parameters in profiles.

The HAM default in MMTTY makes it easy to return to a basic set of RTTY operating parameters. However, with new modes, contest conditions and other situations, custom sets of operating parameters can be useful.

To save an RTTY profile:

- **Using the Sound card data window's <Mode> menu**: go through the RTTY setup and set all the options the way you want to operate. Open RTTY mode and set all your parameters the way you operate. Don't forget the MMTTY settings under the wrench icon on the toolbar.
- Once you have set up the system to your liking, save the profile so the system will initialize with your parameters each time you open the RTTY mode:
  1. Go to Mode ➪ Setup RTTY profiles.
  2. Select <**Save RTTY profile to file**> and follow the prompts, giving it a filename.

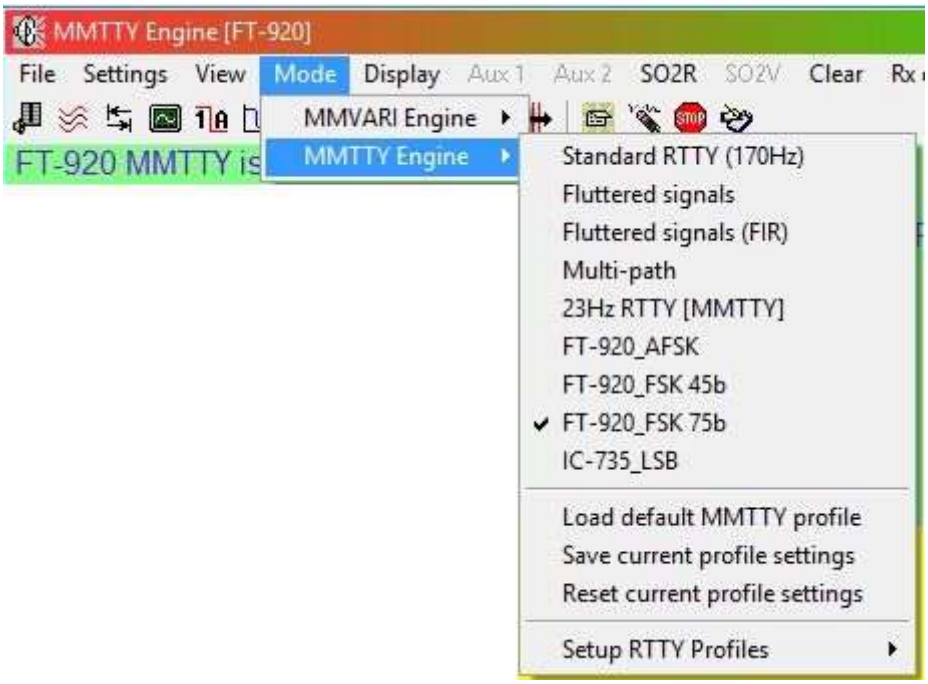
[360] To calculate the mid-point (the median frequency), simply add the mark and space frequencies together, then divide by two.

Copyright © 2026 Robert C. Furzer K4CY

3. Repeat step 2 above and select <**Assign RTTY Menu**>.
4. Assign the exact name you gave the profile that you saved (without the .*Pro* extension).
5. Click <**Mode**> and you will now see your saved profile in the menu. Click to tick it as your default profile each time you open RTTY.

You can have additional RTTY profiles with a menu item for each one. Logger32 recalls the last used profile the next time RTTY is selected.

Once you have established a profile you can also save it to a file using the menu item under **Mode** ⇨ **MMTTY Engine**, saving the setup under the currently selected profile.

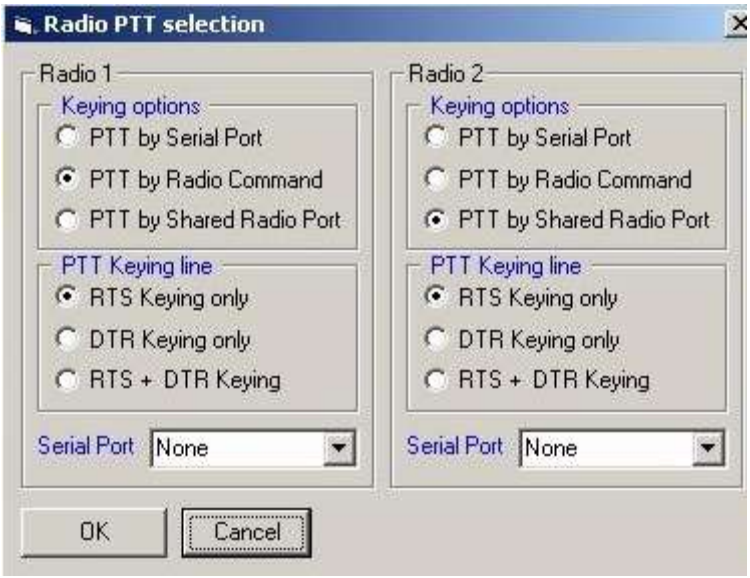


- **Using the MMTTY "Profile" button**: follow the same procedure as outlined above, using the Sound Card MODE menu.

21.12 Transmitting RTTY

Logger32 uses MMTTY and MMVARI soft modems to encode and decode RTTY. AFSK and FSK keying methods are supported by both MMTTY and MMVARI.

Copyright © 2026 Robert C. Furzer K4CY



- **Radio PTT options**: the PTT setup applies to both MMTTY and MMVARI (If you change PTT settings it affects both programs). Open the Sound card, click **Settings ⇨ Radio PTT options**, set the appropriate PTT options and port and click <**OK**> ▶
- Setting up MMTTY for AFSK:
  1. Open the Sound Card and select MMTTY.
  2. Open **Settings ⇨ MMTTY settings ⇨ RTTY Keying**.
  3. Click to select <**Radio 1 uses AFSK**>.
  4. Open the MMTTY toolbar and the <**MISC**> tab.
  5. Under the TX Port select (.) Sound and click <**OK**>.

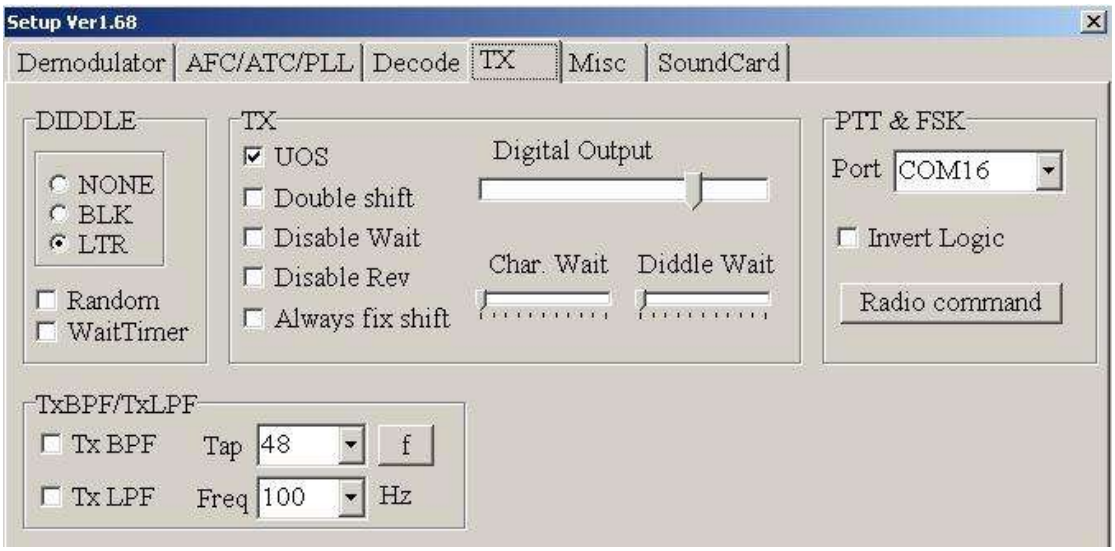
  MMTTY is now ready for RTTY operation.
- Setting up MMVARI for AFSK:
  1. Open the Sound card data window for MMVARI, mode RTTY-L.
  2. Open **Settings ⇨ MMVARI settings ⇨ RTTY Keying**.
  3. Click to select <Radio 1 use AFSK for RTTY>.

  MMVARI is now ready for RTTY operation.
- **Setting up RTTY for FSK**: FSK keying works in both MMTTY and MMVARI. There are five ways to run RTTY using FSK:
  1. Dedicated COM Port *Either an actual RS232 port or a USB-RS232 adapter
  2. Parallel port *Requires EXTFSK64 add-on
  3. EXTFSK ADD-ON *Used by MMTTY to interface with a USB-RS232 adapter
  4. EXTFSK64 ADD-ON *Used by MMTTY & MMVARI for 64-bit computers.
     *Supports 75baud RTTY, serial (COM) and parallel (LPT) ports.
  5. FSK8250 ADD-ON *Used by MMVARI for legacy serial (COM) ports.

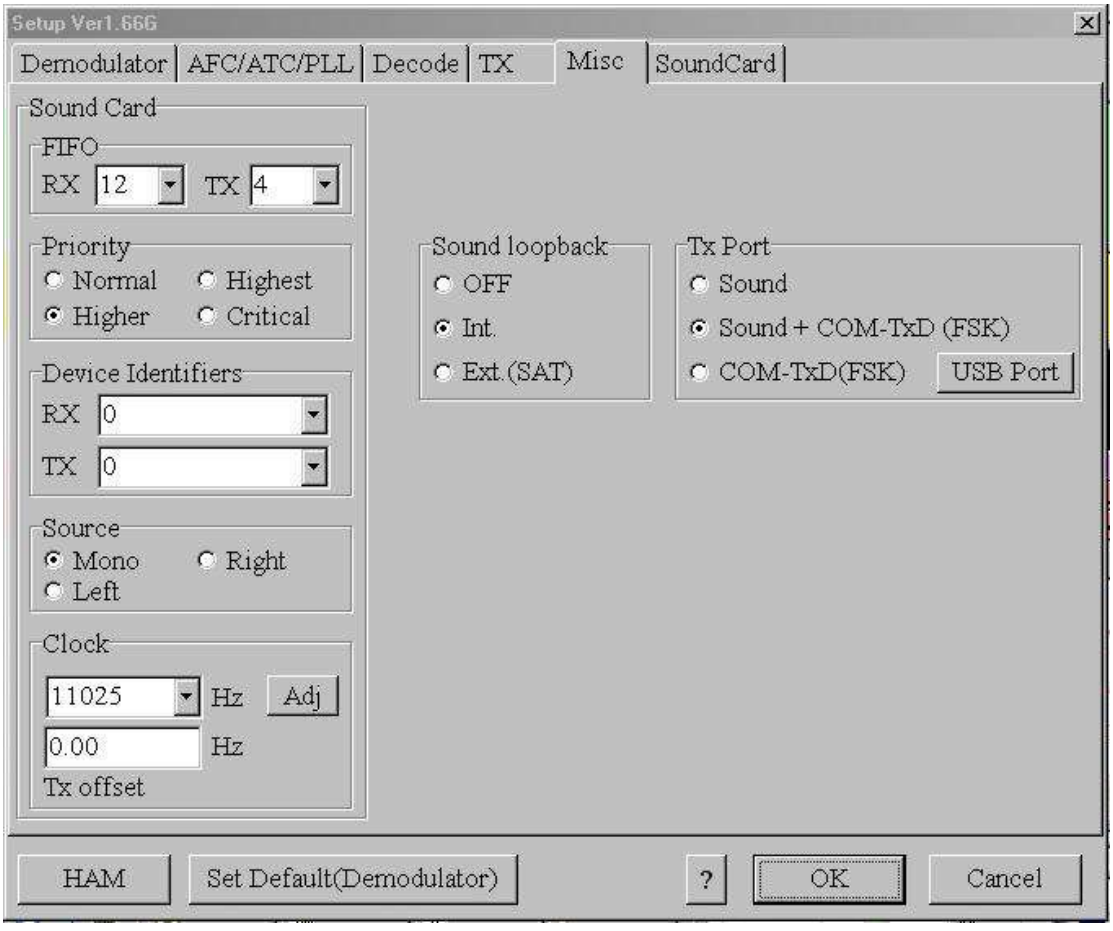
  *Can be used with MMTTY, but is not required.
- RTTY FSK keying on a dedicated serial (COM) port. This is available in both MMTTY and MMVARI. The PTT defaults to the RTS line with FSK keying on the Txd line. MMVARI supports most generic USB adapters without add-ons. MMTTY supports legacy RS232 (COM) ports and USB-RS232 adapters. MMVARI will work on a legacy COM port, but requires the FSK8250 add-on driver for this.
- Setup MMTTY for FSK keying on a dedicated COM port:
  1. Set the Radio PTT options as defined above in the Radio PTT Options section.

Copyright © 2026 Robert C. Furzer K4CY

2. Open **Settings ⇨ MMTTY settings ⇨ RTTY Keying** and tick <**Radio 1 uses FSK**> or <**Radio 1 uses FSK and AFSK**>.
3. Open the MMTTY setup, TX tab. Select a COM port in the PTT & FSK section – specifically, a port *other* than the CAT port ▼



4. Click the MMTTY setup and select the <**Misc**> tab.
5. Select one of the FSK options in the <**TX Port**>.
   - <**Sound** + **TxD (FSK)**> provides FSK keying and also allows you to monitor the audio from the sound card. <**COM-TxD (FSK)**> provides the same FSK keying but the sound card output is muted.

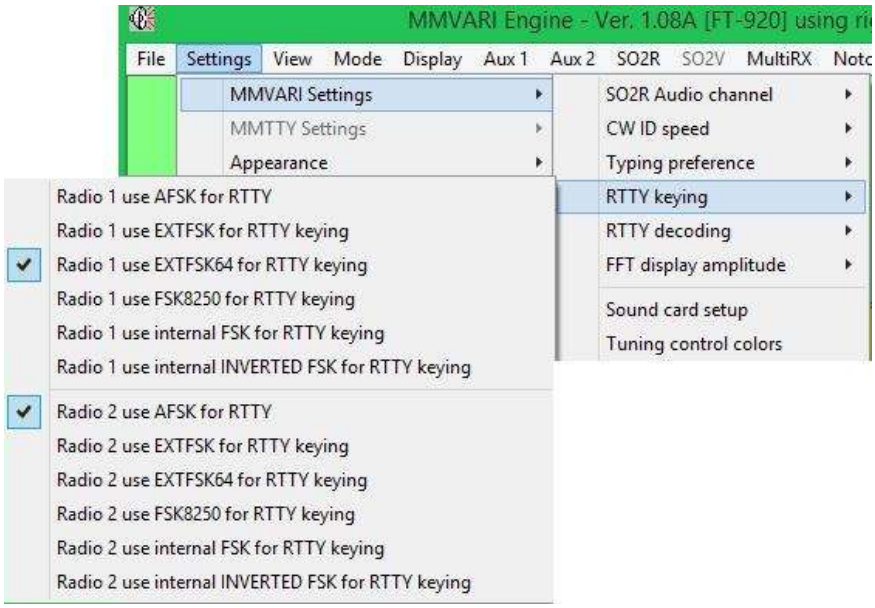


Copyright © 2026 Robert C. Furzer K4CY

Save your settings: open **Mode ⇨ MMTTY** and click <**Save current profile settings**>.

21.12.1 Setup MMVARI for FSK Keying on a dedicated COM port

1. Set the Radio PTT options.
2. Define an FSK port *other* than the CAT port under **Settings ⇨ MMVARI Settings ⇨ RTTY Keying**.
3. If the FSK keying port uses a USB adapter tick <**Use FSK for RTTY keying**> or <**Use Internal INVERTED FSK for RTTY Keying**>. If an RS232 serial port is used, tick <**Use FSK8250 for RTTY Keying**>.



21.12.2 FSK keying using the *EXTFSK* or *EXTFSK64* add-ons

The setup outlined below is the same for *EXTFSK* or *EXTFSK64*. Use *EXTFSK* if you are running the 32-bit version of Windows. Either will work on 64-bit Windows systems[361].

This add-on can be used in both MMVARI and MMTTY. *EXTFSK.DLL* must be in *C:\Logger32*.

EXTFSK is a user-customizable PTT/FSK **A**pplication **P**rogram **I**nterface of MMTTY to interface to a Windows DLL just like a plug-in module.

You can use *EXTFSK* through an RS232 serial (COM) port, USB-RS232 adapter, or even a Centronics parallel (LPT) port (a legacy from the 1980's!).

The *EXTFSK* add-on is used to drive USB-RS232 adapters and some USB interface equipment. *EXTFSK* is usually required to drive USB-RS232 adapters where the speed of the USB device does not support RTTY's slow 45 baud rate.

EXTFSK provides the added advantage of passing the FSK/PTT signals to the Txd, RTS or DTR signal lines. It also allows reverse FSK keying.

21.12.3 MMTTY setup using *EXTFSK*

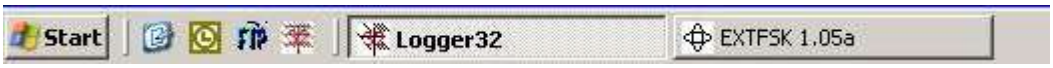
The following steps also apply to *EXTFSK64* when using a COM port:

1. Open the Sound card data window.
2. Select MMTTY.
3. Open **Settings ⇨ MMTTY Settings ⇨ RTTY Keying** and select <**Radio 1 uses EXTFSK**>

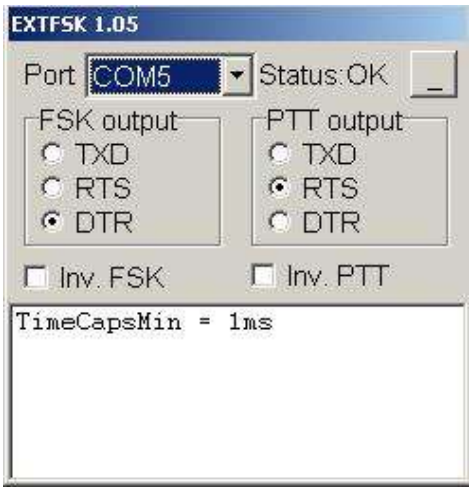
[361] If you are unsure, press <**Windows+X**> then tap Y to see information about your Windows system.

Copyright © 2026 Robert C. Furzer K4CY

4. Click <**OK**> to start the add-on. An EXTFSK button appears in the Windows task bar ▼



5. Click the EXTFSK button to open a Setup window ▶
6. Select the port you plan on using for FSK. This *must not* be the port used for CAT.
7. Choose the signal line you would like to use for PTT and FSK.



The line selected for PTT will key the radio *in addition to* the PTT line/method set in **Settings ⇨ Radio PTT options**. You can therefore use a dedicated port for FSK *and* retain your original settings for PSK.

To save your settings, open **Mode ⇨ MMTTY** and click <**Save current profile settings**>.

MMTTY is now ready to go!

21.12.4 MMVARI setup using *EXTFSK*

The process is virtually the same as for MMTTY (see above) except under **Settings ⇨ MMVARI Settings ⇨ RTTY Keying** you click <**Radio 1 uses EXTFSK for RTTY Keying**>.

21.12.5 RTTY FSK Keying using the *FSK8250* add-on

This add-on emulates a true FSK port using RS232 COM ports or USB-RS232 adapters. It is used with MMVARI to support FSK interface adapters that are designed to connect to an RS232 serial port. It supports all keying speeds that are available in MMVARI.

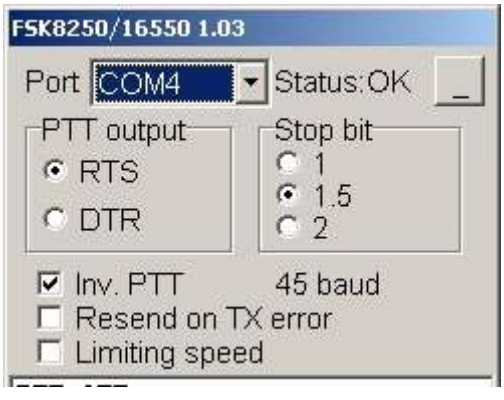
First, put *FSK8250.FSK* in the *C:\Logger32* folder.

21.12.6 MMVARI setup using *FSK8250*

1. Open the Sound card data window.
2. Select MMVARI in RTTY-L mode.
3. Open **Settings ⇨ MMVARI Settings ⇨ RTTY Keying** then select <**Radio 1 uses FSK8250 for RTTY Keying**>. The FSK8250 button appears in the Windows task bar ▼



4. Click the FSK8250 button to open a Setup window ▶
5. Select the port you plan on using for FSK- *not* the CAT port.
6. Select the signal line you would like to use for PTT.



The line selected for PTT will key the radio as well as the selected PTT line/method used in **Settings ⇨ Radio PTT options**. This allows you to use a dedicated port for FSK and retain your original settings for PTT.

Copyright © 2026 Robert C. Furzer K4CY

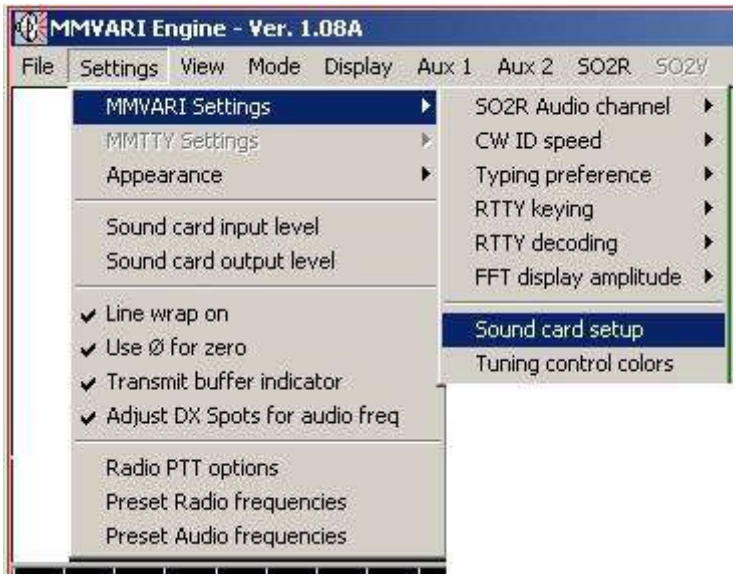
Select RTTY using the Sound card data window's mode menu, the 1st panel on the status bar, or a button programmed with a macro such as $**RTTY**$, $**RTTY-i**$ or $**MMVARIMode *N***$.

21.13 Calibrating the sound card

Computer sound cards are *supposed* to run at a given clock frequency which determines the sampling frequency, most often 11025 Hz. However, cheap cards are not that accurate and frequency errors are common. Calibration involves measuring the actual sampling frequency of the computer sound card, allowing Logger32 to compensate for any error. The actual sound card clock frequency is set by a crystal and is not changed: this is purely a software adjustment to the generated tones.

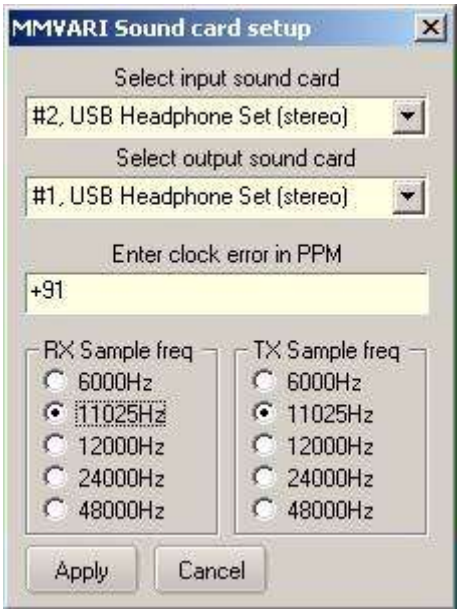
21.13.1 MMVARI sound card sample rate

Proper sample frequency is necessary for accurate data decoding. The sample frequency also affects the quality of the transmitted signal. MMVARI defaults to 11025 Hz which works OK with most sound cards. On slow PCs, however, the sample rate may need to be reduced to (say) 6000 Hz. Conversely, high-end audio systems and radios can sample at 48000 Hz or more.



◀ To accommodate these variations, the software can compensate for sound card sample frequency errors under **Settings ⇨ MMVARI Settings ⇨ Sound card setup**.

Select the desired sample frequencies and click <**Apply**>. This dynamic setting is applied immediately ▼



The 11025 Hz value is based on the old CD sampling standard (44100 Hz divided by 4). However today's sound cards are based on the DVD standard of 48000 Hz, so it is better to use 12000 Hz, 24000 Hz or 48000 Hz.

With some sound cards, the sample rate affects the AFC. If your system is slow to lock on to the frequency when you click a signal on the waterfall, try changing the sample rate.

The next few sections discuss two ways of calibrating Logger32 in RTTY[362]. Once you have done this, you can use the value you have found to modify the PSK section of *C:\Logger32\Logger32.INI* file to improve PSK operation.

[362] If you are using more than one sound card for digimodes, calibrate each card separately since they have their own frequency references.

Copyright © 2026 Robert C. Furzer K4CY

21.13.2 Calibration using MMTTY and a frequency counter

This process uses an audio frequency counter:

1. Set the frequency counter to measure the audio frequency of the audio output from the sound card (the line that goes from the sound card to the audio input of the radio).
2. Turn off AFC using the RTTY toolbar panel.
3. Click **Setup** ⇨ **Misc** tab and confirm that the clock frequency at lower left is set to 11025 Hz.
4. Turn off diddle on the TX tab.
5. Set the mark frequency to 2000 Hz on the Demodulator tab.
6. Click <**OK**> to close this display.
7. With the transmitter disabled (*e.g.* turn off VOX or disconnect the PTT line) go to transmit in MMTTY. So long as no characters are being sent, it generates the steady mark tone of 2000 Hz.
8. Measure the frequency of the tone using the frequency counter.
9. Calculate the *actual* sampling frequency from the measured tone. The proportion of error of the tone shows how far the frequency is off. For example, with a mark frequency of 2000 Hz, and a measured tone of 2010 Hz, the actual sampling frequency is high – in fact, 11080.125 Hz *i.e*. 2010/2000 x 11025.

   Now correct the sample frequency …

   - Click **Setup** ⇨ **Misc** tab and set the frequency to 11080, putting it as close as you can to the calculated value.
   - Click <**OK**> to exit Setup.
   - Close and re-start Logger32.
   - Re-check the frequency of the 2000 Hz tone to confirm the correction.

21.13.3 Calibration using a time standard on the short waves

MMTTY has a special display that allows you to find the true clock frequency of the sound card using an *accurate* 1-second tick received over the air from reference transmitters such as:

- **WWV** and **WWVH** at 2500, 5000, 10000, 15000 and 20000 kHz.
- **GBR** on 60 kHz.
- **RWM** on 4996, 9996 and 14996 kHz.
- **CHU** on 7335 kHz.

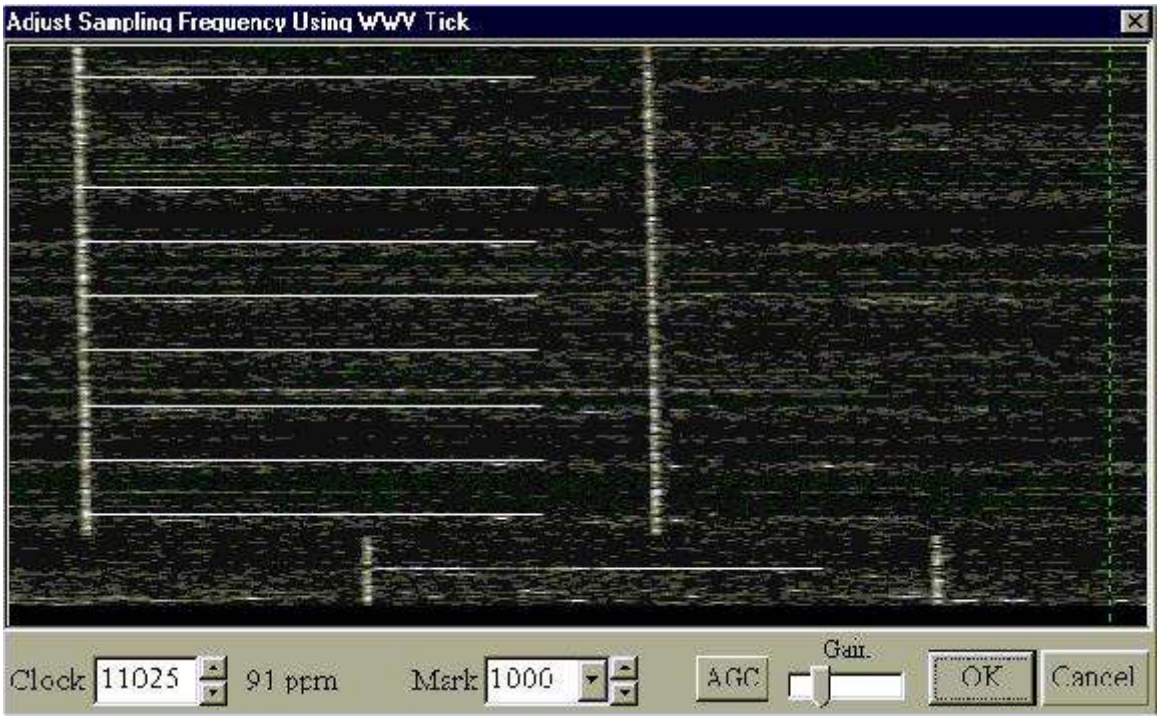
Having tuned your receiver to a reasonably strong time standard, the MMTTY calibration routine monitors the sound card output for two seconds, painting a line across the screen showing the strength of the sounds. If the clock is set to 11025 Hz, the software puts 11025 dots across the screen. If the clock has been adjusted to, say, 11030 Hz, then 11030 dots are painted across the screen. If the configured clock value matches the sound card's actual sample rate, the ticks will be in exactly the same place on each line, gradually painting two lines of ticks vertically on the screen. However, if the clock value is incorrect, the tick lines will slant one way or the other.

1. Click the wrench icon **Setup** ⇨ **Misc** tab.
2. Click <**Adj**> under the clock section to open the calibration display.
3. Turn off the AGC with the button at the bottom (the button should be 'up' with no shadow).
4. Tune your receiver's VFO to the standard frequency. You should hear the ticks.

Copyright © 2026 Robert C. Furzer K4CY

5. Wait about three minutes. If you are using 11025 Hz calibration, you should see two lines of marks (vertical or slanted), corresponding to the 1-second tick sound bursts transmitted by the time-standard broadcast station. You will see only one line with 8000 or 6000 Hz.
6. Right-click to move the vertical green reference line to the tick line and compare the tick line to the vertical reference.
7. Click a low tick mark (the bottom one if possible) and move the cursor to the top of the line. You will see a yellow line on the display. Move the cursor to overlay the yellow line on the tick line and click a tick mark near the top of the line. The measured clock frequency appears in the adjust window.
8. Continue watching for a few minutes to make sure that the new tick line is vertical, adjusting as necessary.
9. When you are happy, click <**OK**> to save the setting and exit the calibration function.
10. Close and restart Logger32 for the new clock value to take effect.

Here is the calibration display showing a sound card clock that is pretty close to its intended value of 11025 Hz, with just a slight slant ▼

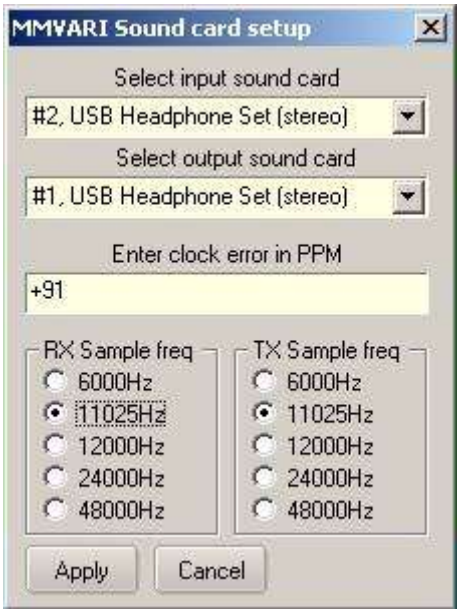


Hints:

- If the tick line is wide and fuzzy, reduce the sensitivity with the gain control at the bottom, or with the gain controls on your receiver. Try to click at the same relative place on both top and bottom ticks *e.g.* the left edges.
- If the tick line slopes markedly, you may need to repeat the calibration, getting closer each time.
- If the time standard station broadcasts a strong carrier along with the ticks making it hard to see the tick line, try notching out the steady tone.

Copyright © 2026 Robert C. Furzer K4CY

21.13.4 MMVARI calibration procedure



Complete the MMVARI RTTY calibration procedure for MMVARI to determine the sound card clock error in **P**arts **P**er **M**illion, then enter it into the MMVARI Sound Card setup form ▶

The PPM adjustment value must be in the range of +/- 9999 Hz. Positive values make the clock go faster, negative values slow it down.

21.14 Digital signal quality

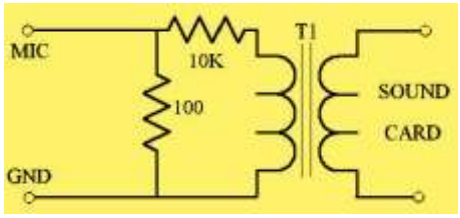
Any nonlinearities in the *analog* parts of the 'communications system' comprising the transmitter, ionosphere and receiver, lead to distortion that may cause errors in the bits and bytes of digital data, which may in turn prevent good copy (depending on their severity and any digital error detection and correction schemes in use). Furthermore, severe nonlinearity typically created by over-driving/flat-topping in any part of the transmitter (including the computer sound card or terminal unit) tends to broaden the transmitted signal's bandwidth, potentially causing interference to other radio users (especially locals but potentially worldwide with good propagation). All in all, nonlinearity is bad news, hence the value of a cautious approach that errs on the side of under- rather than over-driving all stages.

Here are some tips to avoid being 'that station' who causes grief for other radio spectrum users:

- Use **F**requency **S**hift **K**eying rather than **A**udio **F**requency **S**hift **K**eying for RTTY, if possible. FSK electronically switches a carrier oscillator between the mark and space frequencies, bypassing the audio modulation (and associated problems) used for AFSK.
- Disconnect or disable your microphone when transmitting digital signals:
  - If your radio has LINE level input/output, that may bypass the highly-amplified MIKE input *and* provide a constant gain level more suited to a PC audio card's typical audio voltages.
  - If your radio only has a MIKE input, a simple fixed resistor attenuator/voltage divider can get around the problem of needing to choose between the lowest one or two PC output and microphone input settings.
- Turn off the speech compressor or other audio shaping/filtering in your transmitter. Some radios do this automatically in DATA mode, otherwise use the button, radio menu setting or CAT command.
- Take it easy on your **A**utomatic **L**evel **C**ontrol. Set the PC audio output and transmitter audio input gains carefully such that there is barely if any ALC activity, and preferably monitor the ALC level routinely in case the settings change – which may happen unintentionally due to Windows updates, software bugs or hardware issues (such as RF interference caused by your own transmissions, and power glitches).
- Aim to 'balance' the gains of all the transmission stages, such that none of them is working flat-out. Pay particular attention to the audio stages, since any distortion generated there will be amplified by subsequent stages.

Copyright © 2026 Robert C. Furzer K4CY

- Take care over earthing/grounding and shielding to avoid introducing hum and other problems:
  - Invest a little in good quality, well-shielded audio and serial/USB leads and connectors.
  - Galvanic isolation between the PC and radio using audio transformers, opto-isolators for PTT/FSK lines and *ferrite* clamp-on toroids/beads can further reduce hum, RF feedback *etc*.
  - Here is a simple schematic diagram for a 100:1 voltage divider plus a 1:1 audio transformer ▶



- If you can, it is worth monitoring your transmission using the transceiver's monitor audio, at least occasionally or whenever adjusting the settings. Doing so might alert you to hum, or the following issue …
- Either mute all other PC audio sources apart from the soft modem (*e.g.* MMTTY or MMVARI) while using digital modes, or dedicate one sound card *exclusively* for this purpose. In particular, be sure that various audio chimes and bleeps used for warnings and alerts in Windows plus other software (including Logger32!) are not transmitted over the air[363].
- If possible, transmit high tones towards[364] the righthand (HF) side of the waterfall/spectrum display, rather than low tones towards the left. Thus, any audio harmonics are more likely to be attenuated by the filtering in your transmitter.
- Keep your power down. Modern digital modes are generally much more sensitive than analog/legacy modes, hence even virtually inaudible signals on noisy paths can often give partial or complete copy. Don't be an alligator[365]: if you can't copy weaker stations calling you despite receiving strong signal reports from others, you should probably turn down the wick.

21.14.1 Bad received signals

The presence of extra lines (mixing products) around someone's signal suggests that he *may* be overmodulating and transmitting a dirty signal, but a strong, clean signal can cause similar symptoms due to receiver overload.

The surest way to determine whether the signal is clean or dirty is to use the IMD indication in the fourth pane from the left in the status bar, also shown at the bottom left of the RX Tuning window. It will turn green at about -23 dB of IMD, but the distortion is negligible at -20 dB.

If you see a strong signal that is generating extra bars on the display, but whose IMD is good, then the distortion is most likely caused by your receiver. Use your crystal filters or passband tuning to reduce the strength of the station by moving him out of the receiver's passband, or reduce the RF gain, increase attenuation or turn your beam away to reduce the signals hammering your front end and mixer. Logger32's built-in DSP does a good job of copying through such distortion products, and if the strong station is the one you want to work, the DSP ignores the bars outside the main RX passband lines.

---

[363] Although you don't *need* to listen to the screeching digital signals, if you do so you may notice those Windows chimes, coughs, yapping dogs and other background audio from the shacks of ops that don't pay sufficient attention to these tips. It's a useful prompt to double-check your own setup.

[364] Don't go *too* far HF though, or the higher of your tones may be attenuated by the very same roll-off that reduces unwanted harmonics!

[365] Big mouth, small ears. Much better to be an elephant, meercat or hawk.

Copyright © 2026 Robert C. Furzer K4CY

21.15 2Tone

Logger32 supports G3YYD's 2Tone RTTY engine (software modem).

2Tone can be run in parallel with MMTTY or MMVARI, allowing simultaneous RTTY signal display using two different decoding schemes. The 2Tone decoder is suspended when MMTTY or MMVARI transmit to dedicate the CPU to generating well-timed RTTY signals. The 2Tone decoder is only available in MMVARI while in the RTTY mode.

2Tone and MMTTY are tightly coupled. Activating Squelch, Net, or AFC from either 2Tone or MMTTY changes the setting on the other. Clicking or right-clicking the MMTTY spectrum or waterfall moves 2Tone to the same frequencies as MMTTY.

Transmitting with 2Tone is only available while using MMTTY. It can be configured for AFSK or FSK. The 2Tone decoder is also available in the SO2R and SO2V windows. A copy of *2Tone.exe* needs to be placed in the Logger32 MMTTY2 folders.

Operational note: disable your radio's RTTY twin peak filtering when using 2Tone. The 2Tone decoder uses very narrow twin peak filters. If the filters are slightly offset, it can cause clipping/distortion of the incoming signal, reducing performance.

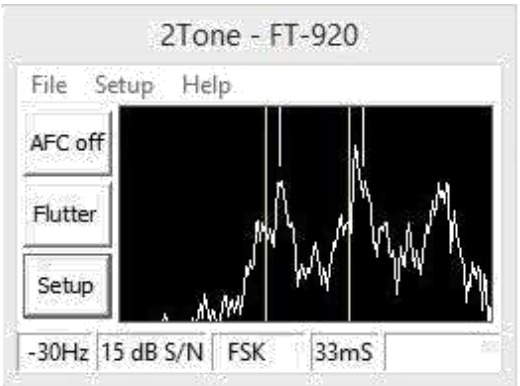
21.15.1 2Tone installation and setup

The current version/latest release of 2Tone can be downloaded from the 2Tone downloads page. Install it following the brief instructions in *2Tone.pdf* from the downloaded *.zip* file.

To open the program, click <**2T**> on the MMTTY or MMVARI main, SO2R or SO2V menus ▼



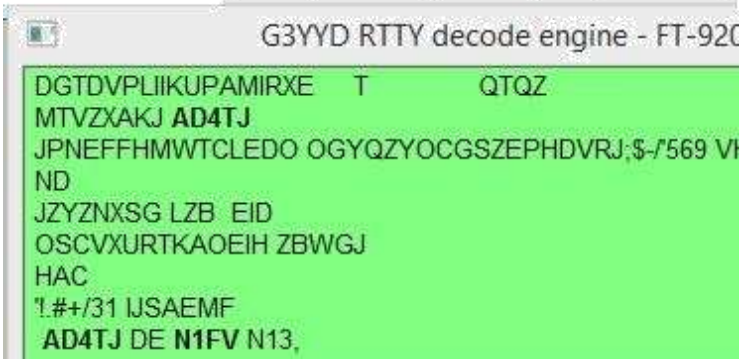
2Tone opens with the Decode and Spectrum/Setup panes shown ▶



Click <**Setup**> to select the sound card and RTTY decode parameters, then click <**OK**> to save the settings.

Set the 2Tone mark and space settings to coincide with the default audio settings in this example:

- 2Tone Mark/Space: MMTTY / MMVARI Default Audio.
- Low 1275/1445 1360.
- High 2125/2295 2210.



With the above settings, triggering the $**Align**$ function in MMTTY or MMVARI with a macro, or right-clicking the waterfall, puts both programs at the same frequency for simultaneous decoding.

Copyright © 2026 Robert C. Furzer K4CY



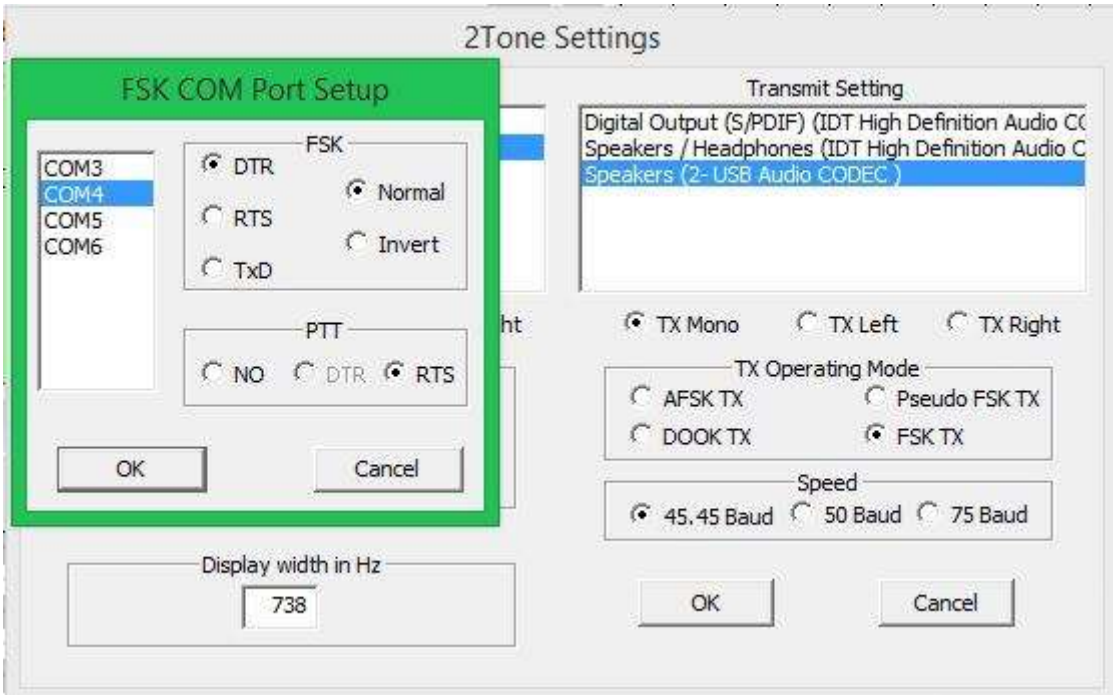
◀ In 2Tone, open the setup window and select your sound card/s for both receive and transmit if you plan on using AFSK.

Selecting <**FSK TX**> lets you choose the COM port plus FSK and PTT keying lines. PTT is optional depending on hardware configuration.

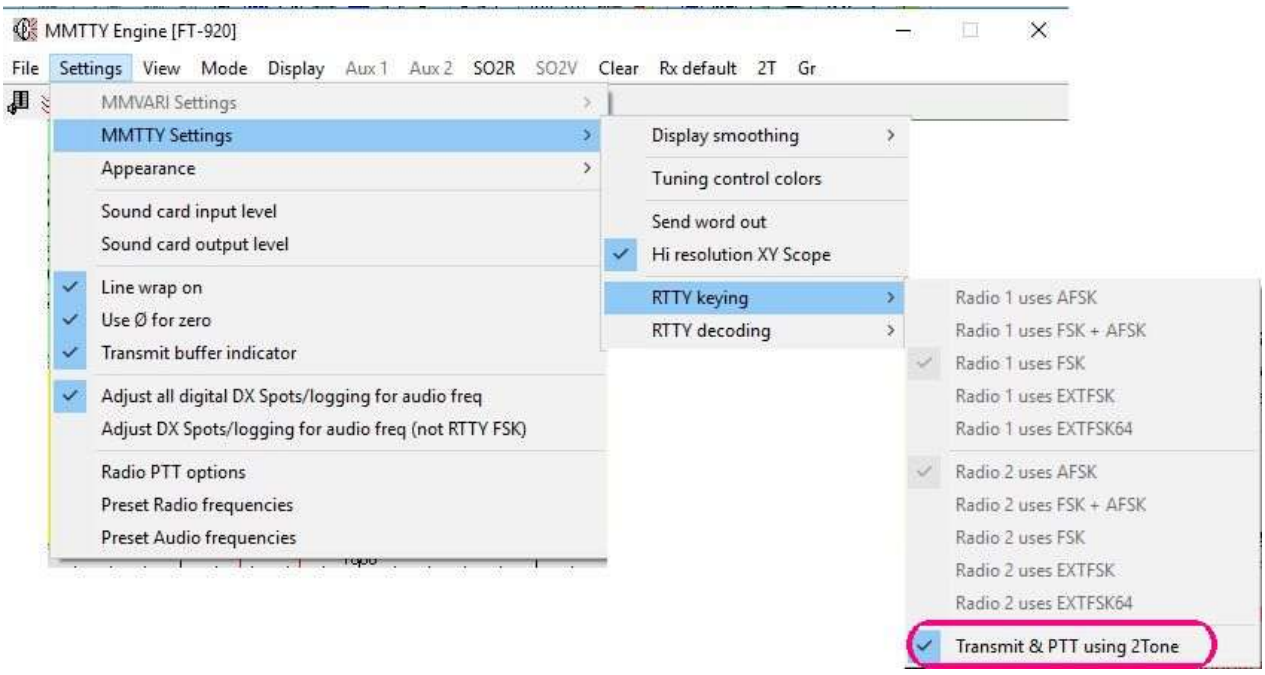
If you plan on transmitting FSK with 2Tone, you cannot use the same COM port as MMTTY since MMTTY opens first and seizes full control of the port. Set the MMTTY "TX" FSK port to <**None**> to avoid a conflict.

2Tone does not support EXTFSK, however it should work with a USB-RS232 adapter.

Here is a typical FSK transmit setup in 2Tone ▶

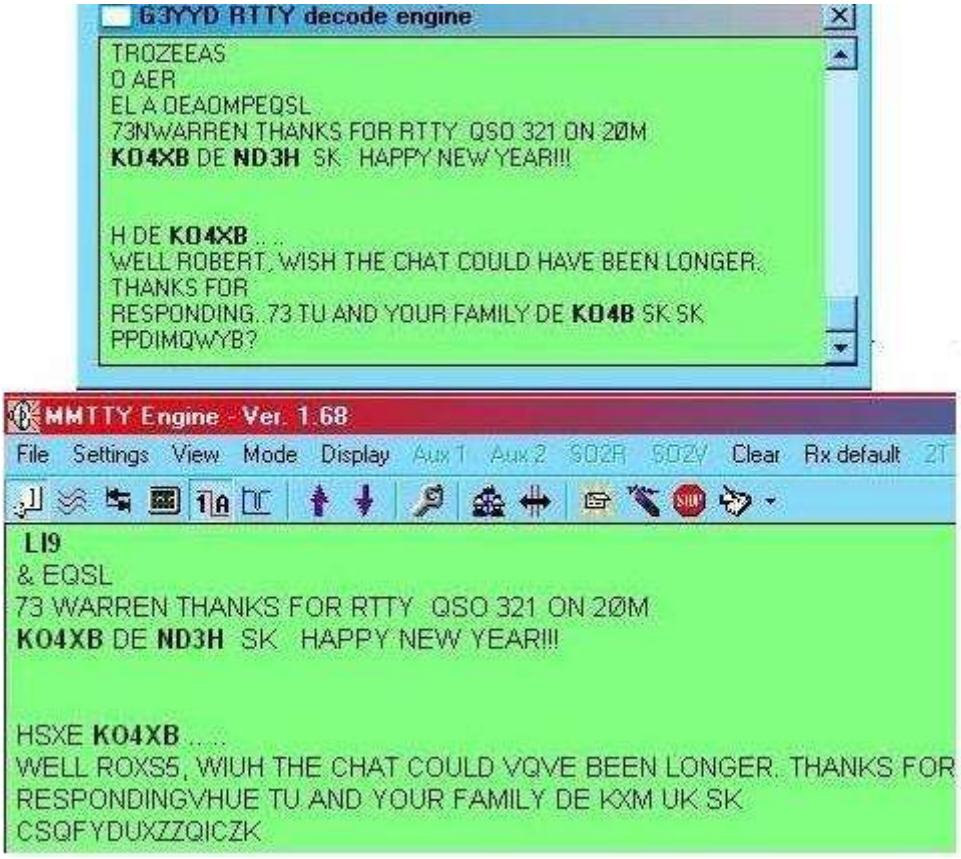


Copyright © 2026 Robert C. Furzer K4CY



The setup/spectrum pane can be re-opened again by clicking the 2Tone box in the Windows task bar. The Spectrum display pane may be left running during operation.

This shows the 2Tone decoder running in parallel with MMTTY.



Notice the improved copy available in 2Tone compared to MMTTY on this signal.

21.15.2 2Tone operating

The Top button toggles AFC on ⇄ AFC off. N1MM+ also controls this button.

Copyright © 2026 Robert C. Furzer K4CY

The Middle button optimizes the decoder for various propagation/signal characteristics:

The Setup button displays the 2Tone settings dialogue:

- **FSK TX in TX Operating mode** grouping displays the FSK COM Port Setup dialogue and space fading at the same time.
- **Flat** is designed to help copy in slow QSB, where mark and space tones fade and recover together.
- **Flutter** is designed to improve copy under very rapid fading conditions.
- **Selectiv(e)** is for when the mark and space signals fade independently of each other, as is quite common on the lower HF bands.
- **Spread** can help with the signal spectrum is spread over a wider bandwidth by polar paths or auroral conditions, typically on the higher HF bands when the sun is active.

21.15.3 The FFT display

Tune the radio until the RTTY twin peaks and the short yellow (or white with AFC off) vertical tuning indicators are near the vertical yellow lines. The long vertical lines show the set tone frequencies. The displayed bandwidth can be adjusted, click Setup button, between 449 and 1292Hz at 45.45 baud centered on the tone pair. Bandwidth at 75 bauds is 741 to 2133 Hz. It has a 60dB dynamic range and updates every 85 milliseconds at 45.45 baud. Unlike MMTTY FFT, the display frequency range is *not* changed by AFC.

Squelch: on the Setup menu, clicking Squelch toggles it on/off. Characters are displayed when the signal to noise is 1.5dB or higher. An occasional character will be displayed with a noise input. It takes a while after a signal goes before the output is squelched - this is deliberate to ensure nothing is missed in any replies. When squelched, 2Tone stores the last 4 characters. On squelch opening these characters are sent to the DI followed by newly received characters.

AFC (**A**utomatic **F**requency **C**ontrol) gives the best decode performance by locking onto a signal within 60 Hz of the set tones, even a RTTY signal that is well below the decode threshold.

Status bar: the numeric display along the bottom of the 2Tone window - left-hand side shows the direction and value in Hz to tune the receiver to be exactly on frequency. In practice the receiver should be tuned to within 30 Hz of the given signal. The short vertical lines at the top of the FFT display shows where the AFC is compared to the set mark and space frequencies. AFC on is yellow while AFC off is White with the decoder tone frequencies unchanged from the set values. When transmitting 2Tone stops updating AFC preserving the AFC value for receive.

Signal to Noise ratio, next to the AFC value in the status bar is the measured signal to noise ratio of the decoded signal in dB. This is an average of S/N measured over several RTTY characters. The threshold to keep in mind is 10dB. Above this, the error rate will be reasonable and as it drops below 10dB decode errors will increase very rapidly. Sometimes a seemingly strong signal has a poor signal to noise ratio due to some types of signal propagation. It is a good indicator of why a strong signal has poor decode. A poor quality transmit signal can also have a degraded signal to noise ratio.

Net: Net **On** moves the transmit frequency to match the received tones. Net **Off** transmits at a fixed tone (either the default or where it was when Net was turned **Off**).

Baud Rate: can be selected in Set up dialogue for 45.45 (normal standard for amateur RTTY), 50 or 75.

Copyright © 2026 Robert C. Furzer K4CY

Stop Bit Length: on the status bar fourth in from the left is a display of the received stop bit length. Ideally this should be around 32 to 34 milliseconds although some RTTY software departs from the 1.5 stop bit standard. Values around 44 and 48ms are common in contest set ups, but do increase decode error rate. At low signal to noise ratios, this value will move either side of the transmitted value. 2Tone transmit uses the 1.5 stop bit standard.

Receiver AGC: should be set to slow as fast will confuse the selective fade algorithms and cause AGC generated IMD. Normal SSB AGC setting is good but avoid the CW fast setting.

Carrier QRM: tuning in a carrier without any modulation and little QSB will result in no output from 2Tone, the AFC and S/N will not update. If a RTTY signal is QRM'd by a carrier on one of the tones, 2Tone will often decode the signal correctly even when it is weaker than the carrier … but do not expect miracles!

TX Sound Card PTT: clicking AFSK, DOOK or pFSK opens a dialogue for COM port PTT. If VOX or other PTT (*e.g.* Logger32) is used, click no, otherwise select the COM port and output line required.

TX AFSK: AFSK transmit audio is preferred as the audio amplitude is constant and with Net **On** will track the received signal frequency. Ensure the audio input of the transmitter is not overdriven. Use of tone frequencies above 1500Hz is preferred so that audio harmonics are attenuated by the radio's SSB transmit filters.

TX DOOK (**D**ifferential **O**n/**O**ff **K**eying): the signal waveform varies in amplitude and both tones are present at the same time during transition from one tone to the other. This requires the use of a linear transmitter as intermodulation products will broaden the signal width. Again use frequencies above 1500 Hz.

TX pFSK (**p**seudo **FSK**): generates a 5 kHz keyed tone on both left and right channels of the selected sound card. The tone is on for space and off for mark. A suitable detector circuit is required, its output feeding the FSK input of the radio.

Search & Pounce: as the transmit frequency does not change with AFC, adjust your radio/s XIT control so the small vertical AFC lines are co-incident with, or very close to, the long vertical lines. Not doing so will result in off-frequency transmission.

TX FSK: makes use of a COM port's DTR, RTS or TxD lines to key a radio's FSK input used via a keying transistor. Normal keying is -12v for Mark and +12v for Space with Invert being the other way round. Open Setup and click FSK TX in the TX Operating mode group to open a dialogue for setting Com Port, DTR or RTS and Normal or Invert shift sense. The COM port is released during reception so another copy of 2Tone can use it for transmission. It is possible to use one of the DTR/RTS lines with Di1 and the other for Di2 for easy SO2R operating. Careful programming results in low levels of timing jitter. A CPU with 2 or more processors will minimize timing jitter. Excessive timing jitter will cause received character errors.

TX PTT: only works on FSK and is selected when FSK is setup. -12v PTT off, +12v PTT on.

RX Sound Card: select the correct input and tune in a strong RTTY signal. Adjust the receiver and sound card gains so the peaks on the FFT (spectrum display) are about 90% of the maximum height of the display. The FFT has a degree of AGC (**A**utomatic **G**ain **C**ontrol) so that the gain settings are not critical.

Display: 2Tone checks on startup that it is within the display area of your monitor(s), if it is not it will automatically center itself on the nearest monitor. If the 2Tone window is not visible it is because it is either behind another window on the display or has been minimized. Going to Menu Setup, Topmost on Checking this item will put 2Tone always on top.

Copyright © 2026 Robert C. Furzer K4CY

SO2R: more than one 2Tone can be used at the same time by using separate folders, each with a copy of *2Tone.exe*. By using one 2Tone in the Main MMTTY window configure the Setup dialogue to use Left "line in" and Left "line out" for the main radio (Radio 1) with another 2Tone for the MMTTY SO2R window with Right "line in" and Right "line out" for the SO2R (Radio 2) radio. This is in conjunction with wiring the Radio 1 to the left line in/out and SO2R Radio 2 with the right line in/out. See also notes under TX FSK above.

Play WAV file: reads in a sound file that has been saved via the 2Tone command (see below) and plays it at full PC speed through 2Tone. Just select the file to play and open it. Clear the Di window before playing. Use it for post contest analysis. Will only play 2Tone saved files.

Save Text: decoded text can be saved to a file using **File ⇨ Save Text** (a toggle: click it again to stop saving the decoded text, or close 2Tone: it defaults to off). The sound card on transmit is set for 16 bits at 12,000 samples per second at 45.45 baud and proportionally faster for 50 and 75 baud. The receive sample rate is 48,000 for all baud rates. On receive and transmit sound card channels are chosen in 2Tone setup for mono, left or right.

2Tone has been optimized to work with 45.45, 50 and 75 baud RTTY and has no other baud settings. Transmit is set for one start bit, 5 data bits and 1.5 bit stop length it timing accuracy is as accurate as the sound card sampling rate for AFSK, DOOK and pFSK. FSK timing accuracy is as accurate as the CPU crystal plus some unavoidable jitter caused by Windows operating system.

The receive tone filters used for Flat, Flutter and Selective use 2nd order Nyquist filters bandwidth of baud rate in Hz. Spread decoder uses a raise cosine Beta=0.5 with twice baud rate bandwidth. Flutter and Spread decoders also use post detection filtering with a bandwidth that varies with the measured signal to noise ratio.

The Selective decoder processes both tones separately using a threshold value derived from signal and noise amplitudes averages and then combined with the current tone amplitude. The individual mark and space tone values are combined to produce a final mark or space result for asynchronous decoding. Unlike standard FSK mark/space determination this system makes use of the absence of signal as well as the presence of signal and by treating the two tones separately and combining them together single tone copy is possible. This is sometimes called in-band diversity.

On transmit AFSK tapered cosine filtering, known as Tukey1 windowing, of the keying waveform is used to minimize occupied bandwidth. The filtered keying waveform frequency modulates a carrier tone generating an FSK waveform with no amplitude variation.

Transmit DOOK uses raised cosine low pass filtering of the keying waveform that then amplitude modulates the mark tone and an inverse keying waveform modulates the space tone. During Mark/Space transitions, both tones are present at the same time and the amplitude of the transmitted waveform varies, which will result in some broadening of the transmission due to transmitter IMD products https://en.wikipedia.org/wiki/Window_function#Tukey_window.

While the pseudo FSK mode outputs a 5 kHz tone with tone on for space, the transmit start up sequence is one character length (165 ms at 45.45 baud) of space tone followed by one character length of mark tone and then a shift character appropriate to the following printable character. The startup sequence has been optimized for 2Tone type decoders (FLdigi RTTY mode uses a similar technique) and is also suitable for MMTTY type decoders. All transmit characters are one start bit, 5 data bits and 1.5 bit stop length.

Configuration settings of 2Tone are stored in *2Tone.ini* in the same folder. If you delete this file, programmed defaults will be used until they are changed.

Copyright © 2026 Robert C. Furzer K4CY

Read more technical information on RTTY modulators and the history of technical developments at w7ay.net/site/Technical/RTTY%20Demodulators/

21.16 GRITTY

The GRITTY decoder for RTTY may be used with Logger32 to supplement MMTTY and MMVARI.

Download GRITTY from www.dxatlas.com/GRITTY and install it in the default location.

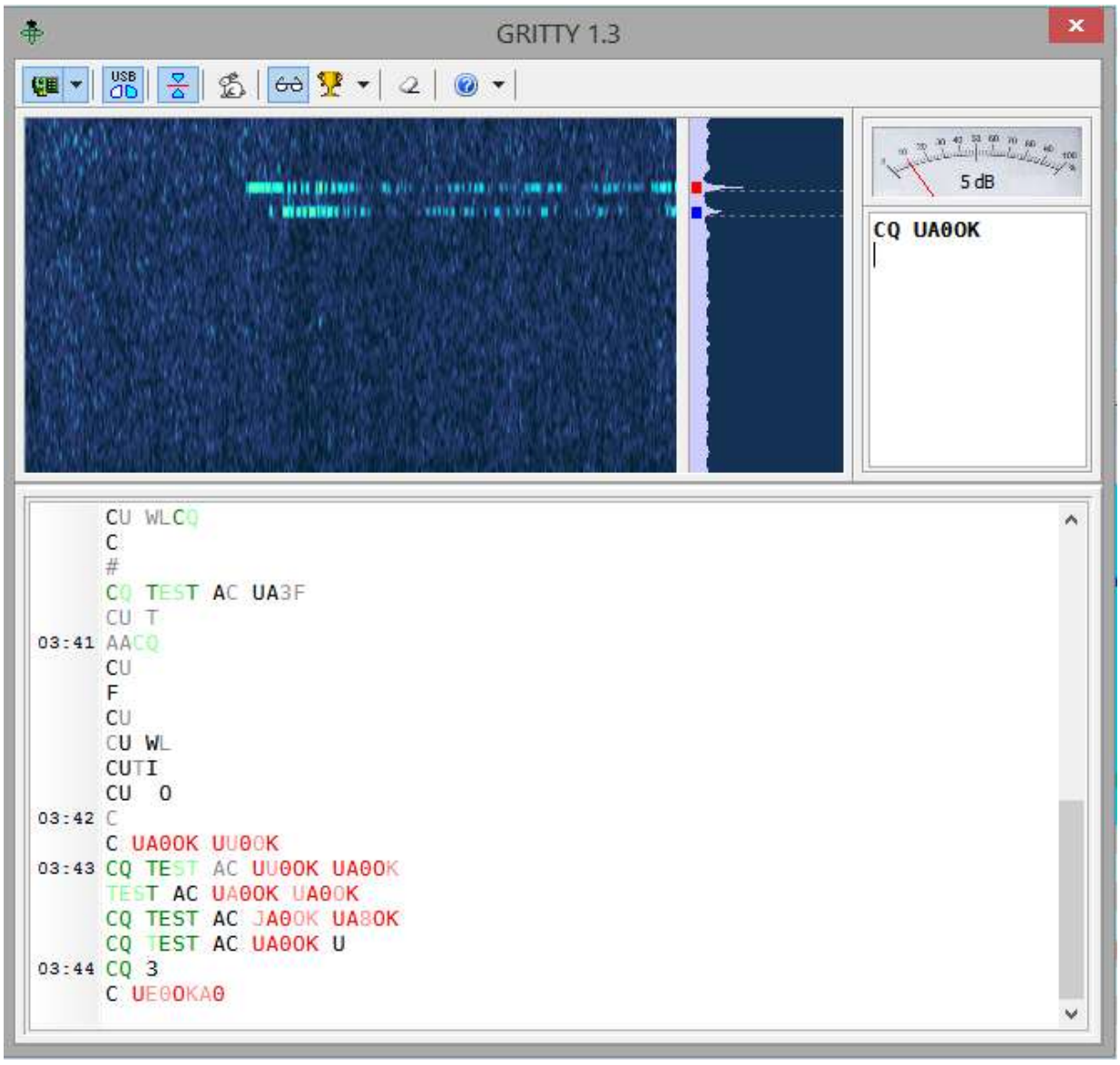
Click the "?" button and at least glance through the help before proceeding.

21.16.1 Operating GRITTY

With the sound card open in RTTY mode, <**Gr**> on the menu opens GRITTY ▼



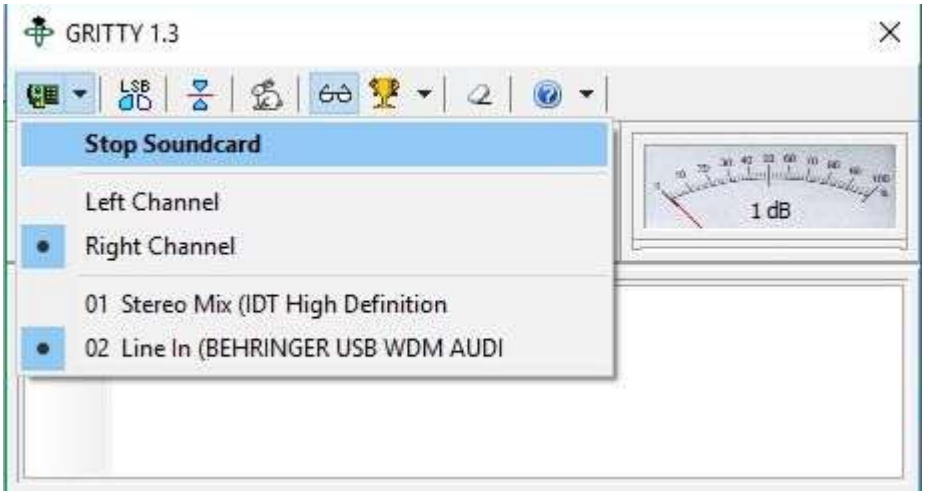
GRITTY opens with a horizontal waterfall and decode pane ▼



Copyright © 2026 Robert C. Furzer K4CY

- **Sound card setup**: click the down-arrow beside the first icon to select the desired sound card and channel ▶▼





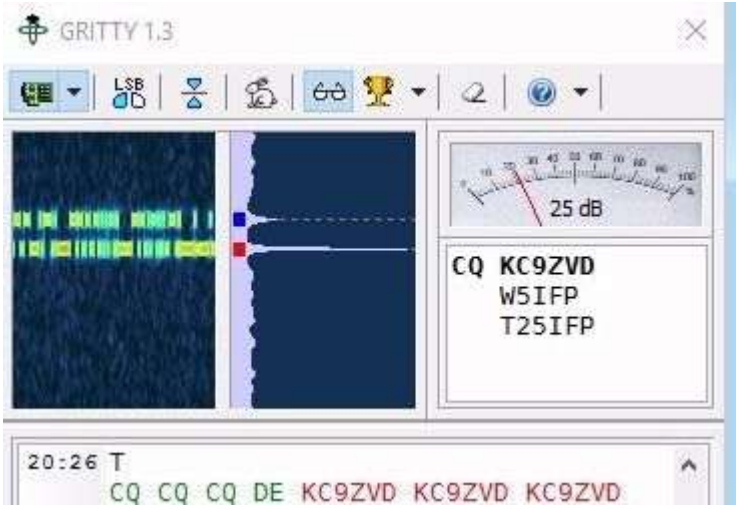
GRITTY defaults to high tones (mark 2125 and space 2295). If you use low tones, change GRITTY to be compatible with the setting in MMTTY/MMVARI by modifying the *GRITTY.ini* file. See the GRITTY help for details.

Note: Logger32's default audio frequency setting is halfway between mark and space, whereas GRITTY defaults to the highest tone frequency.

21.16.2 GRITTY colors

Although the colors of the highlighted text and background of the default GRITTY decode pane are fixed, you can change the text and background colors of Logger32's GRITTY decode pane using **Settings** ⇨ **Appearance**. The same settings apply to MMTTY/MMVARI.

21.16.3 Using GRITTY



GRITTY's decode pane at the bottom is similar to the MMTTY and MMVARI receive pane ▶

Just below the SNR meter is a pane listing callsigns copied.

Callsigns are picked out in bold. Click a decoded callsign to transfer it to the log entry pane.

Copyright © 2026 Robert C. Furzer K4CY

21.16.4 Scrolling

While scrolling, “Freeze…” appears in the caption ▶

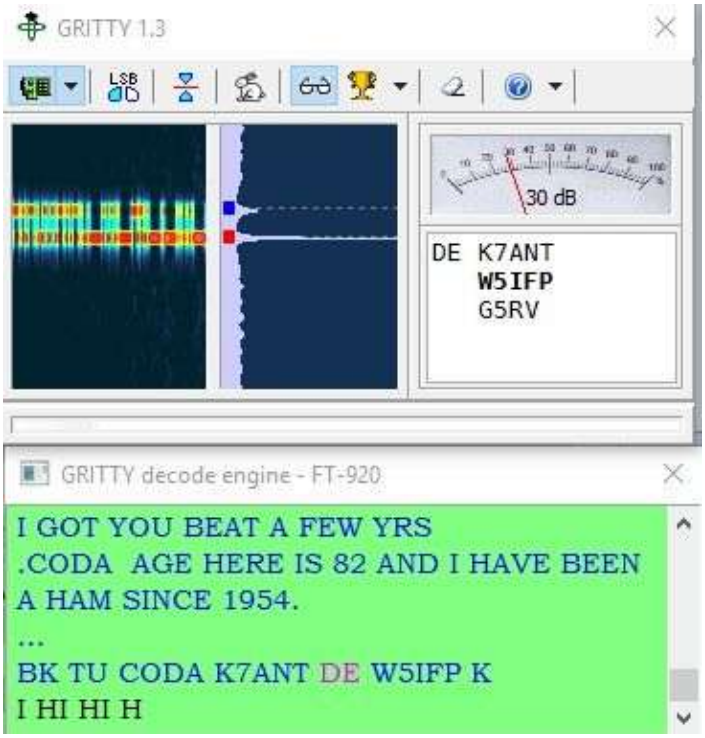


Signal decoding continues but is not displayed. Freeze times-out and self-cancels if no further scrolling is detected for a while. If the cursor is moved away from the receive pane, freeze mode is immediately canceled and incoming text printing is resumed.

21.16.5 Screen position

The GRITTY application has a default decode pane and a Logger32 decode pane. To conserve screen space and eliminate duplication, the default decode pane may be minimized by sizing from the bottom up and placing the remaining decode pane below the waterfall.

Alternatively, move either pane to another monitor if you have more than one.



21.16.6 Signal alignment

GRITTY decoding is fixed at the audio default tones. In order to decode a signal, it should be aligned to the default position either by tuning the VFO or right-clicking the signal on the waterfall/spectrum, aligning the received and transmitted signals.

21.17 Sound card digimode FAQs

Q. Why are right-click auto fill data not showing up in my log?

A. The sound card receive window can transfer information into the log entry pane. Simply by right-clicking text in the receive window, you are presented with a list of fields to choose from, and the selected data is sent to that field (see Sound Card Data window for more on this). For this feature to work, the field you are selecting data for in the right-click list must be one of the fields in the log entry pane for the data to transfer properly.

Q. Why does Logger32 not work with the sound system built-in to my PC motherboard?

A. First check the Windows sound mixer settings. To be honest, it is best to install a separate sound card or USB sound system for digital modes anyway.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why do my macros do not work properly and consistently?

A. Do the macro names have $ before and after (*e.g.* $**MyCall**$)? Carefully check the spelling of the macros, especially longer ones such as $**UpperOrLower**$[366]. If a macro using $**Receive**$ fails to transmit text before going to receive, it may help to add a few carriage returns to the beginning of the macro.

Q. Why don't special characters appear using Alt + ASCII codes?

A. Hold down <**Alt**> while you tap out the three-digit ASCII code on the *numeric* keyboard, not the main alphabetic keyboard. Don't skip any leading zeros. See macros for more.

Q. How come when I type, the mouse cursor disappears?

A. Open your mouse control software panel to see if you have <**Hide cursor when typing**> enabled. If so, disable it. This is common on laptops with a touchpad on the keyboard. A separate USB mouse might help.

Q. Logger32 does not show any signals on the spectrum display. What have I broken?

A. On the Sound card data window menu, click **Settings ⇨ Sound card input level** and make sure that you (or any software that takes control of the audio) have not muted a control or set the level too low. Check that your cat did not disconnect the audio lead connecting the radio to the computer … and, oh yes, check that the radio is actually switched on with an antenna attached!

Q. Why doesn't the XY scope display appear in RTTY?

A. Turn the XY scope on by clicking **View ⇨ Show RTTY XY Scope**. If it is enabled but invisible, try **View ⇨ Find lost windows**.

Q. Where has my received text gone?

A. When you use the pause function, the RX pane background turns white. If you have a white font, the text disappears. It's still there, just invisible. Adjust the RX window background color while paused using the Sound card data window menu **Settings ⇨ Appearance ⇨ Rx window pause background**. Even a light gray may give just enough contrast to read white text.

Alternatively, change the text color.

Or don't use pause.

Q. I've lost a display that I used to see. How can I retrieve it?

A. On the Sound card data window menu, click <**View**> and select the relevant item.

---

[366] Macros are not case-sensitive though. We use initial capitals in the manual for readability: you're welcome to do the same in your macros, or not. See if we care.

Copyright © 2026 Robert C. Furzer K4CY

Q. How do I transmit a CWID?

A. To send a CW identification at the end of your transmission, use the $**CWid**$ macro with your callsign. Change the CW sending speed using **Settings ⇨ MMVARI Settings ⇨ CW ID** and choose between 9, 12, 19 and 37 WPM.

Q. Why don't the align (frequency change) or QSY features (right-clicking the target signal) work?

A. You *may* be using a radio in which this feature only works with one of the VFOs. Switch VFOs – or radios – and try again.

Q. Why are people complaining about my audio being distorted?

A. Disconnect the microphone from your radio, or disable it, or turn the mike gain right down, when you operate digimodes. Disconnect your computer sound card from the audio input to the radio when you operate phone. Alternatively, buy a radio that disables the microphone when in DATA mode, and disables the LINE IN/OUT when in voice modes. Check your audio levels on transmit are not over-driving the radio.

Q. Why does my transmit power vary between QSOs?

A. Your radio has both transmit and receive passbands. If your transmit audio begins to roll off at about 2500 Hz, you will find your transmitted signal to be weaker if you transmit and receive at, say, 2800 Hz, even though on receive you may be able to copy a signal at that frequency. See here for more.

Q. Using VOX, why does my rig not stay on receive after a transmission? The transmitter cycles on and off!

A. Open the audio mixer and turn off any unused inputs to the sound card. If you are using line input, mute the microphone input or turn it down to zero (although it may turn off your speaker sound).

Q. When I try to get Logger32 to copy a signal, what makes it jump to a stronger signal nearby?

A. This is the action of the AFC. Toggle AFC on ⇄ AFC off by clicking the AFC panel in the status bar. It can be useful if copy deteriorates due to frequency drift, or if a station deliberately slides away from an adjacent signal causing QRM.

When the other station turns it over, Logger32 changes frequency. This is also AFC action.

Q. I lost the tuning display or Aux window. Where did it go?

A. If you changed screen resolution, it might be off your screen. Go back to the old screen resolution, gather your windows to the middle of your screen, and then reconfigure back to the new resolution. Some windows may be hidden behind other windows. If a window disappears, move other windows to see if it is hidden behind them. If all else fails, try **View ⇨ Find lost windows**, close and re=open Logger32, or reboot.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why is my RX screen empty?

A. Assuming you are looking at text of a color that contrasts with the background, here are some things to check:

- Are you monitoring an active frequency with decodable signals? Are you tuned to one of them?
- Are you using the correct mode?
- Do you have the receive levels set right?
- Is your radio on and connected? What about the antenna for this band?
- Has a solar flare wiped out the ionosphere? Can you hear *any* signals on or around this frequency?

Q. What messes up the waterfall display after it has been working properly for a while?

A. Some waterfall problems appear to be related to the use of power management (Windows may automatically turn off the display after a period of inactivity) or screensavers. Try disabling power management and/or the screensaver in Windows.

If there are problems with the way text prints (characters appear slowly, out of order, or the wrong characters) or other video problems, that may be a hardware issue with your video card. Try reducing hardware acceleration using Windows **Start** ⇨ **Control Panel** ⇨ **Display** ⇨ **Settings** ⇨ **Advanced** ⇨ **Performance** tab, moving the hardware acceleration slider to the left, one position at a time.

Q. How come I can run PSK OK, but Logger32 goes slow or freezes in RTTY and FT#?

A. You may be running out of computer resources with the extra load from MMTTY or JTDX|WSJT-X software. Try running only the programs you need, ideally just Logger32 plus the digimode software.

Q. Why is my transmit text all lowercase or CAPITALS in PSK? I thought PSK allowed mixed case!

A. It does … if configured accordingly. On the Sound card data window menu click **Settings** ⇨ **MMVARI Settings** ⇨ **Typing preferences** and choose mixed case (as typed).

Q. When I run *MMTTY.EXE* from the Logger32 folder, why do I get error messages?

A. Not all the MMTTY files have been loaded into this folder. If you are running MMTTY purely to check/change some of its parameters, ignore the error messages and hit <**Enter**> until you reach the expected MMTTY opening screen.

Q. What makes my radio transmit when the PC is booting?

A. This is a known problem with serial ports where the modem control line you have configured and connected to your radio's PTT input is grounded by Windows during the boot sequence.

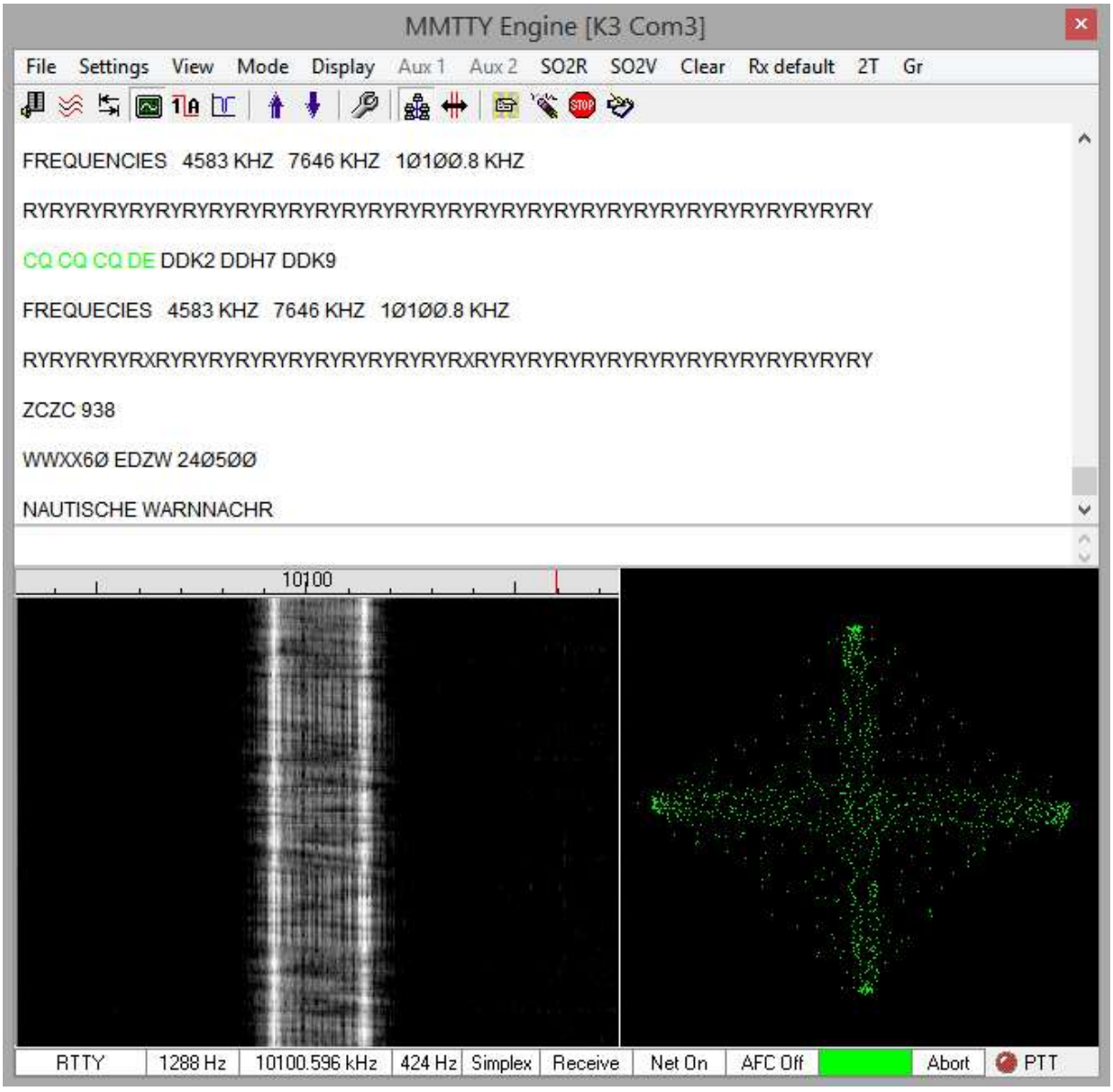
Copyright © 2026 Robert C. Furzer K4CY

You can simply wait to turn on the radio until the PC has booted, or disable its transmitter (*e.g.* enable TEST mode by holding the right side of the MODE button on a K3), or temporarily disconnect the serial port connector during the boot sequence.

Q. I can't find any RTTY signals on-air to test my configuration. What can I do?

A. Try monitoring DDK2/DDH7/DDK9, the longstanding German QRO commercial meteorological station just above the lower edge of the 30m amateur band (a useful band-edge marker!). It transmits weather reports around the clock using 50 baud Baudot RTTY with a ~450 Hz shift (MMTTY's 425 Hz shift setting is close enough), and can be received around the globe for many hours a day.

The screenshot below was taken in New Zealand at 05z using a simple quarter wave ground plane antenna, with a signal strength of S7 on my K3 ▼ I'm getting near-perfect copy, using USB on the radio and normal (not reverse) shift in MMTTY.



The messages vary in format but look out for recognizable German and English words, latitudes and longitudes, plus the classic RYRYRYRYYRYRY RTTY tuning/holding sequence (chosen because of the alternating mark-space sequences).

Copyright © 2026 Robert C. Furzer K4CY

There is greater amateur activity on RTTY at weekends. Listen around 14080 kHz, and on most other bands tune HF of the FT# frequencies until you hear the characteristic 'diddle' of RTTY signals and see the tramlines on your waterfall display.

Another option is to use pre-recorded RTTY signals. You'll find some on the Web, or you can make your own recordings during, say, an RTTY contest.

Q. What causes a Soundcard Error 12 message?

A. Conditions that can trigger this Windows message:

- Input buffers overflowed.
- Timed-out waiting for input buffers.
- Output buffers Underflowed.
- Timed out waiting for output buffers.
- Function isn't supported.
- Error value out of range.
- Invalid flag passed.
- Invalid parameter passed.
- Card doesn't support 16-bit, 8000 Hz, mono format.

Q. Why are so many people getting my callsign wrong on RTTY?

A. Lacking the fancy error correction facilities available in more modern digital modes, reception mistakes are more common with RTTY. When the receiving station displays ('prints') your transmission, junk characters often appear immediately after your carrier drops as the receiver and sound card's ALC and squelch facilities adjust themselves to the no-signal condition.

When ending a transmission, it helps to send a trailing space and maybe a K - something to bear in mind, perhaps, when composing your RTTY macros. For example, Bob's RTTY macro that concluded with "$**Call**$ DE $**MyCall**$$**Receive**$" *may* print as "... DE K4CYA" if, by sheer chance, a noise burst at the bitter end decoded to an "A". However, if Bob had used "$**Call**$ DE $**MyCall**$ K$**Receive**$", it would have printed as "... DE K4CY KA", retaining Bob's full callsign intact. Likewise, "$**Call**$ DE $**MyCall**$ $**Receive**$" (with the trailing space before dropping back to receive) would probably have printed his callsign OK, and is a trifle shorter.

I am very happy. Time to say, thank you very much for the fantastic Logger32 team. You are doing really, really nice work. Support is matchless good!

Bernd, DL9YAJ

Copyright © 2026 Robert C. Furzer K4CY

22 Data Terminal (TNC)

> "You can have data without information, but you cannot have information without data"
>
> *Daniel Keys Moran*

Logger32's Data Terminal emulates a standard terminal program, providing an interface between your PC and a hardware **T**erminal **N**ode **C**ontroller (TNC). Logger32 supports generic standalone TNC2s and TNCs such as KAM, PK-232, PK-900 and MFJ-1278. It does not support soft modems that rely on host software to support the AX.25 packet radio protocol.

The program provides for TNC mode changes with the click of the mouse.

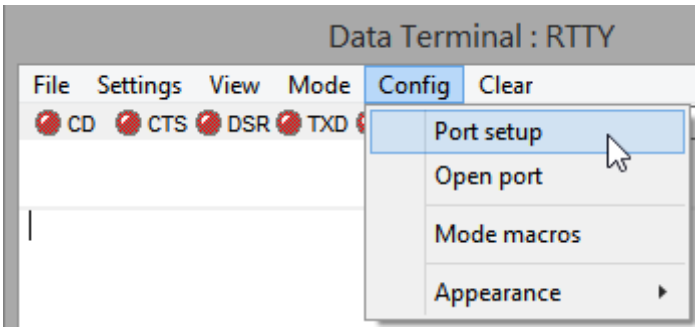
The Data Terminal functions in much the same fashion as the Sound card data window. Many of the same macros are available to make things easier for you.

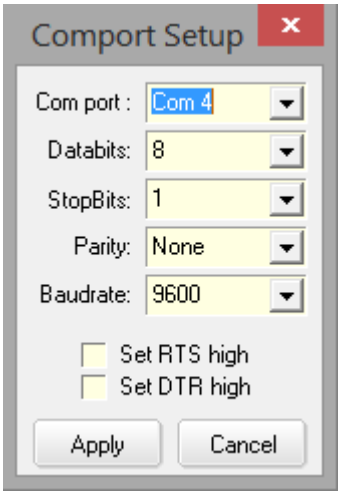
22.1 Data terminal setup



◀ Open **Data Terminal** using toolbar icon #12 vaguely reminiscent of a token ring LAN diagram.

Click **Config** ⇨ **Port setup** ▶





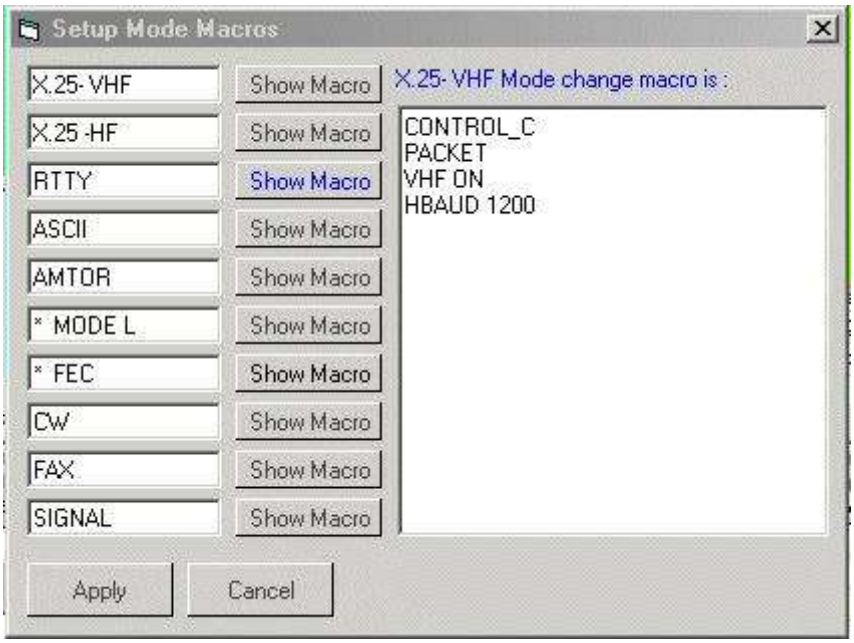
◀ Set your serial port parameters to match the TNC terminal settings, then click <**Apply**> to save the settings and close the window.

Copyright © 2026 Robert C. Furzer K4CY

22.2 Mode macros

If you have a multimode TNC, set up the mode macros. The Logger32 defaults are for the KAM: otherwise they need to be tailored for your TNC.

Select **Config** ⇨ **Mode macros** ▶



In the left column, name the modes. Then click <**Show Macro**> for any row and on the right side type the corresponding commands to put the TNC into that mode.

Finally, click <**Apply**> to save the data and close the window.

To change TNC modes, click <**Mode**> on the TNC window menu and select the desired mode to command the TNC accordingly.

22.3 Data terminal logging

You can enter data into the log entry pane in the same way as for the Sound card data window.

Clicking a callsign in the received window instantly pops it into the log entry pane Call field.

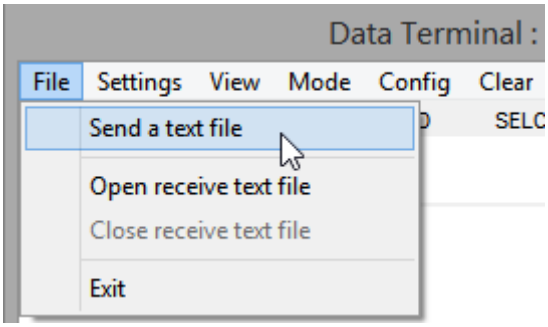
A *right*-click in the received text brings up a little menu with six options ▶ What kind of data have you selected?



22.4 File transfer

22.4.1 Send a text file

Click **File** ⇨ **Send a text file** and you are prompted for the path and name of the text file to send ▶



Click the file and <**OK**>, or simply double-click it, to send it to the TNC.

Copyright © 2026 Robert C. Furzer K4CY

This is one way of initializing all the parameters in the TNC with a setup file after its config disappears into never-never land or the battery dies.

Alternatively, you could define an “Initialize” macro that sends all the configuration parameters to the TNC as if it was changing modes.

22.4.2 Receive a text file

- Click **File ⇨ Open receive text file** and specify the path and name to save an incoming text file to disk.
- Click <**OK**> to stream incoming data from the TNC to the file.
- When done, close the file using **File ⇨ Close receive text file**.

This is an excellent way to *save* your TNC parameters when it is setup and working as you wish: open a receive text file and give the TNC the dump parameters command (*e.g.* “Disp Z” on the PK-232). Close the file, edit out the garbage at the beginning of the file and you will have a good file to initialize the TNC as mentioned above.

22.5 Data terminal macros

Logger32 supports numerous macro commands in the Data Terminal, with 48 user-programmable macro buttons per TNC mode. With up to 10 modes[367], there is capacity for 480 macro buttons.

The buttons send macro commands and text to the TNC, in the same fashion as the Sound card data window – either by clicking the buttons on the screen or using the Function keys.

Control commands are case-sensitive. The PK-232 requires the control code in CAPITALS, *e.g.* <**Ctrl**+**C**> in a command macro can be sent as either “control_C” or “CONTROL_C” (curiously, the word ‘control’ itself is *not* case sensitive!).

TNC “control_(x)” macros do not need to be enclosed with $ signs since they are interpreted as immediate commands, but other macros do require the $signs$.

22.5.1 Connect and disconnect macros

Packet connect macro:

- control_C (forces the TNC into command mode)
- C $**Call**$ (uses the callsign from the log entry pane Call field)

Packet disconnect macro:

- control_C
- D

[367] You can set up more than one mode line for the same mode, providing further sets of 48 macros for the mode if you need them.

Copyright © 2026 Robert C. Furzer K4CY

22.5.2 TNC RTTY CQ macros

The following macro shows the proper use of the 'X' and 'Control-D' TNC commands embedded in a text macro. The 'X' command forces the TNC to enter transparent mode. 'Control_D' stops transmission after the transmit buffer has been sent.

Note: 'Control_D' should be added to the end of the text stream. If the 'Control_D' is placed on a separate line, it will cause the error message CMD:"***Transmit Data Remaining" to be sent by the TNC:

- control_C
- X
- "CQ CQ CQ de $**MyCall**$ $**MyCall**$
- "CQ CQ CQ de $**MyCall**$ $**MyCall**$ K 'Control_D'

22.6 KAM+ TNC

This section provides information on setting up Logger32 to operate with the KAM+ TNC using KAM-8.0 or 8.2 firmware.

One of the Logger32 beta team ran a KAM+ in its default setting with these personal preferences:

- INTFACE TERMINAL
- CWPTT ON
- ECHO OFF
- HEADERLN ON
- USERS 5/5

With those settings, the KAM+ operates AMTOR, ASCII, CW, G-TOR, PACKET, PACTOR and RTTY.

The modes are selected from the KAM+ Terminal mode using the CONTROL_C instruction. Enter the CONTROL_C in full, in CAPITALS. The KAM+ command set, however, is not case sensitive and commands may be entered in both upper and lower case. For example, "mycall" and "MYCALL" are both acceptable.

In the text descriptions and examples below, any text below that is bracketed by a pair of $ is a macro command described in the macros chapter.

22.6.1 KAM+ AMTOR mode

The AMTOR mode selection macros are:

CONTROL_CX	For command mode
echo on	To see your transmitted text
AMTOR	To go to AMTOR stand by mode

Or

CONTROL_CX	For command mode
echo on	To see your transmitted text
AMTOR $**Call**$	To go to AMTOR calling the selected callsign

Copyright © 2026 Robert C. Furzer K4CY

The following KAM+ AMTOR instructions can be used in Logger32:

CONTROL_CA	Abort link
CONTROL_CD	Break link and stay in standby
CONTROL_CE	Returns to standby after transmit buffer is sent. Also puts +? if linked
CONTROL_CI	Inverts mark and space
CONTROL_CR	Immediately returns to receive mode regardless of TX buffer
CONTROL_CS	Selects the next mark/space shift (Hz)
CONTROL_CT	Enters transmit mode
CONTROL_CX	Returns to command mode
CONTROL_CW	Sends the Who Are You inquiry

An example AMTOR macros is:

CONTROL_CT	Enter transmit mode
Hello $**Call**$, $**Name**$ from Mike de $**MyCall**$	Send his call, name and mycall
You are 599-30-$**SerialNum**$	Send report, zone and serial number
CONTROL_CE	Return to standby

22.6.2 KAM+ ASCII mode

The ASCII mode selection macro is:

CONTROL_CX	For command mode;
echo on	To see your transmitted text
ascii	To set 110 baud ASCII (the default speed)

The following KAM+ ASCII instructions can be used in Logger32:

CONTROL_CE	Returns to standby after transmit buffer is sent. Also puts +? if linked
CONTROL_CI	Inverts mark and space
CONTROL_CR	Immediately returns to receive mode regardless of TX buffer
CONTROL_CS	Selects the next mark/space shift (Hz)
CONTROL_CT	Enters transmit mode
CONTROL_CX	Returns to command mode
CONTROL_Cn	Where n = baud speed from the following list:

1 = 45	5 = 100	9 = 300
2 = 50	6 = 110	0 = ascbaud
3 = 57	7 = 150	
4 = 75	8 = 200	

An example ASCII macro is:

CONTROL_CT	Enter transmit mode

Copyright © 2026 Robert C. Furzer K4CY

$**Call**$ de $**MyCall**$	Send his call and my call
CONTROL_CE	Return to standby

22.6.3 KAM+ CW mode

The CW mode selection macro is:

CONTROL_C	Enter command mode
Echo on	See your transmitted text
cw 15	Set CW at 15 WPM

The following KAM+ CW instructions can be used in Logger32:

CONTROL_CE	Returns to standby after transmit buffer is sent. Also puts +? if linked
CONTROL_CL	Locks transmit and receive speeds
CONTROL_CR	Immediately returns to receive mode regardless of TX buffer state
CONTROL_CT	Enters transmit mode, sending key strokes immediately
CONTROL_CU	Unlocks the speed to allow tracking
CONTROL_CX	Returns to command mode
CONTROL_Cn	Where n = CW speed from the following WPM list:

1 = 5	5 = 25	9 = 45
2 = 10	6 = 30	0 = 5
3 = 15	7 = 35	
4 = 20	8 = 40	

An example macro to call CQ:

CONTROL_CT	Enter transmit mode
CQ CQ CQ de $**MyCall**$ $**MyCall**$	Call CQ x3 followed by my call x2
CQ CQ CQ de $**MyCall**$ $**MyCall**$	Same again
CQ CQ CQ de $**MyCall**$ $**MyCall**$	And again
CONTROL_CE	Return to standby

An example contest macro:

CONTROL_CT

$**Call**$ UR 599 OC001 $**SerialNum**$ de $**MyCall**$ +

CONTROL_CE

Some TNCs require a short delay after issuing a CONTROL_X instruction or TNC command. For example, the KAM+ TNC does not like a CONTROL_CT instruction to put it in transmit mode immediately following a CW speed set command. The KAM+ instruction sequence:

CONTROL_CX

CW $speed+$

CONTROL_CT

Copyright © 2026 Robert C. Furzer K4CY

does not work correctly. The TNC takes several milliseconds to set the TNC speed before it is ready to receive input so the CONTROL_CT instruction needs to be delayed. The $**Delay**$ macro inserts a 250 millisecond delay before executing the next macro. The following macro correctly sets the CW speed for the KAM+ to, say, 20 WPM:

CONTROL_CX
CW 20
$**Delay**$
CONTROL_CT

22.6.4 KAM+ G-TOR mode

The G-TOR mode selection macros are:

CONTROL_CX	Return to command prompt
echo on	See your transmitted text
gtor	Sets G-TOR in standby mode

or

CONTROL_CX	Return to command prompt
echo on	See your transmitted text
gtor $**Call**$	Sets G-TOR mode calling the specified station

The following KAM+ G-TOR instructions can be used in Logger32:

CONTROL_CA	Abort link
CONTROL_CB	Enter transparent mode (to send binary code)
CONTROL_CD	Break link and stay in standby
CONTROL_CE	Returns to standby after transmit buffer is sent also puts +? If linked
CONTROL_CR	Immediately returns to receive mode regardless of TX buffer
CONTROL_CS	Selects the next mark/space shift (Hz)
CONTROL_CT	Enters transmit mode
CONTROL_CX	Returns to command mode
CONTROL_C0	Sets automatic baud rate
CONTROL_C1	Forces 100baud (irs)
CONTROL_C2	Forces 200baud (irs)
CONTROL_C3	Forces 300baud (irs)

An example G-TOR CQ macro is:

CONTROL_CT	Enter transmit mode
CQ CQ CQ de $**MyCall**$ $**MyCall**$ $**MyCall**$	Send the CQ message
PSE ARQ IN GTOR MODE ONLY	Additional text transmitted
K K K	Additional text transmitted

Copyright © 2026 Robert C. Furzer K4CY

CONTROL_CE	Return to standby

22.6.5 KAM+ Packet mode (AX.25)

Packet mode selection macros:

CONTROL_CX	Return to command prompt
echo off	Do not view transmitted text

Packet is pretty straightforward to most people (said with tongue in cheek). It is the default mode of the KAM+. The KAM+ is dual port in this mode {default switch STREAMSW 7E/$7C (~/|)} which are the tilde and pipe keys.

Some commands will affect both ports and some will affect the VHF port only, so we specify port paths for connect requests or beacon traffic. HF is port 1, VHF is port 2.

As the default mode is packet, the CONTROL_C instruction is not required in the macros. However, there does need to be a carriage return at the end of the last command to instruct the TNC to execute the macro statements.

Macros to switch between VHF and HF with the TNC using the default settings are:

~a	HF streamswitch character $7e or what you have set for your switch
|a	VHF streamswitch character $7c or what you have set for your switch

Note: The ‘a’ can be any letter from a to j, providing 10 user ports.

Example macro to connect to a VHF DX cluster:

|e

[callsign of the cluster node goes here]

or if it is on HF

~c

[callsign of the cluster node goes here]

The following may be looked at if you are in the situation where your network is shared by DX clusters, BBSs and APRS traffic.

The commands below are examples only. Enter your own callsign and the calls of the high site digipeaters with appropriate paths.

Remember, the KAM+ is dual port in packet mode and if your HF is purely used for voice I strongly suggest that when entering commands into the TNC that you use the following format:

command set /command example beacon /e 20

The following commands will program your beacon to transmit on VHF only every 20 minutes

Command HF port/VHF port

BEACON E 10/E 35

BTEXT :blna :HF gateway is active 10.149lsb ui with gate hf->vhf

blt 1 E 00:05:00/E 00:20:00

blt 2 E 00:12:00/E 00:30:00

Copyright © 2026 Robert C. Furzer K4CY

blt 3 E 00:22:00/E 00:40:00

blt 4 E 00:10:00/E 00:50:00

lt 1 =3748.33ST14520.11E- Station description

lt 2 =3748.33ST14520.11E- Station description

lt 3 =3748.33ST14520.11E- Station description

lt 4 :blna :HF-gateway on 10.149lsb ui with gate hf->vhf only

ltp 1 aprs v MMM/APRS

ltp 2 aprs v echo,MMM/aprs v LLL

ltp 3 aprs v MMM,OOO/aprs V WIDE2-2,LLL

ltp 4 aprs v MMM/aprs v trace2-2,LLL

The above commands all correspond with each other:

blt2: lt2: ltp2 where:

blt is beacon times per port

lt2 is information that is to be beaconed

ltp2 is the port and path with digipeater routes

22.6.6 KAM+ PACTOR mode

The PACTOR mode selection macros are:

CONTROL_CX	Return to command prompt
echo on	View transmitted text
pactor	PACTOR standby mode
CONTROL_CX	Return to command prompt
echo on	View transmitted text
pactor $**Call**$	PACTOR call the specified callsign
CONTROL_CX	Return to command prompt
echo on	View transmitted text
ptlisten	PACTOR listening mode for FEC and ARQ

The following KAM+ PACTOR instructions can be used in Logger32:

CONTROL_CA	Abort link
CONTROL_CD	Break link and stay in standby
CONTROL_CE	Returns to standby after transmit buffer is sent. Also puts +? if linked
CONTROL_CR	Immediately returns to receive mode regardless of TX buffer
CONTROL_CS	Selects the next mark/space shift (Hz)
CONTROL_CT	Enters transmit mode
CONTROL_CX	Returns to command mode

Copyright © 2026 Robert C. Furzer K4CY

CONTROL_C0	Sets automatic baud rate
CONTROL_C1	Forces 100baud (irs)
CONTROL_C2	Forces 200baud (irs)

An example CQ macro is:

CONTROL_CT	Enter transmit mode
CQ CQ CQ de $**MyCall**$ $**MyCall**$ $**MyCall**$ -- PTOR	Send CQ message
PSE ARQ	…
K K K	…
CONTROL_CE	Return to standby

22.6.7 KAM+ RTTY mode

The RTTY mode-selection macro is:

CONTROL_CX	Return to command prompt
echo on	Display transmitted data
rtty 45	Enter RTTY mode at 45 baud

The following KAM+ RTTY instructions can be used in Logger32:

CONTROL_CE	Returns to standby after transmit buffer is sent. Also puts +? if linked
CONTROL_CI	Inverts mark and space
CONTROL_CL	Sends letters-shift character
CONTROL_CN	Sends numbers-shift character
CONTROL_CR	Immediately returns to receive mode regardless of TX buffer
CONTROL_CS	Selects the next mark/space-shift (Hz)
CONTROL_CT	Enters transmit mode
CONTROL_CX	Returns to command mode
CONTROL_Cn	Where n = baud speed from the following list:

1 = 45	5 = 100	9 = 300
2 = 50	6 = 110	0 = ascbaud
3 = 57	7 = 150	
4 = 75	8 = 200	

An example CQ macro:

CONTROL_CT	Enter transmit mode
CQ CQ TEST de $**MyCall**$ $**MyCall**$ $**MyCall**$	Call CQ
CONTROL_CE	Return to standby

Copyright © 2026 Robert C. Furzer K4CY

23 Macros

> "A script is a story waiting to be told"
>
> *Steven Zaillian*

Many of Logger32's interactions with external devices, such as a CAT-connected radio, can be automated or scripted using macro commands and executed by clicking buttons. Up to 48 programmable buttons are available in each of the Sound card data window, Data Terminal, CW Machine window and **R**adio **C**ontrol **P**anels.

Likewise, using macros, strings can be generated and inserted into CW or digimode messages at run time. For example, the macro $**Call**$ included in a pre-formatted exchange or greeting message is interpreted when the message is played by, say, MMVARI or the CW machine, and is replaced with the callsign of the station we are logging at that moment.

23.1 Macro strings and commands

Logger32's macro strings and commands are described below[368].

After each macro is named, its applicability is shown [in *italics*] on the right:

CW applies to the CW Machine.

DATA applies to the Data Terminal (TNC).

RCP applies to the **R**adio **C**ontrol **P**anels.

PSK applies to the Sound card data window.

MMTTY applies to the Sound card data window and all profiles in MMTTY.

MMVARI applies to the Sound card data window and all modes in MMVARI.

DX applies to the DX cluster window.

Hinson tip: test/experiment with these macros by opening the **R**adio **C**ontrol **P**anel, CW Machine or Sound card data window running, say, MMVARI, then right-click and program a test message including the macro into one of the buttons. Save it, then click the button to run the macro to see what happens.

Hinson tip: *before* testing/experimenting with macros, disable your transmitter or QSY to some quiet frequency and minimize the RF power to avoid puzzling and annoying other hams.

[368] Macro commands are case-insensitive. Personally, I find mixed 'camel' case a little easier to interpret by eye. The PC doesn't care.

Copyright © 2026 Robert C. Furzer K4CY

$**AFC**$ *[CW, MMTTY, MMVARI]*

Toggles the **A**utomatic **F**requency **C**ontrol function **AFC on ⇄ AFC off**.

$**Align**$ *[MMTTY, MMVARI]*

Adjusts the radio frequency so the captured signal appears at the default audio frequency. Find out how to use align with mouse clicks in the Sound card data window chapter.

$**ASCIInnn**$ *[DATA, MMTTY, MMVARI]*

Inserts the numbered ASCII character into a message string. Look up the decimal code numbers for ASCII characters here. The code number from the ISO Latin-1 character set is normally 3 decimal digits but leading zeroes can be omitted.

Use this if you want to incorporate and send characters that are not readily available on your keyboard, such as the Euro symbol € using $**ASCII128**$.

Hinson tip: the sending software *may* or may *not* transmit the chosen character, depending on its programming and the character set available in the current radio mode. Likewise, the recipient *may* or may *not* see the character you *intended* to send. Use with caution.

$**Band**$ *[CW, DATA, MMTTY, MMVARI, DX]*

Inserts the band currently displayed in the log entry pane.

$**Bookmark**$ *[CW, DATA, MMTTY, MMVARI]*

Inserts a Pseudo DX spot into the DX Spots pane and BandMap.

$**Call**$ *[CW, DATA, MMTTY, MMVARI, DX]*

Inserts the callsign currently displayed in the log entry pane Call field (if any), in other words the station you are presently working. If you haven't copied and typed in their callsign yet, it does nothing: it doesn't mysteriously figure it out for you. That would be spooky.

$**CallChanged**$ *[CW]*

Almost the same as $**Call**$, this macro only inserts the callsign into a message *if* the call has been changed since it was last sent. This can be useful when correcting an erroneous callsign.

The macro will not work standalone: it has to be embedded within other data. If the macro generates an empty string, the following space is skipped, avoiding a d o u b l e – s p a c e.

For example, create the macro message "$**CallChanged**$ 5NN Gary". In the CW Machine, enter a callsign of AA1A and click to trigger the macro. It should send "AA1A 5NN Gary". Click the button to trigger the same macro again: *this* time, it just sends "5NN Gary" since the callsign is unchanged. Now change the callsign to "AA1AB": the same macro now sends "AA1AB 5NN Gary" when triggered for the first time since correcting the callsign.

Copyright © 2026 Robert C. Furzer K4CY

$**CallsignBeforeClearLog**$ *[CW, MMTTY, MMVARI]*

This can be used in conjunction with $**ClearLog**$ to respond to duplicate callers during contests or DXpeditions. It gives the callsign currently in the Call field of the log entry pane, then clears the log entry pane.

Example: $**Transmit**$ $**ClearLog**$ $**CallsignBeforeClearLog**$ QSO B4 TU QRZ $**Receive**$

Hinson tip: it is generally easier and safer to work and log duplicates, since it could be that the previous QSO was incomplete, uncertain or wrongly logged by the other person. They may just be keen to pass on an updated report, a cheery greeting or check propagation, again. Or something. Don't stress. Don't mess. Don't delay. Simply work 'em again, log 'em again and move on … as if nothing unusual just happened. Rise above it.

$**CallsignGetsFocus**$ *[CW]*

At the end of a transmission, sets the focus on the Callsign field in the CW Machine.

$**Clear**$ *[CW, DATA, MMTTY, MMVARI]*

Immediately clears (wipes) the TX, RX and Call fields.

This command can be used *only* as a standalone macro. No other text can be included in the command. Program a macro button with $**Clear**$ to wipe the lot with one quick press.

$**ClearBuffers**$ *[MMTTY]*

Clears (wipes) the TX window and buffer, plus the RX window. Stops sending the present text but the transmitter *remains* in transmit, ready for you to continue typing or sending a message macro.

This command can be used *only* as a standalone macro. No other text can be included in the command.

$**ClearLog**$ *[CW, DATA, RCP, MMTTY, MMVARI]*

Clears (wipes) the log entry pane.

$**ClearTXBuffer**$ *[MMTTY]*

Clears (wipes) the TX buffer, wiping any text you have typed-ahead.

This command can be used *only* as a standalone macro. No other text can be included in the command. Program a button with this to avoid having to select the text and hit the delete button if you change your mind about sending something.

$**ClearCallsignOnQSYOn**$ *[RCP]*

Ticks the <**Clear Callsign On QSY**> option.

$**ClearCallsignOnQSYOff**$ *[RCP]*

Unticks <**Clear Callsign On QSY**>.

Copyright © 2026 Robert C. Furzer K4CY

$**ClearQSYMarker**$ *[RCP]*

Resets the current <**Clear Callsign On QSY**> base frequency in memory, as if you have not yet QSYd.

$**Command**$ *[CW, DATA, MMTTY, MMVARI]*

Sends the ASCII text string following the command to a CAT-connected radio. See also $**HexCommand**$.

If used from the CW Machine, one control command should be given at a time. To allow multiple commands from a single macro, the CW Machine accepts "/" as a command separator.

If used with SO2R, the macro can accept the format $**Command[radio1text]|[radio2text]**$ where [**radio**N**text**] is an optional parameter and the "[" and "]" are *not* part of the parameter.

As the TEN-TEC OMNI VII uses *both* ASCII characters *and* hex numbers, a further modifier is available: hex numbers can be passed in this macro within "<" and ">". Thus a compound macro such as $**Command *A<B2>**$ is acceptable (although I have no idea what it does!). For more information see the TEN-TEC chapter.

Control_<X> *[DATA]*

Inserts <**Ctrl**+**X**> where <**X**> is a control character, usually a capital letter. Notice there are no "$" symbols enclosing this particular macro.

$**CW**$ *[MMVARI]*

Sends the text string following the command (up to 40 characters) on CW, then switches to RX. Give anyone monitoring your strange transmissions a clue about which mode you are using – particularly if you are CQing or developing some obscure, rare, novel or experimental mode.

$**CWid**$ *[MMVARI]*

Transmits your callsign in Morse code at the end of the current transmission, then switches to RX. Use this if you wish or are formally required to identify your transmissions: check your license small-print for details.

$**CWSpeedDn**$ *[CW]*

Decreases the CW keying speed by one WPM.

This command can be used *only* as a standalone macro. No other text can be included in the command. Define a 'Slower' macro button to trigger $**CWSpeedDn**$.

$**CWSpeedUp**$ *[CW]*

Increases the keying speed by one WPM.

This command can be used *only* as a standalone macro. No other text can be included in the command. Define a 'Faster' macro button to trigger $**CWSpeedUp**$.

Copyright © 2026 Robert C. Furzer K4CY

$**CWTextGetsFocus**$ *[CW]*

At the end of a transmission, focuses on the text field in the CW Machine.

$**Delay**$ *[DATA]*

Pauses 250 ms before sending the next character to the TNC *e.g.* giving the TNC a chance to settle down and get comfortable after sending it the instruction to change modes.

$**Down**$ *[MMVARI]*

Moves the main audio passband down 1 Hz *unless* AFC is locked on, in which case this command is callously ignored, leaving MMVARI in charge of the passband.

$**EndTime**$ *[DATA, MMTTY, MMVARI]*

The current UTC time will be logged as the QSO end time. See also $**StartTime**$.

$**Escape**$ *[DATA]*

Sends the ESC character (chr$(27)) to the TNC.

$**File**$ *[DATA, MMTTY, MMVARI]*

When followed on the same line with a disk, folder and text file name (*e.g.* $**File**$*C:\LOGGER32\BRAG.TXT*), the contents of that file are loaded into the TX buffer. If the filename is given alone with no folder or disk, the command defaults to the Logger32 folder.

The $**File**$ macro in the Data Terminal delays showing the text being loaded until the load is complete. Apparently, this is necessary for the software to ensure that any <**Esc**> characters embedded in the text are stripped out, with an appropriate delay inserted into the string. The same applies to any Control_<x> strings in the text.

> **Hinson tip**: you may feel the desperate need to brag about every last bit of equipment in your station but *please* consider the poor person on the other end or monitoring on the side, waiting patiently for their chance to go. Do you think anyone really cares that your ancient Amstrad laptop has a 286 processor with 16 megabytes of RAM and 5¼-inch floppy disk drive, or that your radio has a built-in automatic antenna tuning unit and digital signal processing? *Maybe* if your shack PC is a quantum computer or a difference engine, that *might* be worth mentioning. Briefly. Otherwise, get a life.

$**FocusToCallsignField**$ *[RCP]*

When an RCP macro completes, the focus normally remains wherever it was when the macro was triggered. Use this macro to focus on the Call field of the log entry pane, specifically, on completion, ready to start entering the next callsign even if that wasn't where it was focused before.

Copyright © 2026 Robert C. Furzer K4CY

$**Greeting**$ *[CW, DATA, MMTTY, MMVARI]*

Send a greeting appropriate to the local time of the person with whom you are in contact. If Logger32 cannot determine their local time (*e.g.* for a /MM or /AM station), a default generic greeting will be sent. See here for more.

$**HexBytes**$ *[DATA]*

Sends the hexadecimal string following the command to the Data Terminal, antenna switch port.

$**HexCommand**$ *[CW, DATA, MMTTY, MMVARI]*

Sends a command to the radio as a hexadecimal string. See also $**Command**$.

When used in the CW Machine, only a single command should be issued: to send multiple commands, use the slash "/" as a command separator.

See the SO2R chapter for notes on using this macro with SO2R.

$**IcomVFOB**$ *[RCP]*

Reads and displays the VFO B frequency for most ICOM radios when operating split – a clever workaround since (most) ICOMs don't normally report the VFO B frequency via CAT.

$**LastHighlightedCall**$ *[MMTTY, MMVARI]*

Places the last highlighted callsign into the log entry pane.

$**LastHighlightedCallAndLookup**$ *[MMTTY, MMVARI]*

As $**LastHighlightedCall**$ plus it does looks up the callsign using HamQTH.com, QRZ, HamCall or whatever you have configured for callsign lookups.

$**LastLoggedCallsign**$ *[MMTTY, MMVARI]*

Used like $**Call**$, this macro inserts the callsign from your previous QSO. You must have logged a previous QSO since starting Logger32 for this macro to work since it doesn't actually look in the logbook for the callsign - it simply recalls the most recent callsign you have logged during the present session.

$**LastLoggedName**$ *[MMTTY, MMVARI]*

Inserts the person's name from the previous QSO. You must have logged a QSO and recorded the person's name since starting Logger32 for this to work since it doesn't actually look in the logbook for the name - it simply recalls the previous operator's name you've logged in the present session.

$**LastQSOBand**$ *[DATA, MMTTY, MMVARI]*

Inserts the band of your previous QSO. You must have logged a QSO since starting Logger32 for this to work since it doesn't actually look in the logbook for the band - it simply recalls the band of the previous QSO you logged during the present session.

Copyright © 2026 Robert C. Furzer K4CY

$**LastQSODate**$ *[DATA, MMTTY, MMVARI]*

Inserts the UTC date of your previous QSO, in the same format as your log. You must have logged a QSO since starting Logger32 for this to work since it doesn't actually look in the logbook for the information - it simply recalls the date of the previous QSO you logged during the present session.

$**LastQSOTime**$ *[DATA, MMTTY, MMVARI]*

Inserts the UTC date of your previous QSO, in the same format as your log. You must have logged a QSO since starting Logger32 for this to work since it doesn't actually look in the logbook for the information - it simply recalls the time of the previous QSO you logged during the present session.

$**LastQSOMode**$ *[DATA, MMTTY, MMVARI]*

Inserts the mode of your previous QSO. You must have logged a QSO since starting Logger32 for this to work since it doesn't actually look in the logbook for the information - it simply recalls the mode of the previous QSO you logged during the present session.

$**Log**$ *[CW, DATA, MMTTY, MMVARI]*

Logs the QSO info currently in the log entry pane then clears the pane, ready for you to log the next QSO *e.g.* here is a friendly closing message that ends by dropping back to receive and logging the QSO:

Thanks $**Name**$ for this $**Mode**$ QSO.

Please QSL via LoTW.

73 $**Call**$ de $**MyCall**$ SK $**Receive**$ $**Log**$

$**LogImmediate**$ *[CW, MMTTY, MMVARI]*

Logs the QSO and clears the TX buffer *immediately* without sending any remaining text still in the buffer. Use this in a 'done' or 'interrupt' macro.

$**LongDate**$ *[DATA, MMTTY, MMVARI]*

Inserts the current UTC date in the format dd mon yyyy *e.g.* 16 Mar 2Ø24

See also $**ShortDate**$.

$**LongDateAndTime**$ *[DATA, MMTTY, MMVARI]*

Inserts the current UTC date and time into the message in the format dd mon yyyy, hh:mm:ss *e.g.* 16 Mar 2Ø24, 23:57:28

See also $**ShortDateAndTime**$.

$**LongTime**$ *[DATA, MMTTY, MMVARI]*

Inserts the current UTC time as hours, minutes and seconds *e.g.* 23:58:18

See also $**ShortTime**$.

Copyright © 2026 Robert C. Furzer K4CY

$**Lookup**$ *[CW]*

Regardless of where in the macro is placed, it immediately triggers a callsign auto-lookup (if configured) for the callsign in the Call field of the log entry pane.

$**Loop**$ *[CW, MMTTY, MMVARI]*

Used at the end of a macro, this restarts the macro *after* a five second delay as if you had waited a bit then clicked the button. The macro therefore continues looping indefinitely until interrupted *e.g*. by hitting <ESC>. Add multiple $**Loop**$s to extend the delay by a further 5 seconds for each one, or better still read on …

$**Loop N**$ *[CW, MMTTY, MMVARI]*

Restarts the macro after an N-second delay *e.g*. $**Loop 6**$ waits 6 seconds before replaying.

$**MMVARIMode *N***$ *[MMVARI]*

Changes the Sound card data window to one of MMVARI's modes, where ***N*** is the mode's menu index number as shown here (the embedded space and any leading zeros are optional):

$**MMVARIMode 0**$	GMSK (MBCS)	$**MMVARIMode 17**$	MFSK-L 22
$**MMVARIMode 1**$	FSK (MBCS)	$**MMVARIMode 18**$	MFSK-L 32
$**MMVARIMode 2**$	FSK-W (MBCS)	$**MMVARIMode 19**$	MFSK-L 64
$**MMVARIMode 3**$	BPSK 31 (MBCS)	$**MMVARIMode 20**$	MFSK-U 4
$**MMVARIMode 4**$	BPSK 63 (MBCS)	$**MMVARIMode 21**$	MFSK-U 8
$**MMVARIMode 5**$	BPSK 125 (MBCS)	$**MMVARIMode 22**$	MFSK-U 11
$**MMVARIMode 6**$	BPSK 250 (MBCS)	$**MMVARIMode 23**$	MFSK-U 16
$**MMVARIMode 7**$	BPSK 31	$**MMVARIMode 24**$	MFSK-U 22
$**MMVARIMode 8**$	BPSK 63	$**MMVARIMode 25**$	MFSK-U 32
$**MMVARIMode 9**$	BPSK 125	$**MMVARIMode 26**$	MFSK-U 64
$**MMVARIMode 10**$	BPSK 250	$**MMVARIMode 27**$	QPSK-L 31
$**MMVARIMode 11**$	RTTY-L	$**MMVARIMode 28**$	QPSK-L 63
$**MMVARIMode 12**$	RTTY-U	$**MMVARIMode 29**$	QPSK-L 125
$**MMVARIMode 13**$	MFSK-L 4	$**MMVARIMode 30**$	QPSK-U 31
$**MMVARIMode 14**$	MFSK-L 8	$**MMVARIMode 31**$	QPSK-U 63
$**MMVARIMode 15**$	MFSK-L 11	$**MMVARIMode 32**$	QPSK-U 125
$**MMVARIMode 16**$	MFSK-L 16		

$**Mode**$ *[DATA, MMTTY, MMVARI]*

Inserts the digital mode currently shown in the log entry pane *e.g*. BPSK 31

Copyright © 2026 Robert C. Furzer K4CY

$**MouseTF-Set**$ *[CW, RCP, MMTTY, MMVARI]*

A macro function to replicate the TF-Set front panel button on some Kenwood radios, enabling you to listen briefly on your transmit frequency while operating split. When you press and hold down a button defined with this macro, Logger32 sends the TS1; CAT command, telling the radio to shift the VFO temporarily to your transmit frequency. After you release the mouse button, it sends TS0; to resume listening on the original receive frequency.

Note: this only works on Kenwood radios if you have ticked <**Check to enable Kenwood TF-Set state polling**> under **Setup ⇨ Radio ⇨ Radio 1|2 configuration**. More here.

$**Msg*N***$ *[CW]*

This immediate CW macro only works with WinKeyer2. Macros $**Msg1**$ to $**Msg6**$ simply trigger the replay of WinKeyer2's internal CW memories 1 to 6.

Hinson tip: these are handy macros because the 4 physical buttons on the WinKeyer2 are disabled while the PC has control, while messages 5 and 6 are not so easy to send anyway.

$**MultiRX**$ *[MMVARI]*

Activates MMVARI's MultiRX function.

$**MyCall**$ *[CW, DATA, MMTTY, MMVARI]*

Inserts the current "operator" value … which *should* be the callsign you are identifying with over the air – not your name or some other string.

$**Name**$ *[CW, DATA, MMTTY, MMVARI]*

Inserts the contents of the Name field currently shown in the log entry pane.

$**NameGetsFocus**$ *[CW]*

At the end of a scripted transmission, sets the focus on the Name field in the CW Machine, ready for you to type in the other person's name … or tab to a different field.

$**Net**$ *[MMTTY, MMVARI]*

Toggles **Net On** ⇄ **Net Off**.

$**NumQSOs**$ *[DATA, MMTTY, MMVARI]*

Inserts the total number of QSOs logged with the station currently being worked.

$**QSX+nnnn**$ or $**QSX-nnnn**$ *[MMTTY, MMVARI]*

For split operation, shifts the radio's *transmit* VFO up or down by nnnn Hz *e.g.* $**QSX-2000**$ shifts the transmit frequency 2 kHz LF, $**QSX+1500**$ shifts the TX 1.5 kHz HF.

Any split can be instantly cancelled (zeroed) by the $**Simplex**$ macro.

Copyright © 2026 Robert C. Furzer K4CY

$**QSY(nn)nnn.(nn)**$ *[DATA, MMTTY, MMVARI]*

Immediately tunes the radio VFO to the specified value in kiloHertz (nnnnn.nn kHz). For example: $**QSY7070**$ for 7,070 kHz or $**QSY14070.22**$ for 14,070.22 kHz. The megaHertz digits are optional: if omitted, a frequency *within* the current band is assumed. The decimal point and the final two digits are also optional: if omitted, .00 is assumed.

The macro can be invoked from any mode: the radio will change mode for the new frequency if a different mode is specified in your Bands & Modes table.

The macro is ignored if the radio is transmitting at the time *e.g.* "Fred - QSY to 14080",$**RTTY**$,$**QSY 14080.00**$ will send Fred the message but will *not* action the QSY macro while the radio remains in transmit, so insert a macro to drop back to receive first *e.g*. "Fred - QSY to 14080",$RTTY$,$**Receive**$,$QSY 14080.00$

$**QTH**$ *[MMTTY, MMVARI]*

Inserts the contents of the QTH field from the log entry pane.

$**Radio1**$ and $**Radio#2**$ *[RCP]*

Selects radio #1 or #2.

$**Radio1->Radio2**$ *[RCP]*

Copies the frequency and mode *from* radio 1 *to* radio 2, so both radios end up on the same frequency and mode as radio 1 was on when the macro was executed.

$**Radio2->Radio1**$ *[RCP]*

Copies the frequency and mode from (you guessed it) radio 2 to radio 1.

$**Radio1Offset *N***$ *[RCP]*

Where ***N*** is the transverter offset in kHz. For example, to transvert Radio 1 from 28 MHz to 144 MHz, the macro would be $**Radio1Offset 116000**$.

To turn off or cancel the transverter offset, use $**Radio1Offset 0**$.

$**Radio2Offset *N***$ *[RCP]*

Use this to configure Logger32 for the transverter connected to Radio 2.

$**RadioAndTone**$ *[MMTTY, MMVARI]*

Returns the RF center frequency (radio carrier frequency plus audio tone frequency) if using PSK. MMTTY reports the RF frequency of the RTTY mark tone (the radio frequency *plus* the mark tone audio frequency). MMVARI reports the median *i.e*. the RF frequency half way between the RTTY mark and space frequencies.

Copyright © 2026 Robert C. Furzer K4CY

$**RadioControlPanelMacro*NN***$ *[CW, MMTTY, MMVARI]*

Execute a defined **R**adio **C**ontrol **P**anel macro without the Sound card data window or CW Machine losing focus. ***NN*** is between 01 and 48, corresponding to the 48 RCP macro buttons. Use this, for instance, to call an RCP macro that automatically increments the VFO by 1 kHz after logging a QSO in an RTTY, PSK or CW Sprint.

This macro executes RCP macros assigned to the radio in focus. If you have 2 radios, be sure the desired function is defined in the same macro slot for both radios - otherwise you need separate macros in the CW Machine and Sound card data window for each radio.

This macro still works even if the RCP is not currently shown.

$**RadioFreq**$ *[DATA, MMTTY, MMVARI]*

Returns the VFO (nominal RF carrier) frequency of the radio *e.g*. "28Ø94.11Ø kHz"

$**RadioPortOpen**$ and $**RadioPortClose**$ *[RCP]*

Use these macros to open or close the CAT port – the one for Radio 1 if you have an SO2R setup (use $**SO2RPortOpen**$ and $**SO2RPortClose**$ for … the other radio).

$**Receive**$ *[CW, MMTTY, MMVARI]*

Switches the radio from transmit to receive when the transmit buffer is empty *i.e*. the message has been sent.

$**ReceivedRST**$ *[CW, DATA, MMTTY, MMVARI]*

Inserts the contents of the RST received field from the log entry pane. The default is 599 if nothing is entered.

$**ReceivedRSTn**$ *[CW]*

Inserts the contents of the RST received field from the log entry pane, replacing the digit 9 with an N. Defaults to 5NN if there is no RST in the field.

$**ReceivedGrid**$ *[DATA, MMTTY, MMVARI]*

Inserts the received grid square from the log entry pane.

$**Reset**$ *[CW]*

Immediately halts the parallel or serial port CW keying and/or releases the PTT control line, sending the radio to receive. This command can be used *only* as a standalone macro. No other text can be included in the command. Use it as an emergency abort button.

Copyright © 2026 Robert C. Furzer K4CY

$**ResetRadio1**$ and $**ResetRadio2**$ *[RCP]*

Returns radio 1|2 to its previous VFO frequency, wherever it was before the last CAT-commanded QSY.

For example, if have been CQing on radio 1 using CW on 28025 kHz, then click a DX spot for someone on SSB on 28455 kHz, radio 1 changes frequency and mode for me to listen for the DX. If the DX is busy or too weak to bother calling, I can click an RCP button containing the macro $**RestRadio1**$ to return to my CQ frequency and mode.

Each time I click the macro button, the radio swaps back to the previous frequency and mode, making it easy to continue checking briefly for the DX between my CQ calls in the hope that eventually we can connect.

$**Rotator**$ *[CW, DATA, MMTTY, MMVARI, RCP]*

Rotates the antenna to the computed short path azimuth for the callsign in the log entry pane, as if you pressed <**Ctrl**+**A**>.

$**RotatorLP**$ *[CW, DATA, MMTTY, MMVARI, RCP]*

Rotates the antenna to the computed long path azimuth for the callsign in the log entry pane, as if you pressed <**Alt**+**A**>.

$**RTTY**$ *[MMTTY, MMVARI]*

Changes the Sound card data window to RTTY mode with normal/uninverted tones.

$**RTTY-i**$ *[MMTTY, MMVARI]*

Changes the operational mode of the Sound card data window to MMTTY reverse (RTTY with mark and space **i**nverted). If a special MMTTY profile has been selected, this macro opens MMTTY with the profile last used, retaining the selected mode. If the profile last used was the one of the default profiles, it will open to that profile but with inverted tones.

$**RTTYBPFOff**$ *[MMTTY]*

Turns off the receive bandpass filter.

$**RTTYBPFOn**$ *[MMTTY]*

Turns on the receive bandpass filter.

$**RTTYFigures**$ *[MMTTY]*

Shifts the RTTY transmission to the figures (numeric) character set.

$**RTTYLetters**$ *[MMTTY]*

Shifts the RTTY transmission to the letters character set.

Copyright © 2026 Robert C. Furzer K4CY

$**RTTYMarkFrequency**$ *[MMTTY]*

Returns the RTTY mark frequency. This is normally the frequency of the upper tone but will be the frequency of the *lower* of the two RTTY tones if you are in reverse mode.

$**RTTYNormal**$ *[MMTTY]*

Operate RTTY using normal shift (LSB).

$**RTTYReverse**$ *[MMTTY]*

Operate RTTY using reverse or inverse shift (USB).

$**RTTYSetup**$ *[MMTTY]*

Displays the MMTTY setup form as if you pressed the "Setup" button on the RTTY toolbar.

$**RTTYShift**$ *[MMTTY]*

Returns the current setting of the RTTY shift *i.e*. normal or reverse.

$**RTTYSquelchOff**$ *[MMTTY]*

Disables RTTY squelch. The decoder attempts to decode even very weak signals.

$**RTTYSquelchOn**$ *[MMTTY]*

Enables RTTY squelch. The decoder ignores signals below the squelch threshold.

$**RXToneFreq**$ *[MMTTY, MMVARI]*

Returns the mark tone currently displayed in the second panel of the Sound card data window status bar (see the More complex macros section).

Unlike MMTTY, MMVARI reports the median frequency half way between Mark and Space.

$**SELCAL**$ *[DATA]*

Inserts the contents of the SELCAL field (top left corner of the Data Terminal).

$**SentRST**$ *[CW, DATA, MMTTY]*

Inserts the contents of the RST sent field of the log entry pane. The default is 599 if the field is empty.

$**SentRSTn**$ *[CW]*

Inserts the contents of the RST sent field from the log entry pane, replacing the digit 9 with an N. Defaults to 5NN if there is no RST in the field.

Copyright © 2026 Robert C. Furzer K4CY

$**SerialNum**$ *[CW, DATA, MMTTY, MMVARI]*

Inserts the serial number of this QSO, using the value shown in the 9th panel on Logger32's status bar.

$**SerialNum-1**$ *[CW, DATA, MMTTY, MMVARI]*

Inserts the serial number of the *previous* QSO. Use this macro if you need to re-send a contest report having already (prematurely!) logged the QSO.

$**SetIcomClock**$ *[RCP]*

Sets the radio clock to local time on the IC-7300 *(specifically*[369]*)*.

$**SetIcomClockUTC**$ *[RCP]*

Sets the radio clock to UTC on the IC-7300 *(specifically)*.

$**SetIcomDate**$ *[RCP]*

Sets the radio clock to today's local date on the IC-7300 *(specifically)*.

$**SetIcomDateUTC**$ *[RCP]*

Sets the radio built-in clock to today's UTC date on the IC-7300 *(specifically)*.

$**SetIcomUTCOffset**$ *[RCP]*

Sets the radio's built-in clock offset from UTC on the IC-7300 *(specifically)*.

$**SetKenwoodClock**$ *[RCP]*

Sets the radio built-in clock to local time on *some* Kenwoods – we're not sure which ones.

$**SetKenwoodClockUTC**$ *[RCP]*

Sets the radio built-in clock to UTC on *some* Kenwoods.

$**ShortDate**$ *[DATA, MMTTY, MMVARI]*

Inserts the current UTC date in the format dd/mm/yy *e.g.* 15/Ø1/24

See also $**LongTime**$.

[369] It *may* work on other ICOMs: if it works for yours, please let us know via the Logger32 reflector.

Copyright © 2026 Robert C. Furzer K4CY

$**ShortDateAndTime**$ *[DATA, MMTTY, MMVARI]*

Inserts the current UTC date and time in the format dd/mm/yy hh:mm *e.g.* 15/Ø1/24 Ø2:Ø5

For example, the macro "It is $**ShortDateAndTime**$ UTC right now" sends something like "It is 16/Ø3/24 22:16 UTC right now", sending the actual UTC date and time at the moment the macro is triggered and interpreted.

See also $**LongDateAndTime**$.

$**ShortTime**$ *[DATA, MMTTY, MMVARI]*

Inserts the current UTC time in the format hh:mm *e.g.* Ø2:Ø5

See also $**LongTime**$.

$**Simplex**$ *[MMTTY, MMVARI]*

Cancels split operation. It un-splits. Transmission and reception use the same frequency.

$**SlaveCommand xxxx**$ *[RCP]*

Sends a command string xxxx to the slave radio.

$**SlaveHexCommand xxxx**$ *[RCP]*

Sends a hexadecimal CAT command string xxxx to the slave radio.

$**SlavePortClose**$ *[RCP]*

Closes the slave port.

$**SlavePortOpen**$ *[RCP]*

Opens the slave port.

$**SlaveSync**$ *[RCP]*

Sends the main radio's frequency and mode across to the slave radio.

$**SO2R**$ *[RCP]*

Opens the SO2R **R**adio **C**ontrol **P**anel.

$**SO2RPortOpen**$ and $**SO2RPortClose**$ *[RCP]*

Use these macros to open and close the CAT port for radio 2 in an SO2R setup. See also $**RadioPortOpen**$ and $**RadioPortClose**$ for radio 1's CAT port.

$**SO2V**$ *[RCP]*

Opens the SO2V **R**adio **C**ontrol **P**anel.

Copyright © 2026 Robert C. Furzer K4CY

$**SO2Von**$ *[RCP]*

Opens the SO2V **R**adio **C**ontrol **P**anel.

$**SO2Voff**$ *[RCP]*

Closes the SO2V **R**adio **C**ontrol **P**anel.

SO2Von and **SO2Voff** can be used either standalone or as part of split/unsplit macros.

$**Speed-**$ *[CW]*

Temporarily slows down CW keying by two words per minute. This only takes effect within the current transmission: the speed resumes the original setting at the end of the current transmission. Use this to send or repeat critical information m o r e s l o w l y if necessary within a message (*e.g.* repeating your callsign, serial number or zone a little slower if someone hasn't copied them correctly the first time).

$**Speed+**$ *[CW]*

Temporarily speeds up CW keying by two words per minute. This only takes effect within the current transmission: the original speed resumes automatically at the end of the current transmission.

Hinson tip: use this to shave a few milliseconds off the over when sending a standard contest or pileup report maybe. Don't overdo it though: you are expected to be able to copy CW reliably at the higher speed since it is considered rude to transmit CW faster than you can receive. If you normally dawdle along at 25 WPM but send 5NN at 40+ WPM, some wags will blast their entire report to you at 40+ WPM, just once, and move swiftly along. Missed it? Too bad: you asked for it!

$**SplitAudioAlert**$ *[RCP]*

Sounds the 'spleeeet' audio alert whenever a macro puts the radio into split mode.

$**SplitVisualAlertOn**$ *[RCP]*

Shows the visual split alert whenever a macro puts the radio into split mode.

$**SplitVisualAlertOff**$ *[RCP]*

Clears the visual split alert message from the screen.

$**SRXGetsFocus**$ *[CW]*

At the end of a transmission, sets the focus on the received serial number field in the CW Machine, ready for you to type in the serial number sent by the person you are working during a contest.

Copyright © 2026 Robert C. Furzer K4CY

$**StartTime**$ *[DATA, MMTTY, MMVARI]*

Uses the current UTC time as the QSO start time when the QSO is logged. See also $**EndTime**$.

Hinson tip: if you are trying to *transmit* the UTC date and/or time that the QSO started within a digimode message, try using $**LongDateAndTime**$, $**ShortDateAndTime**$, $**LongDate**$, $**ShortDate**$, $**LongTime**$ or $**ShortTime**$.

$**sRX**$ *[CW, DATA, MMTTY, MMVARI]*

Sends the *received* serial number from the log entry pane (if any) *e.g.* to confirm to the person we are working that we received and typed in their number correctly.

$**sTX**$ *[DATA, MMTTY, MMVARI]*

The content of the STX field (sent serial number) from the log entry pane is transmitted.

Note: on CW, use $**SerialNum**$ instead.

$**TF-Set**$ *[CW, RCP, MMTTY, MMVARI]*

For Kenwood radios, toggles the TF status of the radio by sending the TS1; or TS0; CAT commands. It only works if you have ticked <**Check to enable Kenwood TF-Set state polling**> under **Setup ➪ Radio ➪ Radio 1|2 configuration**. It only works on Kenwood radios. See also $**MouseTF-Set**$. More here.

$**TNCdate**$ *[DATA]*

This macro string is substituted with the date and time as yymmddhhmmss – specifically the *local* (PC) date and time intended for setting the TNC's internal real-time clock. Some TNCs require a word to prefix this macro's output *e.g.* "DATE $**TNCdate**$".

$**ToggleRadios**$ *[CW, DATA, MMTTY, MMVARI]*

This immediate macro swaps to the *other* SO2R radio, as if you had pressed <**Ctrl**+**T**>.

This command can be used *only* as a standalone immediate macro. No other text can be included in the command.

$**Transmit**$ *[CW, MMTTY, MMVARI]*

Grounds the PTT to turn the transmitter on and transmit the text in the TX window. Can be used with MOX and/or slow typing.

$**Tune**$ *[CW]*

In the CW Machine, keys down to tune the transceiver/ATU. This is a toggle.

$**TwoHexBytes xx xx xx**$ *[DATA]*

Send hexadecimal characters to the TNC.

Copyright © 2026 Robert C. Furzer K4CY

$**uHam XX**$ *[RCP]*

Send op command without the need for the <Cr> terminator, where **XX** is the appropriate *micro*HAM control op command.

$**Up**$ *[MMTTY, MMVARI]*

Moves the main passband up 1 Hz. If AFC is locked on, this command is ignored.

$**UpperOrLower**$ *[MMTTY, MMVARI]*

Returns a plus sign (+) if the radio is in USB or a minus sign (-) if the radio is in LSB (see More complex macros).

$**Version**$ *[DATA, MMTTY, MMVARI]*

Sends your logging software name and version *e.g.* Logger32 Version 4.Ø.318

$**Wait *N***$ *[RCP]*

Inserts a delay of ***N*** seconds between radio commands.

$**WinkeyMergedLetters**$xy *[CW]*

Send xy, the two following characters, as a prosign without an inter-character space using a WinKeyer *e.g.* $**Winkeymergedletters**$BT sends <**Esc**>BT to the WinKeyer which generates the **– ●●● –** hyphen/break character in Morse code. See also ^, the final entry in this list of macros.

[*[MMTTY]*

Turn RTTY diddle off. This command also works if typed into the TX window.

] *[MMTTY]*

Turn RTTY diddle on. This command also works if typed into the TX window.

| *[CW]*

Send an extended inter-character space, 50% longer than normal. Use this to avoid similar/short dotty characters being confused or lost *e.g.* SP2|EWQ

---

^ *[CW]*

Concatenate letters without the letter space. CW prosign sequences such as AR, SK and BK can be generated in macros using a caret between the letters to be joined *i.e.* A^R, S^K and B^K respectively.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: macros are *only* triggered by clicking macro buttons or pressing function keys. If you type them directly into the TX window expecting them to be interpreted, you will be sadly disappointed. If you don't believe me, feel free to check this out for yourself … and consider reserving one of your macro buttons for experimenting with and debugging macros in future.

Hinson tip: most of Logger32's macros are enclosed by a pair of dollar symbols, with two important consequences. Firstly, you *must* use a matching pair: accidentally neglecting to include either dollar can cause unexpected strangeness. Secondly, anything outside the dollar pair is not treated as a macro/command by Logger32 … so that's your cue to write brief comments explaining your macros. Trust me, even the 'obvious' ones can prove confusing when you are hunting down possible bugs in your macros or trying to explain their operation to friends, perhaps months or years later, maybe in the early hours when P5DX pops up unexpectedly on 160m but the rig stubbornly *refuses* to leave 40m …

23.2 Advanced macros

The following examples illustrate how to use Logger32's more complex, tricky, unusual, advanced, peculiar, obscure or you might say weird macros.

23.2.1 $ASCIInnn$

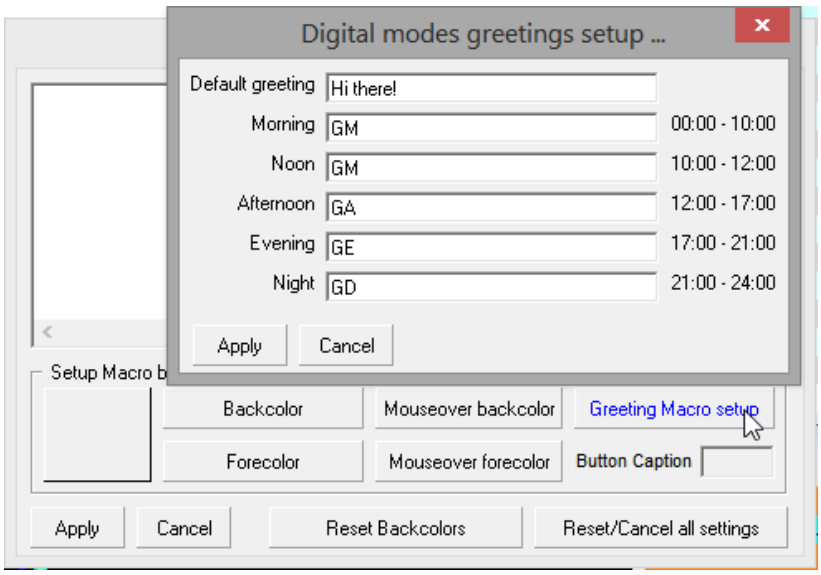
This macro inserts the ASCII character corresponding to code (n)nnn from the ASCII character set, even if it cannot be entered from a keyboard. For example, the ASCII code 191 equates to the inverted question mark "¿". So the string "$**ASCII191**$Que pasa?" would print as "¿Que pasa?".

Leading zeroes in the ASCII code are optional, so both $**ASCII0191**$ and $**ASCII191**$ generate the "¿" character.

The tilde (~) character, meaning "about" or "roughly", cannot be typed directly into a macro but you can use $**ASCII176**$ in its place *e.g.* set up a macro button (say F12) with macro $**ASCII176**$ and a button caption of **chr**$**(0176)**. Now, with the cursor in the TX window, type "The temp here today is <**F12**>90F" to send "The temp here today is ~90F" or "My amp runs <**F12**>1 kW" to send "My amp runs ~1 kw".

23.2.2 $Greeting$

You can define up to five cheerful salutations appropriate to the local times of the DX stations you are working (*e.g.* good morning, afternoon, evening or day) from the <**Greeting Macro setup**> button in any macro setup ▶



The **default greeting** (*e.g.* Hi there!) is used if Logger32 cannot determine the DX time zone (*e.g.* for a /Maritime Mobile station in an unknown grid square).

Copyright © 2026 Robert C. Furzer K4CY

There are separate setup panels for the CW Machine and the digital modes (*e.g.* the exclamation character is rarely used in Morse, so "Hi there!" on data modes might be shortened to plain "HI" for CW).

23.2.3 $UpperOrLower$, $RadioFreq$, $RXToneFreq$ and $RadioAndTone$

These macros are applicable to both PSK and RTTY. Here is an example of how to use these them in PSK. The following macro is defined for the hotkey <**F5**>:

> *Your exact frequency is* $**RadioAndTone**$ kHz*. Here's how I figure that out.*
>
> *My receiver is on $**RadioFreq**$ kHz. I am receiving your signal at $**RXToneFreq**$ kHz.*
>
> *Therefore your transmitted signal is on $**RadioFreq**$ $**UpperOrLower**$ $**RXToneFreq**$ i.e. $**RadioAndTone**$ kHz. QED*

Pressing <**F5**> interprets the macros at run time, generating a message of the form:

> *Your exact frequency is 14084.3 kHz. Here's how I figure that out.*
>
> *My receiver is on 14083.0 kHz. I am receiving your signal at 1.3 kHz.*
>
> *Therefore your transmitted signal is on 14083 + 1.3 i.e. 14084.3 kHz. QED*

To send the equivalent message using RTTY, swap $**RXToneFreq**$ with $**RTTYMarkFrequency**$.

23.2.4 $Command$ and $HexCommand$ macros

These macros are used to send CAT commands to the radio serial port in either ASCII ($**Command**$) or hexadecimal ($**HexCommand**$) formats.

For example, $**Command**$ can change the radio frequency and/or mode using, say, $**Command 14000.123 CW**$ Spaces at the beginning and end of the string are optional but the frequency parameter *must* be spaced from the mode in this example. The frequency is specified in kHz with either a "," (comma) or a "." (period, full-stop) as the decimal separator. The mode must be a valid mode for the radio concerned.

Command macros can be used to implement radio functions not directly supported by Logger32. Program them into macro buttons to invoke the appropriate command sequences by clicking the buttons. For example, you could program a macro button to narrow the filters in your radio and enable **D**igital **N**oise **R**eduction (to focus-in on a weak CW signal, perhaps), and another button to revert to normal bandwidth without DNR.

23.2.5 ConCATenating

If multiple CAT commands are chained together, Logger32 sends them to the radio in the same polling interval, whereas if they are separate, the commands are sent in successive polling intervals. If you are keen to shave some fractions of a second off the time taken to execute a macro, try conCATenating *e.g*. instead of the following two commands taking 2 polling intervals:

- $**Command FT1;**$ Set VFO B as the transmit VFO
- $**Command FR0;**$ Set VFO A as the receive VFO

… the conCATenated $**Command FT1;FR0;**$ sends both commands in the same interval … but notice the lack of comments explaining each individual command.

Hinson tip: keep the commands separate and commented while you are developing and testing your macros. Only conCATenate them when you are happy with them and don't intend to change.

Copyright © 2026 Robert C. Furzer K4CY

23.3 Macros FAQ

Q. When I click a macro button, nothing happens. What's wrong?

A. No idea, sorry. Try clicking harder. Have you fed your mouse lately?

Well, OK then, depending on whether you are having problems with sound card data mode macros, CW Machine macros, CAT command macros or $omethingElseEntirely$, look carefully for little issues such as:

- Clicking the wrong macro button or pressing the wrong function key(!)
- Simple typos in the macros *e.g.* $**Greting**$
- Not-quite-so-simple typos in the macros *e.g.* $**Greet1ng**$
- Missing dollars *e.g.* $**Greeting**… or #**Greeting**#
- Extra dollars *e.g.* $$**Greeting**$$
- Spurious dollars *e.g.* $**Greet$ng**$
- Missing CAT command terminators *e.g.* $**Command FT0**$ not $**Command FT0;**$
- Spur io us spac es *e.g.* $ **Greeting** $
- Macros that are not *supposed* to work with the CW Machine, sound card data modes or **R**adio **C**ontrol **P**anel *i.e.* they have not been coded, so the system doesn't know how to interpret them – hint: in the list above, check the

  [text to the right of the macro name]

- Macros sending commands to the *other* radio or VFO
- Macros sending commands that are ignored or not correctly interpreted by the radio *e.g.* trying to use a function such as turn on a sub-receiver or filter that isn't actually installed in your radio (!), or trying to command a radio that is in some sort of programming or sleep mode, or that has locked-up and is now sulking

> **Hinson tip**: try composing and triggering a basic macro just to check the connections and configs.

- Macros sending CAT commands to radios that are not actually physically connected to the PC right at this moment, or are not turned on, or are using a closed port, or are using the wrong port speed, or are using the wrong syntax (*e.g.* sending Icom commands to a Yaesu) … or something equally daft (been there, done that)
- Macro commands that do exactly what they are supposed to do … but are then followed by further commands that *instantly* reverse the effects (keep one eye on the screen *e.g.* the TX window, and another on the radio looking for clues as to what's going on, or use the radio debug window to check the sequence of CAT commands and responses)

Good luck.

Copyright © 2026 Robert C. Furzer K4CY

24 Programmable macro buttons and hotkeys

> "A button is
> a way to transform
> an idea into action"
>
> *Chris Anderson*

You can program up to 48 macro buttons in each of the Sound card data window, the Data Terminal, the CW Machine window and the **R**adio **C**ontrol **P**anels, plus up to 12 message play buttons in the **D**igital **V**oice **K**eyer. Each button can be programmed to execute scripts containing macro functions/commands (*e.g.* to set the CW keying speed) and/or text (*e.g.* CW message content), and can be given a label with optional colors. Using your imagination, working with the available list of macros, you can make operating on various modes and controlling/using your radios easier and more enjoyable.

This chapter explains in general terms how to program the buttons using Logger32's macro/scripting language. The Sound card data window, Data Terminal, CW Machine, **R**adio **C**ontrol **P**anel and **D**igital **V**oice **K**eyer chapters have more specific details, while the macros chapter elaborates on the available macro functions/commands.

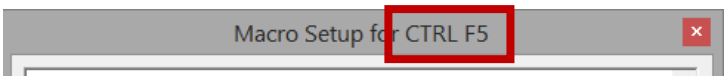
24.1 Programmable buttons and hotkeys

Programmable buttons can be displayed in up to four rows of 12 buttons on each of the Sound card data window, Data Terminal, CW Machine and the **R**adio **C**ontrol **P**anels. That's up to 192 (4x48) buttons per configuration – more than enough for even the most ardent button-pusher.

Each button initially has a default hotkey assigned to it as follows:

- The top (1st) row of buttons defaults to the <**Fn**> keys, (function keys F1 through F12).
- The upper middle (2nd) row of buttons defaults to <**Alt**+**Fn**> keys (hold down Alt and tap the function key).
- The lower middle (3rd) row of buttons corresponds to <**Ctrl**+**Fn**> keys (hold down Ctrl and tap the function key).
- Access the bottom (4th) row of buttons using <**Ctrl**+**Alt**+**Fn**> keys (hold down Ctrl and Alt, then tap the function key with your nose).

If, like me, you forget, the captions of the button right-click setup panes show the corresponding hotkeys ▶



Pressing a hotkey runs the macro script programmed into the corresponding button, just as if you had pointed the mouse cursor at the button and clicked the left mouse button.

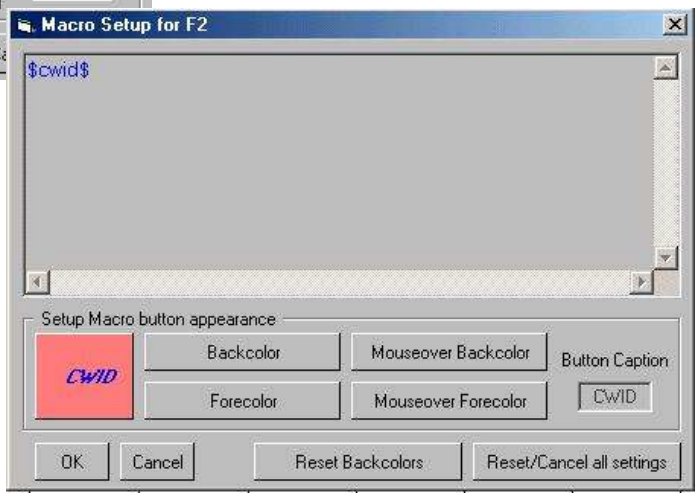
Copyright © 2026 Robert C. Furzer K4CY

24.1.1 Programming the buttons



◀ Right-click the button you wish to program to open a macro setup form.

To program the button, click in the large gray area and type your script consisting of macro commands and/or text strings[370]. Then type a name into the <**Button Caption**> field to use as a macro button label[371].



The macro button's appearance can be customized using the rectangular buttons and previewed in the square button ▶

- **Backcolor**: opens a standard Windows color palette from which to select the color of the button's background[372].
- **Forecolor**: select the type, style and color of the text on the button.
- **Mouseover Backcolor**: select the color to which the button background changes when you mouseover it.
- **Mouseover Forecolor**: select the text/font color for the button when you point the mouse cursor at it, or pass over it on your way to click another button.

---

[370] To enter a special character that is not available on your keyboard, you can enter its three-digit extended ASCII code on the numeric keypad (with Num Lock on) whilst holding down the <**Alt**> key. For example, to enter the ¿ upside-down question mark character used in Spanish, hold down <**Alt**> while you type 168 on the numeric keypad. When you release the <**Alt**> key, the ¿ is displayed. Wake someone up with a ding by including the "bell" character in a message using <**Alt+007**>. When you type a special character in this way, you *must* enter a three-digit ASCII code on the numeric keypad including the leading zero/s for codes below 100. However, digimodes typically support limited character sets that may not be able to communicate them all over the air. It's fun to experiment though.

[371] Special characters can be used in button captions: entering the ASCII code of any character in the **chr$(nnn)** function will display that character as part of the caption text. For example, defining the caption of a button as **chr$(191)** would display "¿" on the button.

[372] Choose your colors and button captions wisely. Color-coding the buttons for standard *vs*. contest QSOs, radio commands *etc*. can help – or hinder – your clicking the correct one. Bright backgrounds stand out nicely, but choose contrasting colors if you expect to read the button text easily: yellow text is almost invisible on a yellow-ish background while white text is hard to make out on any light background, especially given the rush of a big pileup, or the exhaustion 47 hours in to a 48-hour contest …

Copyright © 2026 Robert C. Furzer K4CY

- **OK** saves and enables the script and applies settings when you are ready to give it a go.
- **Cancel** quits, closing the form without saving or enabling any changes you may have made.
- **Reset Backcolors** restores the default background colors (a delightfully drab shade of gray, one of 50).
- **Reset/Cancel all settings** restores the button to the way it was before you began editing it, if you decide it's all too hard. If it had a script, that script will be restored. If the script was empty, it will be empty again. If the button originally said "BOO!" in white italic text on a black background before you started editing the button, that's how it will end up looking.

24.1.2 Testing the buttons

Hinson tip: it is better to find out about problems *before* executing a nonfunctional or faulty macro when anxiously chasing a DX station or during a live QSO, so test every macro button soon after programming it.

With the radio turned off, in TEST mode or at least turned right down to QRPp levels on a vacant frequency, click the button or press the corresponding hotkey to execute the macro and observe its operation.

If there is a problem with a button's operation, edit the script by right-clicking the button and making whatever changes are necessary, then test it again[373].

24.2 Hotkeys (keyboard shortcuts)

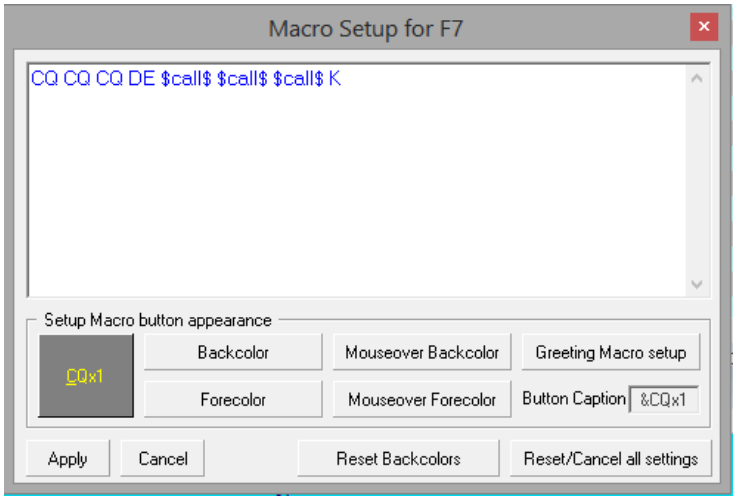
24.2.1 User-definable Alt hotkeys

The hotkey currently assigned to any programmable button is named in the caption of the setup form that opens when you right-click the button (*e.g.* "Macro Setup for F7" means that pressing function key F7 on your keyboard will trigger/run this macro).

The default function keys may be changed as part of the button programming process if you wish. It *may* be easier to remember, say, <**Alt**+**K**> rather than <**Ctrl**+**Alt**+**F6**>.

Assign an Alt+hotkey to a button by using an ampersand (&) in the button caption immediately preceding the character you wish to use as the hotkey.

For example, with "&CQx1" as the button caption, the button will be labeled "CQx1", hinting that <**Alt**+**C**> is the hotkey to trigger that button[374] ▶



[373] After the Nth failed attempt, consider digging out the manual to verify what the script *should* say!

[374] Hotkey Alt+letters are underlined in some menus *etc.* as a visual cue to the applicable shortcuts: this is a Windows convention. Keep an eye out for those subtly underlined letters!

Copyright © 2026 Robert C. Furzer K4CY

24.3 Logger32's default hotkeys

The following is a list of hotkeys active by default within Logger32. Most of these are active while the focus is on the log entry pane.

The following hotkeys are active in the log entry pane, CW Machine and as otherwise indicated.

Hotkey	Function
<**Ctrl**+**A**>	Rotate **A**ntenna to the short path (log entry pane, CW Machine, and Sound card data window) and can be set global.
<**Alt**+**A**>	Rotate **A**ntenna to the long path (log entry pane, CW Machine, and Sound card data window) and can be set global.
<**Ctrl**+**B**>	**B**ookmark (pseudo-spot) a callsign on the BandMaps, DX Spots pane and DX spot map privately, *without* sending it to DX cluster.
<**Ctrl**+**C**>	**C**lear entries.
<**Ctrl**+**D**>	Send a **D**X spot for the callsign currently being logged, or if there is none, the most recently logged callsign.
<**Ctrl**+**E**>	Set QSO **E**nd time.
<**Ctrl**+**F**>	Open the **F**loating callsign window.
<**Ctrl**+**G**>	**G**rab the callsign from the scratchpad.
<**Ctrl**+**H**>	**H**ome all rotators.
<**Ctrl**+**I**>	**I**nternet callsign lookup.
<**Ctrl**+**J**>	Launch **J**TDX/WSJT-X and the UDP BandMap.
<**Ctrl**+**K**>	Start CW Machine to **K**ey the radio.
<**Ctrl**+**L**>	**L**og the QSO.
<**Ctrl**+**M**>	Open the **M**anual add QSO window.
<**Ctrl**+**O**>	Change **O**ffset.
<**Ctrl**+**P**>	Change **P**refix.
<**Ctrl**+**S**>	Set QSO **S**tart time.
<**Ctrl**+**T**>	**T**oggle between Radio #1 and Radio #2. Can be set global.
<**Ctrl**+**U**>	Change **U**ser (operator).

Copyright © 2026 Robert C. Furzer K4CY

Hotkey	Function
<**Ctrl**+**V**>	Open the **V**iew DX map.
<**Ctrl**+**W**>	**W**ipe entries. Can be set global.
<**Ctrl**+**X**>	Swap log entry pane QSO with scratchpad QSO.
<**Ctrl**+**Z**>	Move QSO info to scratchpad.
<**Enter**> or <**Return**>[375]	User-definable in CW Machine only.
<**Ins**> or <**Insert**>	User-definable in CW Machine only.
<**Space**>	User-definable in CW Machine only.

24.3.1 Sound card, data terminal, CW Machine and RCP hotkeys

Hotkey	Function
<**F1**> to <**F12**>	Execute row one of macro buttons.
<**Alt**+**F1**> to <**Alt**+**F12**>	Execute row two of macro buttons.
<**Ctrl**+**F1**> to <**Ctrl**+**F12**>	Execute row three of macro buttons.
<**Alt**+**Ctrl**+**F1**> to <**Alt**+**Ctrl**+**F12**>	Execute row four of macro buttons.
<**Esc**> or <**Escape**>	Changes to receive *provided* the focus is on the Sound card data window.
<**Pause**> or <**Break**>	Toggles between transmit and receive (either way) *provided* the focus is on the Sound card data window.
<**Ins**> or <**Insert**>	Buffers incoming decodes and turns the background white while you browse the now frozen RX window.

[375] The keyboard key may be labelled either <**Enter**> or <**Return**>. 'Return' reminds me of using electric typewriters and teletype terminals, where the key sends the print head to the start of the line and rolls the paper up a line *i.e.* a **C**arriage ***R**eturn* + **L**ine **F**eed (CR/LF). Yes, I am *that* ancient. I remember those stacks of punched cards too, although the Difference Engine was a little before my time.

Copyright © 2026 Robert C. Furzer K4CY

24.3.2 Logbook editing hotkeys

These hotkeys are active while you are editing QSOs in the logbook.

Hotkey	Function
<**Tab**>	Finish editing the field[376] and move focus to the field on the right
<**Esc**>	Abandon any editing in progress. Puts the focus on the Call field of the log entry pane.
<**PgUp**>	If editing can be completed successfully, complete editing and move the visible logbook up one page.
<**PgDn**>	As above, but move down one page.
<**End**>	Moves the edit insertion point to the end of the edit text.
<**Ctrl**+**End**>	If editing can be completed successfully, complete editing and move editing to the right most column on the same row.
<**Home**>	Move edit insertion point to the start of the edit text.
<**Ctrl**+**Home**>	If editing can be completed successfully, complete editing and move editing to the left most column on the same row.
<**LeftArrow**>	Move the edit text insertion point to the left until it is at the start of text. It acts as a normal edit key until it is at the start of text. Then, if editing can be completed successfully, it completes editing and moves the editing point to the right end of the cell to the left.
<**Shift**+**LeftArrow**>	If editing can be completed successfully, complete editing and select the cell to the left.
<**RightArrow**>	Move the edit text insertion point to the right until it is at the end of text. It acts as a normal edit key until it is at the end of text. Then, if editing can be completed successfully, then it completes editing and moves editing to the left end of the cell to the right.
<**Shift**+**RightArrow**>	If editing can be completed successfully, complete editing and select the cell to the right.
<**UpArrow**>	If editing can be completed successfully, complete editing and select the row above.
<**DownArrow**>	If editing can be completed successfully, complete editing and select the row below.

[376] 'Finish editing' means Logger32 saves the change made to that field. Therefore, you may notice a slight pause before the cursor moves on while the updated QSO record is safely saved to disk.

Copyright © 2026 Robert C. Furzer K4CY

Hotkey	Function
<Enter>	If editing can be completed successfully, complete editing and move the focus to the Call field of the log entry pane.

24.3.3 Hotkeys to navigate logbook, awards and worked/confirmed tables[377]

Hotkey	Function
<PageUp>	Scroll one screen up
<PageDown>	Scroll one screen down
Mouse wheel	Scroll one line up/down

24.3.4 Global hotkeys[378]

The following hotkeys work as specified when you are actively using Logger32 *i.e.* it has the focus and is running in the foreground. If you tick them on the **Setup ⇨ Hotkeys** menu, they also work as specified even when Logger32 is in the background *e.g.* while you are browsing the web or reading emails.

Hotkey	Function
<Ctrl+A>	Rotate the relevant antenna (the one defined for the current VFO frequency of the active radio) to the short path direction for the station currently in the log entry pane
<Alt+A>	Rotate the relevant antenna to the long path direction
<Ctrl+T>	Toggle back and forth between radios 1 and 2
<Alt+W>	Wipe (empty) the log entry pane
<Home>	Move the focus to the log entry pane

Hinson tip: it may seem convenient to be able to trigger Logger32 to do stuff while you are busy doing other things but be careful, especially with the Logger32 functions that you don't often use. If you forget that you have assigned the global hotkeys, you may be puzzled and annoyed that they don't work as expected in other programs, and surprised to discover Logger32 doing stuff unexpectedly, apparently 'all by itself'.

[377] If you are already at the top, you can't move up. If you are already at the bottom, you can't move down.

[378] Other hotkeys be configured to function while the focus is on other Windows programs, but be careful: if you configure a Logger32 hotkey for global applicability, particularly one of the common ones (such as **<Ctrl+S>** to **S**ave), it may override the hotkeys in other applications, causing unexpected results and a great gnashing of teeth.

Copyright © 2026 Robert C. Furzer K4CY

24.3.5 Drok USB knob hotkeys

A Drok or similar programmable USB knob can be reprogrammed to send the following sequences when the knob is turned clockwise or counterclockwise. For detailed instructions and pictures, see the Programmable USB VFO knobs chapter.

Hotkey	Function
<**Shift**+**Ctrl**+**Alt**+**L**>	QSY Radio 1 VFO A ***N***[379] Hz HF
<**Shift**+**Ctrl**+**Alt**+**R**>	QSY Radio 1 VFO A ***N*** Hz LF

24.3.6 If Radio Control Panel SO2R is *not* open

Hotkey	Function
<**Ctrl**+**UpArrow**>	Move Radio 1 sub ***N*** Hz HF
<**Ctrl**+**DownArrow**>	Move Radio 1 sub ***N*** Hz LF
<**Alt**+**Ctrl**+**UpArrow**>	Move Radio 1 sub-receiver 1 kHz HF
<**Alt**+**Ctrl**+**DownArrow**>	Move Radio 1 sub-receiver 1 kHz LF

24.3.7 If Radio Control Panel SO2R *is* open

Hotkey	Function
<**Ctrl**+**UpArrow**>	Move Radio 2 sub ***N*** Hz HF
<**Ctrl**+**DownArrow**>	Move Radio 2 sub ***N*** Hz LF
<**Alt**+**Ctrl**+**UpArrow**>	Move Radio 2 sub-receiver 1 kHz HF
<**Alt**+**Ctrl**+**DownArrow**>	Move Radio 2 sub-receiver 1 kHz LF

[379] ***N*** is determined by right-clicking the frequency in the log entry pane and selecting a rate

Copyright © 2026 Robert C. Furzer K4CY

24.4 Macros and hotkeys FAQs

Q. Is there a simple method of copying a macro from one sound card digimode to another?

A. Yes, you can copy individual macros from button to button.

For example, set up the top left-hand macro button <**F1**> in (say) the MMVARI running PSK31 and <**Apply**> it. Right-click the button to re-open the macro setup window and, with this window open, change the MMVARI mode to the one you want to copy to (say) RTTY_U - and then click <**Apply**> again. You will find the F1 macro has been copied from PSK31 to RTTY_U. Note that this method *only* copies to the same button position, and you can *only* do one button at a time.

Tedious, yes, but hey it *is* simple …

Q. OK, so is there a method of copying *all* my macros from one sound card digimode to another?

A. Yes, although it is not as simple and easy as doing them one at a time, making it a little risky.

The trick is to close Logger32, then copy the relevant lines in the relevant *.INI* file/s, changing any mode references as applicable, using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad[380].

You can also check and edit the macros quite easily at the same time, although you'd need to save the file/s and run Logger32 to check them out.

… BUT … before you meddle with *.INI* files, be sure to **make backup copies first** just in case your edits don't go *entirely* to plan.

Hinson tip: earn a gold star by also making an offline backup of the *.INI*s *after* you are satisfied that the changes worked, making it easy to restore the file/s later if something else goes wrong with your system. Backup before ⇝ Edit ⇝ Test ⇝ Correct? ⇝ Backup after ⇝ Rest easy.

[380] If that instruction or this FAQ leaves you confused, trust me: this is not an approach you ought to be taking.

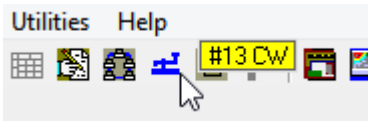
Copyright © 2026 Robert C. Furzer K4CY

25 CW Machine

> "Morse code is the language
> of the telegraph and
> the key to communication"
>
> *William J. Hammer*

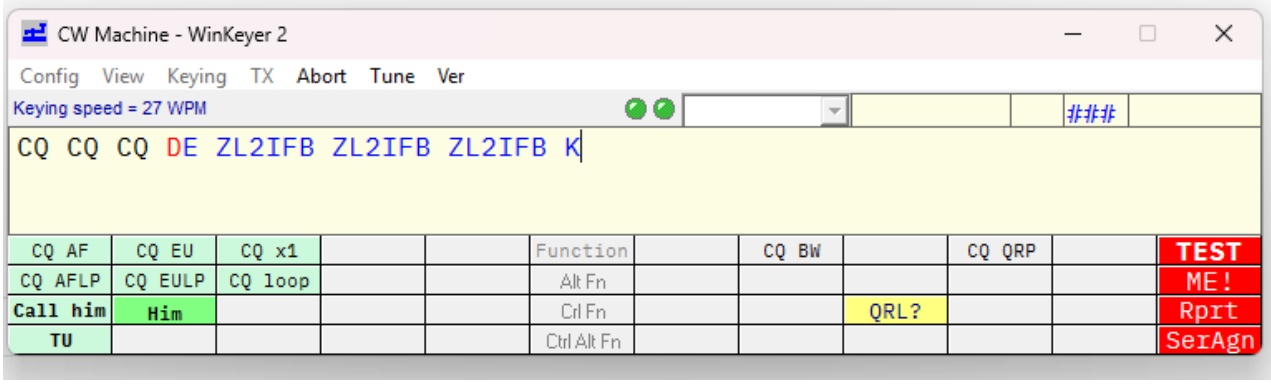
The CW Machine is a separate Morse-sender program, part of the Logger32 suite.

Launch the CW Machine by clicking toolbar icon #13 ▶ or by pressing <**Ctrl**+**K**> (for **K**ey) from the log entry pane. The blue Morse key turns gray as the CW Machine opens.



Hinson tip: the color drains out of all those little icons when the respective windows and functions are open. It can be interesting to click and explore the colorful ones *e.g*. while idly watching your system clock-up boring FT# QSOs. Please finish studying this manual first, though.

The main yellow box ▼ is where you type message text to be transmitted in Morse code[381].



Type as fast as you like: the PC works its way systematically through the message, sending it as Morse code letter-by-letter at your chosen keying speed, turning the blue text red as each letter is transmitted then black once sent. If you are quick, you can even correct typos in the blue text *before* that part of the typed message goes out on the air. Later will be too late.

There are up to 4 rows of 12 buttons that can be color-coded and pre-programmed to send your choice of stock phrases or brag text (like a conventional memory keyer) or to interpret CW macros (*e.g*. substituting the callsign of the station you are currently logging in place of $**Call**$).

Using your imagination and the macros, you can make operating CW easier by pushing buttons and typing rather than squeezing an Iambic paddle, flapping your bug or tapping your Morse key.

---

[381] By default when initially installed, the CW Machine buttons all start out blank nondescript gray. The central macro buttons on this screenshot show the corresponding PC keyboard shortcuts in light text because that's how I labeled those buttons as a reminder for each row. I use the distinctive traffic-light colors to group-together related buttons (*e.g*. the red buttons on the right are typical contest messages).

Copyright © 2026 Robert C. Furzer K4CY

25.1 Connecting and configuring the CW Machine

First, choose which of these three ways you want to generate and send your CW:

1. The CW Machine toggles a control line on a serial or parallel port on the PC, which in turn is connected to the radio's 'Morse key' input through a simple **interface** (software keying);
2. The CW Machine sends message text to a keyer **device** such as a WinKeyer, which is connected to the PC via a serial port *and* to the radio's 'Morse key' input (hardware keying); or
3. Logger32 sends the relevant CAT commands and message text through the CAT connection to a CAT-enabled keyer **function** built-in to the radio (CAT keying). Does not use the CW Machine.

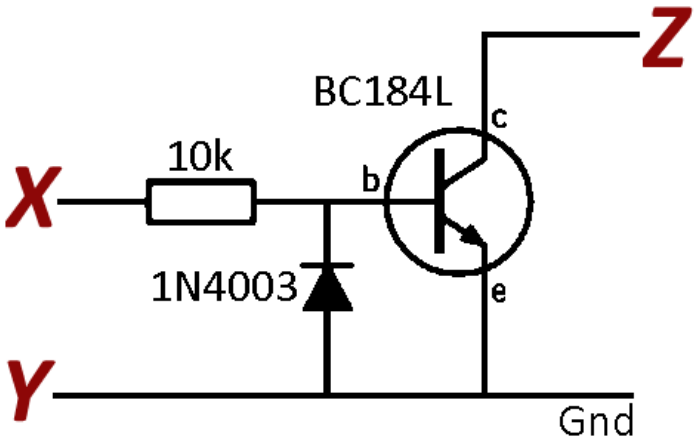
So, how do you want to do it: **interface**, **device** or **function**?

25.1.1 Software keying with an interface

Using software keying, the CW Machine asserts (raises the voltage) on specific control pins on a serial or parallel PC port, to put the transmitter into transmit (PTT) and to send the CW/carrier signal (keying). Since the radio PTT and 'Morse key' inputs have to be *grounded* to activate, you need an inverting interface between the PC port and your radio. Two designs follow.

First, a basic single-transistor interface for a single control line ▼

You will need one for keying *plus* maybe another for PTT control unless VOX works OK on your radio. The design presumes that your radio's 'Morse key' input is normally at a positive voltage until grounded to key the transmitter. Most are. None of the components is particularly critical, making this a classic junk-box project. For example, although a BC184L is specified on the schematic, almost any NPN switching transistor will probably work. The diode isn't even needed for a parallel port interface.

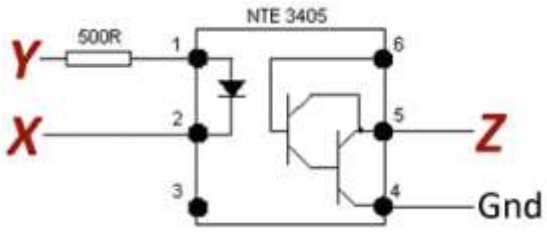


PC port type	Function	X	Y	Z
Serial (COM)	CW keying	DTR line (DB9 pin 4 or DB25 pin 20)	Ground (DB9 pin 5 or DB25 pin 7) plus radio ground	Straight Morse key input on transmitter
	PTT	RTS line (DB9 pin 7 or DB25 pin 14)		PTT input on transmitter
Parallel (LPT)	CW keying	SELIN line (DB25 pin 17)	Ground (DB25 pins 18 to 25) plus radio ground	Straight Morse key input on transmitter
	PTT	INIT line (DB25 pin 16)		PTT input on transmitter

◀ Connections to points **X, Y** and **Z** are shown in the table, using the conventional serial port setup with CW keying on DTR and PTT on RTS (you can use the converse if you are unconventional).

Copyright © 2026 Robert C. Furzer K4CY

Alternatively, here is an opto-isolated interface with just two components plus the appropriate connectors ▶

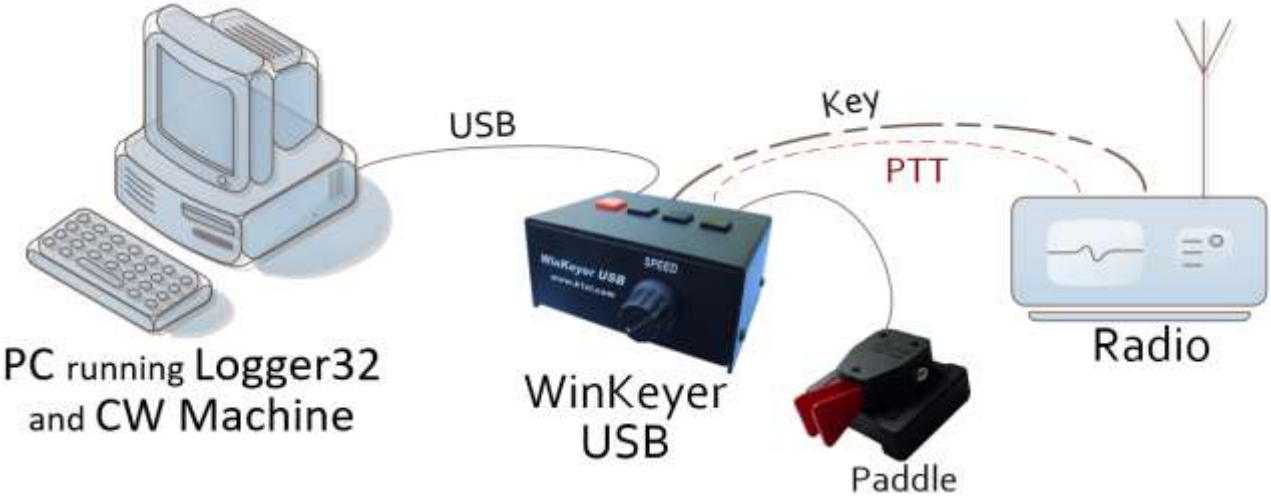


With care, you may be able to squeeze either interface circuit inside the shell of the DB9 or DB25 connector, with flying leads terminated in the appropriate plugs for the 'Morse key' and/or PTT inputs on your radio. Use strain relief on the cable/s, though, to avoid breaking the leads to the transistor or opto-isolator (at the worst possible moment, naturally!).

See below for advice on configuring software keying.

25.1.2 Hardware keying with a device

Logger32's CW Machine supports both the original K1EL WinKeyer and the upgraded WinKeyer2 with additional features and functions.



Various manufacturers of transceivers and peripherals have either embedded genuine WinKeyer chips into their devices, or emulated the commands and functions (presumably under license, hopefully faithfully). Before attempting to use any type of hardware keyer with Logger32, physically connect it to your paddle, PC and radio by following the manufacturer's instructions, and get it working as a standalone CW keyer. This may be as simple as plugging things in, although it may require you to install specific device drivers and WinKeyer configuration software on the PC. This *Logger32* User Manual only concerns the *Logger32* parts of the configuration …

For detailed information on the features, functions, parameters, commands and timing of Morse code, study the excellent WinKeyer documentation available on K1EL's website (*e.g.* the WinKeyer 2 datasheet) plus the manual for your particular device.

Hinson tip: a hardware keyer based on a dedicated microcontroller gives more precise and reliable CW timing than with software keying. Your outgoing Morse messages shouldn't be slowed or interrupted by the arrival of emails, backups, antivirus scans or whatever else happens to tie up your PC at that precise moment. The CW machine sends text characters and single-byte instructions to the WinKeyer in quick data bursts, which buffers them, then generates and sends well-formed CW consistently with the correct speed and timing.

See below for advice on configuring WinKeyer/hardware keying.

Copyright © 2026 Robert C. Furzer K4CY

25.1.3 CAT keying with a function

Some modern radios accept specific CAT commands and text, converting the text into Morse code and transmitting that. For example, the Elecraft K3 has the **KY** CAT command:

KY (CW or CW-to-DATA Keying from Text; GET/SET)

SET format: **KY*[text];** where * is normally a BLANK and **[text]** is 0 to 24 characters. If * is a **W** (for "wait"), processing of any following host commands will be delayed until the current message has been sent. This is useful when a **KY** command is followed by other commands that may have side-effects, e.g., **KS** (keyer speed).
Basic RSP format: **KYn;** where **n** is 0 (CW text buffer not full) or 1 (buffer full). Also see **TB** command.
K2 Extended RSP format (K22): **KYn;** where **n** is 0 (buffer < 75% full), 1 (buffer > 75% full), or 2 (buffer completely empty AND transmit of previous string is complete.

The following keyboard characters are mapped to CW "prosigns":

(KN + AR = BT % AS * SK ! VE

In addition to these prosigns, these special characters can be inserted anywhere in the **KY** command text:

< Puts the K3 into TX TEST mode, until a '>' character is received
> Returns the K3 to TX NORM mode
@ In CW mode, this character normally terminates any CW message (via KY or manual send), emulating the K2. However, tapping **2** in *CONFIG:CW WGHT* changes '@' to a prosign: the 'at' sign as used in e-mail addresses. This is the newest Morse Code character; it can be remembered as the prosign 'AC' (as in "the At Character").
^D (EOT, ASCII 04) Quickly terminates transmission; use with CW-to-DATA.

So, we can compose and send the KY command and text to a K3 via CAT, using the **R**adio **C**ontrol **P**anel. Programming "KY CQ CQ CQ DE K4CY K4CY K4CY +K", for instance, into one of the RCP buttons would let Bob press the button to send a CQ message via CW on his K3, if only he had one. He would probably also program further RCP buttons to alter the K3's CW speed or other parameters, perhaps getting the K3 to send RTTY instead of CW in much the same way.

Hinson tip: CAT keying has nothing to do with the CW Machine, really. It is included here for completeness, satisfying my innate nerdiness. Feel free to experiment if you, too, are a nerd.

25.2 Configuring and using the CW Machine

Launch the CW Machine using the Morse key icon #13, then use top line menu to set things up.

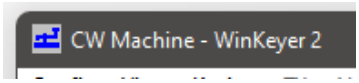
25.2.1 CW machine main menu

- **Config**: see below.
- **View**: see below.
- **Keying**: see below.
- **TX**: transmit text. This is only available if keying is set to <**Manual TX (MOX)**>.
- **Abort**: stop transmitting immediately and clear the transmit buffer. **<Esc>** does the same.
- **Tune**: keys the radio, sending a solid carrier for up to 30 seconds. Click once to key-down, click again (or wait patiently for the 30 seconds to elapse, or **<Esc>**) to key-up.
- **Ver**: displays the version number of the CW Machine.

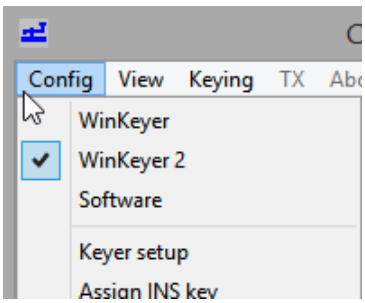
Copyright © 2026 Robert C. Furzer K4CY

25.2.2 CW Machine *Config* menu

The configuration options vary according to the whether you are using hardware or software keying … so …

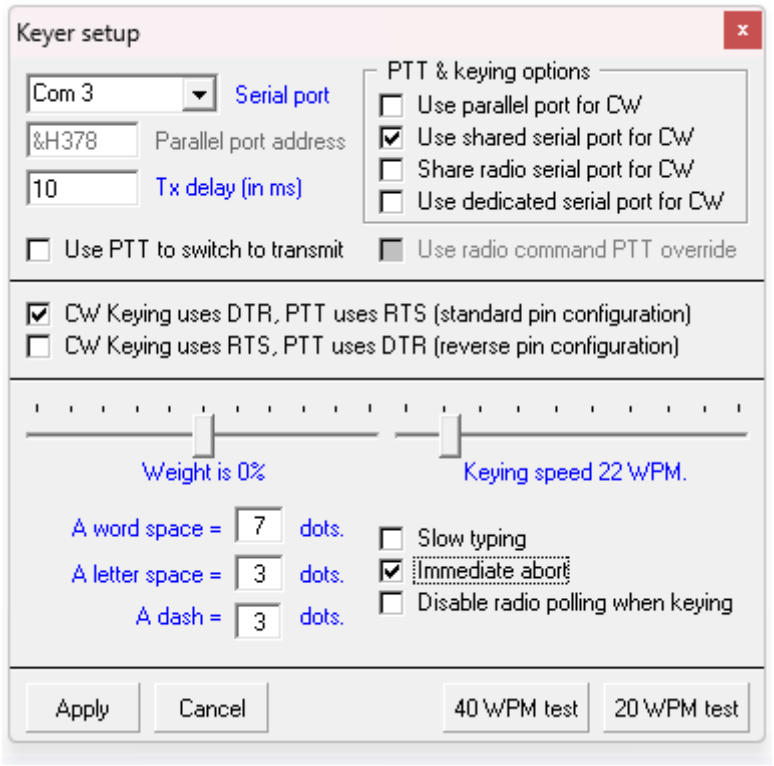
> **Hinson tip**: if you forget which form of keying you are using, check the CW Machine's window caption ▼ for a clue.
>
> 
>

Tick <**WinKeyer**> , <**WinKeyer2**> or <**Software**> as applicable ▶



25.2.3 CW Machine config with *software* keying

Now configure the CW Machine from its <**Config>** menu. Here are the instructions to configure ***software*** keying (skip to the next section for clues on how to configure ***hardware*** keying):



- **Keyer setup**: opens a configuration form with a confusingly-worded range of options ▶

  The CW machine can 'key' the transmitter by using a control line on a serial (COM) or parallel (LPT) port. While the CW Machine can key the port control lines, a simple interface (an inverting switch) is needed between the PC port and the 'key' input to the radio so build or buy and connect the interface to the PC port, following the earlier schematics and instructions.

  Then configure the CW Machine accordingly:

  - Tick <**Use shared serial port for CW**> plus <**CW Keying uses DTR, PTT uses RTS (standard pin configuration)**>[382], then select the correct numbered serial (COM) port from the drop-down list, if you are using a serial port to key your radio.
  - Alternatively, tick <**Use parallel port for CW**> and confirm/correct the hexadecimal address for the LPT port.

Hints below to figure out the port number or address

[382] Unless, for some reason, you chose to swap those two over – in which case tick the 'reverse pin configuration' line instead.

Copyright © 2026 Robert C. Furzer K4CY

For **P**ush-**T**o-**T**alk, determine whether your radio automatically goes into transmit whenever the 'Key' input is closed in CW mode, and reverts automatically to receive a short while later (as with VOX on voice modes).

- If the radio handles PTT itself, you can ignore the PTT settings in the CW Machine: simply send CW and let the radio do its thing[383];
- If you need the CW Machine to handle PTT for your radio:
  - If you are using CAT, tick <**Use PTT to switch to transmit**> *and* <**Use radio command PTT override**>, instructing Logger32 send the PTT commands to the radio via CAT. If this works with your setup, you do not need a PTT interface.
  - Otherwise, build/buy and connect a simple PTT interface to a control line on the same serial or parallel port in the same way as for keying. Typically that means keying on DTR and PTT on RTS for a serial port, or keying on SELIN and PTT on INIT for a parallel port. Also tick <**Use PTT to switch to transmit**>, instructing the CW Machine (as opposed to the radio itself) to control PTT.

Complete the Keyer setup form:

- <**Tx delay**> tells the CW Machine to assert PTT *before* starting to send CW, giving your radio a chance to change-over from receive to transmit. Older steam-powered radios with clunky relays may need a breather of 20 milliseconds or more in order to avoid truncating the first Morse character of an over. Modern radios may get away with a delay of 5 milliseconds or less, right down to zero for true QSK systems.
- Set the **weight** (CW duty cycle), **keying speed** and **timing** of the CW being generated.

  Hinson tip: take care with these adjustments. Badly timed CW can be a struggle to copy. Why make things unnecessarily harder for your QSO partners? The values shown on the Keyer setup form above are generally considered standard, conventional … and best.

- <**Slow typing**> holds the PTT closed while you hunt-and-peck at each letter on the keyboard. **The radio stays in transmit** and does not automatically switch to receive unless/until, having triumphantly completed typing your message, you click <**Abort**>, hit <**Esc**> or execute the $**Receive**$ macro … so don't forget to do that.

  Hinson tip: avoid this setting unless you really *are* a S L O W typist who is easily distracted by the radio's TX/RX relays and receiver audio: someone might be trying to interrupt!

  [This only works if the CW Machine has PTT control: if you are entirely reliant on the radio's VOX for PTT, look for a control knob or menu setting for the VOX delay, somewhere on the radio.]

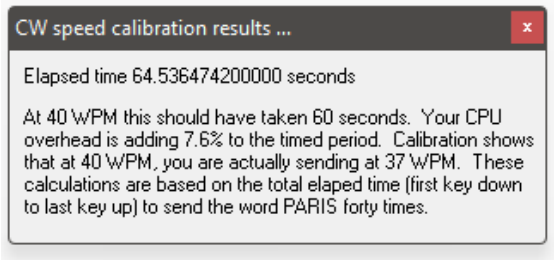
- <**Immediate abort**> means pressing the <**Esc**> key *instantly* stops sending, even while the system is still sending a Morse charact… [Again, this only works if the CW Machine is controlling PTT. If the radio's VOX is in charge, good luck finding a way to interrupt it in a hurry.]
- <**Disable radio polling when keying**> addresses a problem with the TEN-TEC OMNI V and possibly under-powered radios that struggle to respond to the stream of CAT polling requests by Logger32 *while* also sending CW. Evidently, they can't walk and chew gum.

[383] Actually, be sure that <**Use PTT to switch to transmit**> is *un*-ticked to avoid the CW Machine holding on to the PTT after the radio's VOX has already let go, perhaps causing some confusion and consternation.

Copyright © 2026 Robert C. Furzer K4CY

- The two test buttons at the bottom of the Keyer setup form confirm your PC's actual keying speed by measuring the time actually taken to compose (but not transmit) a series of 20 or 40 standard words ("PARIS ") ▶

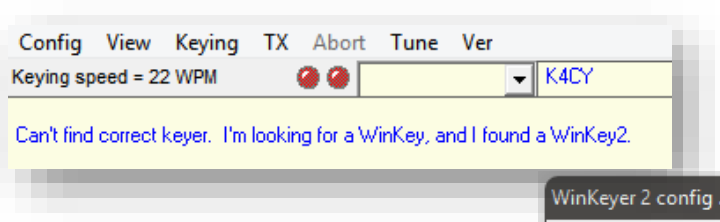
  If you change your mind *during* a speed test, click the CW Machine's <**Abort**> menu to stop prematurely … and ignore the perfect speed results reported incorrectly.



Hinson tip: if the CW generated by your setup is inconsistently timed (noticeably faltering or stuttering) or markedly slow, try disabling radio polling to see if it helps. It is also worth checking in Windows Resource Monitor that the PC is not heavily loaded by other apps interfering with Logger32 and the CW machine: close unnecessary apps and browser windows to reduce the CPU utilization. If nothing improves the CW timing, consider investing in a hardware keyer such as a WinKeyer instead of software keying, or possibly a better PC or radio. Or just hope that nobody notices on-air. For some, that 'hand-sent CW' rhythm has a certain cachet and charm. For pedants, it's just annoying.

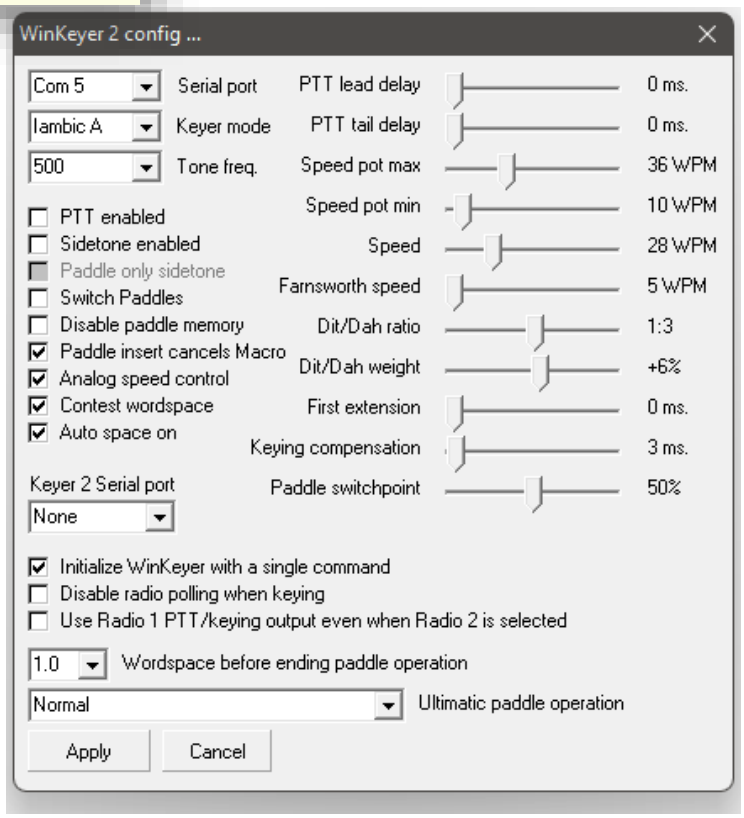
25.2.4 CW Machine config for *hardware* keying with WinKeyer 2

This and the following section explain how to configure **hardware** keying using a genuine WinKeyer or WinKeyer2 device from K1EL, or a keyer device that embeds an actual WinKeyer chip or otherwise emulates a WinKeyer. The configs differ slightly according to which type of device you are using, so here is the most popular first, the Winkeyer2 …



◀ Not sure whether yours is a WinKeyer or WinKeyer2? Pick one: if you chose wrong, the CW Machine politely tells you.

Keyer setup (with WinKeyer 2 hardware keying) opens the WinKeyer 2 config form offering the following extensive array of CW options and keyer settings ▶



- **Serial port**: click the selector button then find and click the PC serial port to which your main or only WinKeyer 2 device is connected.
- **Keyer mode**: the choices are "Iambic A", "Iambic B", "Ultimatic" or "Bug Mode", affecting the timing and generation of CW. The details are beyond the scope of this Logger32 User Manual, so either study the subject or simply try out the different modes to discover what works for you. Or doesn't.

Copyright © 2026 Robert C. Furzer K4CY

- **Tone freq**: what frequency sidetone should the WinKeyer 2 device generate, if at all?
- **PTT enabled:** most radios have a VOX-type control functional that automatically starts them transmitting whenever the Morse key input is closed. With "semi-breakin", the radio *remains* in transmit for a short while after key-up, before returning to receive. With "full-breakin" (QSK), it returns to receive even in the short breaks between elements of a CW character. With this option ticked and with its PTT control line correctly interfaced to your radio, the CW Machine holds the PTT closed while you are sending a CW message from the PC, similar to semi-breakin. Untick it if you prefer to listen for interruptions *while* you are sending.
- **Sidetone enabled:** the CW Machine can generate a sidetone. Read the WinKeyer documentation re its built-in sidetone as it is related to the keying pin in use at the time. Similarly, the *micro*HAM *micro*KEYER has a configurable built-in sidetone, as do virtually all rigs so this function seems largely redundant – except perhaps if you are operating your station remotely, and need the local PC to generate the sidetone since the remote station is out of earshot.
- **Paddle only sidetone:** means the sidetone remains silent while the keyer is sending a stored message or macro, but sounds when you use the paddle to send CW manually making it easier to compose your Morse code. This option is grayed out and unavailable unless sidetone is enabled.
- **Switch Paddles:** tick this to send dahs with the *left* paddle and dits with the *right*. Untick it to return to 'normal' (meaning conventional, but it's not the law).
- **Disable paddle memory**: some CW ops like their dot and dash memories, some don't.
- **Paddle insert cancels Macro**: while the CW Machine is sending a message, a tap on the paddle can interrupt and cancel the sending with this option enabled: handy as an 'emergency stop'!
- **Analog speed control:** tick this if you prefer to use the physical knob on your WinKeyer box to adjust the keying speed (WPM). Un-tick it if you would rather adjust the keying speed digitally by left- or right-clicking CW Machine's speed indicator. This is an either-or setting, not both.
- **Contest wordspace:** reduces the space between words just a fraction for rabid contesters desperate to make as many QSOs as humanly possible in the time allowed, regardless of more conventional/standardized Morse timing. Avoid this setting if you are a purist concerned to maximize the integrity of your CW communications.
- **Auto space on:** the clever WinKeyer2 chip does its level best to prevent us running all our dits and dahs together by caching any characters we may have paddled too closely together (provided they are separated by at least 1 dit space), sending them in sequence with the proper 3 dit inter-character spacing.
- **Keyer 2 Serial port**: click the selector button then find and click the PC serial port to which your *secondary* WinKeyer 2 device is connected. Genuine K1EL WinKeyers each have two sets of CW and PTT outputs, and can switch between them under PC command. With two of these keyers connected, you can presumably switch your CW keying and PTT to any of 4 radios.
- **Initialize WinKeyer with a single command**: send all 15 configuration parameters needed to set-up the WinKeyer as one long command string. We believe this works properly for *genuine* WinKeyers. Users of WinKeyer emulators such as RemoteRig and RigExpert TI-5000, however, may need to *un*tick it, initializing the keyer one parameter at a time, giving the lazy device a chance to keep up. Stop and restart the CW machine after changing this setting. Watch the WinKey debug window to see the initialization command/s being sent.
- **Disable radio polling while keying:** some older, slower radios couldn't deal with CAT polls and commands while they were transmitting. Most today can walk *and* chew gum.

Copyright © 2026 Robert C. Furzer K4CY

- **Use Radio 1 PTT/keying output even when Radio 2 is selected**: tick this to enable and use the 'Radio 1' PTT and CW connectors on the WinKeyer device always, *regardless* of which radio is currently selected in Logger32. Use this if your radio switching is handled by an SO2R controller device. The SO2R controller is simply directed to use radio 1 or radio 2 through its serial connection to Logger32, and it then handles all the downstream sequencing and switching of receiver audio, paddle/microphone inputs, transmit-lockouts *etc*. See the SO2R section for more details. *Un*tick this if you simply have radio 1 directly connected to the radio 1 CW and PTT outputs on the WinKeyer, and radio 2 connected to the radio 2 outputs. If you are only using one radio, connected to the Radio 1 outputs on the WinKeyer, ticking this option means it will still be keyed even if for some strange reason you select Radio 2 in Logger32.
- **Wordspace before ending paddle operation**: the WinKeyer switches over from PC/data to paddle input when you start using the paddle, and switches back to PC/data input when you stop using the paddle. This setting tells it how long to wait after you stop paddling before changing back. It needs a bit of time to switch over, so don't set this too short or you will find your paddling is not generating your usual high-quality well-timed Morse because the WinKeyer is constantly switching modes as you paddle instead of generating CW. Conversely, if you set the delay too long, the WinKeyer may ignore or corrupt the very first character or command sent by the PC after you stop paddling.
- **Ultimatic paddle operation:** when the paddles are squeezed together, an Ultimatic keyer can send a sequence of just dits or just dahs, rather than the usual alternating sequence.
- **Sliders for CW and PTT timing, speed etc**: click-and-drag the relevant sliders to the left or right until the following options are set to your liking …
    - **PTT lead delay**: sets the lead-in time between asserting PTT and generating CW, giving your radio just enough time to go into transmit and settle down before being required to start generating and transmitting the carrier signal for the first bit. Increase this value if you know (by monitoring or measuring your own signal) or suspect (*e.g*. if your callsign is often busted on that first character) that your Morse is being truncated at the beginning of each transmission. This is more likely with older, slower radios using relays for transmit/receive switching, and in complex stations with amplifiers, antenna relays, mast-head preamps, sequencers or whatever, and less likely with modern, faster radios competently designed to cope with QSK CW and digital modes.
    - **PTT tail delay**: increase this setting if your radio needs to be held in transmit at the end of a transmission, giving it time to complete sending Morse characters before dropping the PTT. If you notice the radio or amplifier TX/RX relays clattering *during* spaces within your messages, try adding some PTT tail delay – maybe 50 milliseconds – or use automatic semi break-in on the radio, or invest in a full break-in radio and amp.
    - **Speed pot max**: *if* you are using analog speed control, what CW speed in WPM do you want when the speed control potentiometer is turned fully clockwise, "all the way to 11"? Set this at least as fast as you are comfortable to send accurately on a good day when fully caffeinated, and then some if you are prepared to be stretched a little on occasion. If you simply set it to the maximum that the WinKeyer allows, you may find it tough to set the pot to precisely the right speed you want due to its extended range.
    - **Speed pot min**: *if* you are using analog speed control, what CW speed in WPM do you want when the speed control potentiometer is turned fully anticlockwise? What is the slowest CW speed in WPM that you are comfortable to send and receive?

Copyright © 2026 Robert C. Furzer K4CY

> **Hinson tip**: personally, I find *really* slow CW quite difficult to understand because I find myself distracted by the individual dits and dahs, rather than hearing complete Morse characters. It is better, for me at least, to send slow CW characters at a slightly higher speed but w i t h e x t e n d e d g a p s (the Farnsworth technique).

- **Speed**: this is the current CW sending speed in **W**ords **P**er **M**inute. Clicking and sliding this slider does change your CW speed but it is generally easier to turn the speed knob on the front of most WinKeyer devices if you are physically able to do so. Not so easy if you are operating the station remotely, or if you are missing a functional arm.
- **Farnsworth speed**: when sending above this speed setpoint, the WinKeyer operates normally. Below this speed, the characters are sent at the defined Farnsworth speed but with e x t e n d e d c h a r a c t e r s p a c i n g to get down to the chosen WPM.
- **Dit/Dah ratio**: while a dah is normally 3 dits long, it can be shortened or extended with this setting. Take it easy with this adjustment: as you slide away from the conventional timing, your CW becomes harder to copy reliably. Your Morsing might be more 'distinctive' – more memorable, perhaps, but not necessarily happy memories.
- **Dit/Dah weight**: whereas the dit/dah ratio setting only affects the length of the dahs, weighting affects the dits as well. Again, be gentle with this adjustment. If the weight is *too* heavy, the gaps between bits are harder to discern. Your signal edges ever closer to becoming a continuous carrier. Conversely, *excessively* lightweight keying sounds "dotty" or "scratchy". Either way, it is harder to copy than well-formed conventionally-timed CW.

> **Hinson tip**: increasing the weight *a little* may compensate for the character shortening caused by QSK, especially at higher speeds, but it's easy to go past the sweet-spot. Either monitor your transmitted signal (*e.g.* on a remote web-based receiver, using audio software to plot and measure your CW) or ask experienced and competent CW operators for comments on your CW quality as you cautiously adjust this setting. Better still, adjust the **Keying compensation** setting instead …

- **First extension**: after the PTT is asserted, the WinKeyer can lengthen the very first bit just a little to compensate for slow RX/TX changeovers. This is unlikely to be worthwhile or noticeable at slow to medium speeds but at higher speeds (>25 WPM), it can prevent the first bit being shortened, perhaps even lost altogether.
- **Keying compensation**: this setting lets you extend your dits and dahs by a fixed amount (say 2 milliseconds), *regardless* of the sending speed. This adjustment is a better way to fix those little QSK TX/RX delays than heavier weighting since it worth at any speed but, as always, take it easy and check/measure the effect on your transmitted signal.

> **Hinson tip**: the appropriate setting depends on your station 'system', particularly the transmitter. So, if you change radio, you should probably change the compensation and maybe other timing adjustments. If you don't have the time, equipment or inclination to do those checks/measurements and cautious adjustments, simply **reset the WinKeyer to the default settings**, maybe with *slight* tweaks to get that unique sound you're after.

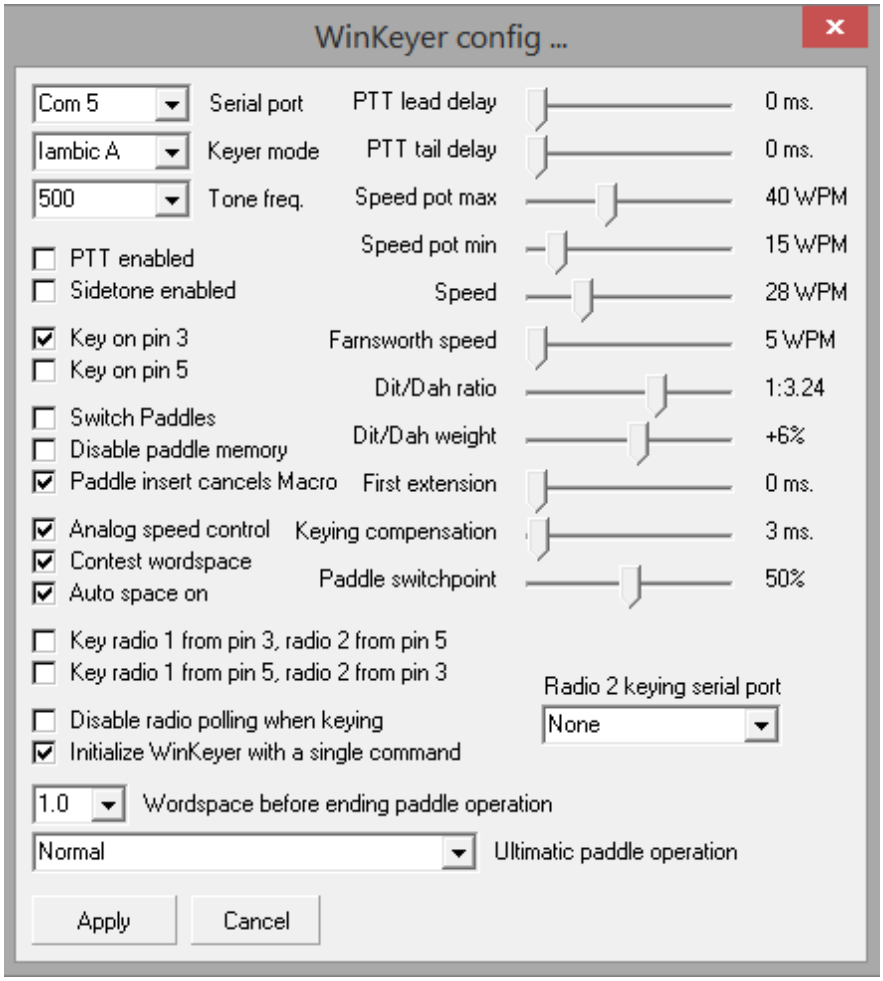
- **Paddle switchpoint**: having pressed one of the paddles, the WinKeyer waits expectantly for the next paddle press. Are you going to press the same paddle again for another dit or dah, press the other paddle for the opposite of the one it is currently sending, or squeeze both paddles together? The switchpoint setting tells the WinKeyer how long to wait: by default the wait-time is one dit-length. You can paddle faster, 'getting ahead of the keyer' by reducing the delay. A little. Don't over-do it.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: learn about the fascinating history of automated CW sending and the different methods of keying that have been invented over the years in an *exceptionally* well-researched and written guide by DJ5IL "All about squeeze keying". For more practical advice, study "The FOC Guide to Morse Code Proficiency" by other FOC friends, collated by me ZL2iFB.

25.2.5 CW Machine config for *hardware* keying with WinKeyer [1]

Here is the configuration form for WinKeyer (the original) ▶



Some of the WinKeyer 2 options are not available, and there are extra choices relating to SO2R setups:

Key Radio 1 from pin 3, Radio 2 from pin 5

– OR –

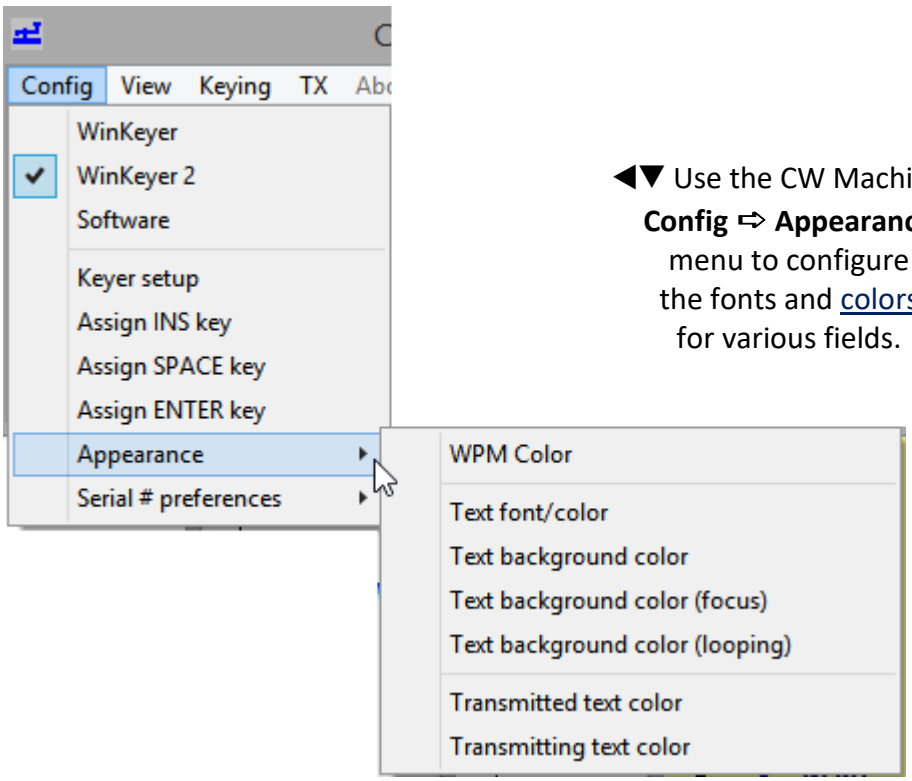
Key Radio 1 from pin 5, Radio 2 from pin 3

Also we can define which serial port will be used for keying Radio 2 (Radio 1 uses the serial port defined at the top of the form).

For both WinKeyer and WinKeyer2, consult the manuals to figure out what the sliders do … or simply adjust them and figure out what changes. If you notice no difference, reset them to the original value. Yes, make a note of the original values before you play around.

Hinson tip: an easy way to record the original settings is to photograph the configuration screen on your smartphone, definitely before and ideally after you finish fiddling with things. The same tip applies elsewhere in Logger32. Alternatively, you are welcome to take on the editing/updating of this User Manual: many of the screenshots littered throughout record *my* settings!

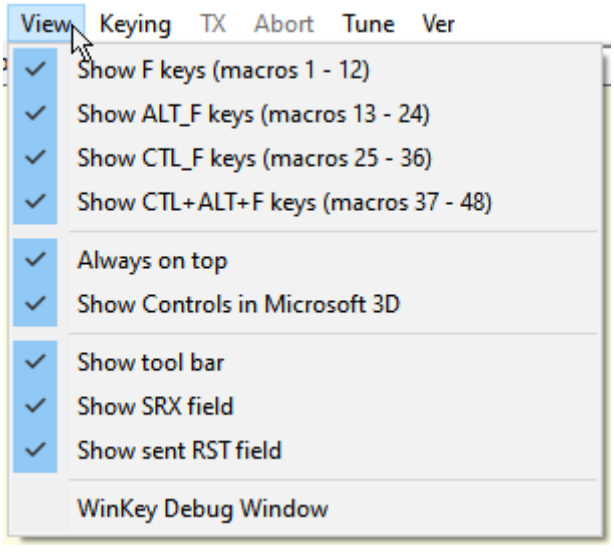
Copyright © 2026 Robert C. Furzer K4CY



◀▼ Use the CW Machine's **Config ⇨ Appearance** menu to configure the fonts and colors for various fields.

- **Serial # preferences:** choose to send the letter "T", the letter "O" or the full five-dashes when transmitting a zero in contest serial numbers. Contesters generally understand T … and many are happy to drop the leading zeroes (flagrantly disregarding any contest rules specifying that serial numbers are to be "*at least 3 digits"*!).

25.2.6 CW Machine *View* menu

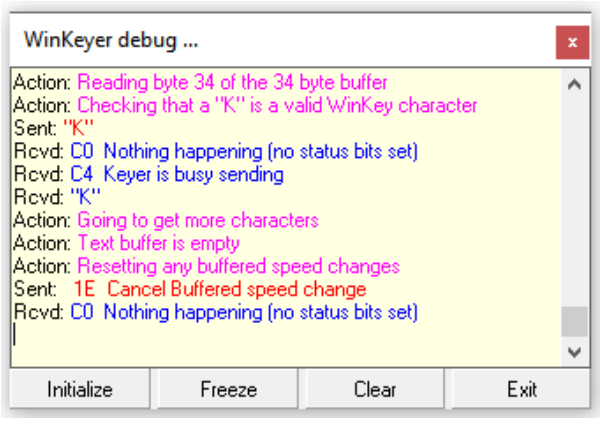


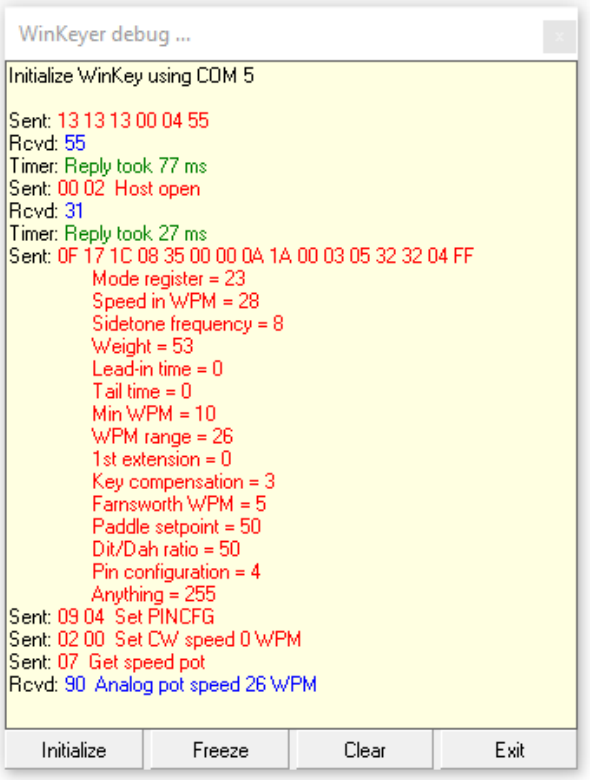
- **Show F keys (macros 1-12)**: displays the first row of twelve CW macro buttons.
- **Show Alt_F keys (macros 13-24)**: displays the second row of buttons.
- **Show Ctrl_F keys (macros 25-36)**: displays the third row of buttons.
- **Show Alt+Ctrl+F keys (macros 37-48)**: displays the fourth and final row of buttons.
- **Always on top**: the CW Machine window always remain visible.
- **Show Controls in Microsoft 3D**: displays the MS Visual Basic controls in with *stunning* 3D effects (well, OK, shading to appear a little more like actual buttons *etc.*).
- **Show tool bar**: display the tool bar or hide it. The CW Machine's tool bar shows the keying speed, TX status LEDs, DX callsign from the log entry pane, current serial number.

Copyright © 2026 Robert C. Furzer K4CY

- **Show SRX field**: display the received serial number entry pane. This pane normally shows '###'. To use it, click or tab into it and type the received serial number: there is no need to delete the '#' symbols.
- **Show sent RST field**: display the sent report.
- **WinKey Debug Window**: display the low-level communications as Logger32 sends commands and text and [hopefully!] receives responses back from the attached WinKeyer ▶
  - Sent: lines are the content or text sent by Logger32 to the WinKeyer.
  - Rcvd: lines are the responses from the WinKeyer to Logger32.
  - Action: lines show Logger32 processing the messages and macros, deciding what to do.

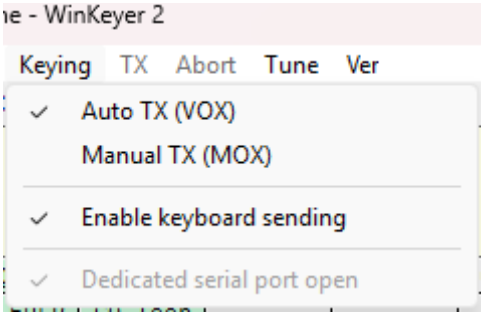
  Click and drag any corner to resize the WinKey debug pane, as usual. Notice the buttons across the bottom of the debug pane:
  - **Initialize** sends an initialization sequence to the WinKeyer, prodding it to disgorge its configuration settings ▶
  - **Freeze** halts the debug display so you can read it and maybe grab a screen shot.
  - **Clear** empties the debug display.
  - **Exit** closes the WinKey debugger.





25.2.7 CW Machine *Keying* menu

- **Auto TX**: if it isn't already sending something else (such as a message triggered by one of the macro buttons), the CW Machine will immediately start sending whatever text you type into the transmit window. Otherwise, the text is held temporarily in the transmit buffer, waiting its turn.



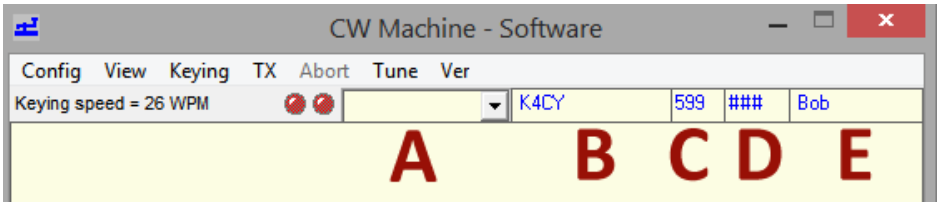
Hinson tip: if you type quicker than the Morse code is sent, you may be able to go back and correct typos in your message *before* it gets sent. The backspace key erases the previous character, whereas clicking the left arrow key moves the cursor back through the message without deleting.

- **Manual TX**: the contents of the transmit buffer will be sent on selecting "TX" (see below).
- **Enable keyboard sending**: lets us use <**Ctrl+K**> to open the function and start typing a message to send as CW.

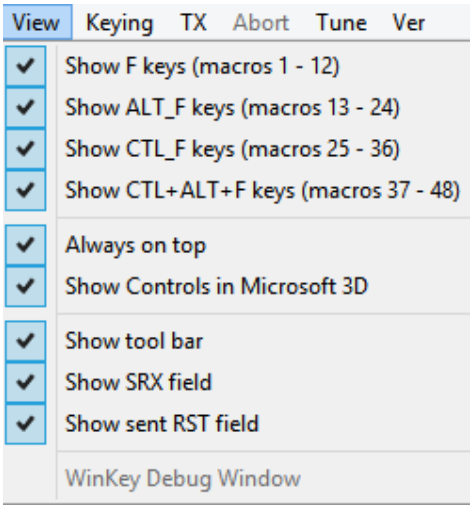
Copyright © 2026 Robert C. Furzer K4CY

- **Dedicated serial port open**: keying will be applied to the port as set in the Config menu. With it unchecked, the PTT and keying LEDs will flash but the port will remain inactive. *This option is only available if you are using a dedicated port in the **Config** ⇨ **Keyer setup** menu.*

25.3 Using the CW Machine



If <**Show tool bar**> is enabled in the CW Machine's <**View**> menu ▼



… ▲ these five fields are shown above the main text area:

A. This is a pull-down selector list of callsigns called or worked recently using the CW Machine[384].

B. The callsign of the DX station you are currently working (if any). This callsign populates the $**Call**$ macro and the Call field of the log entry pane.

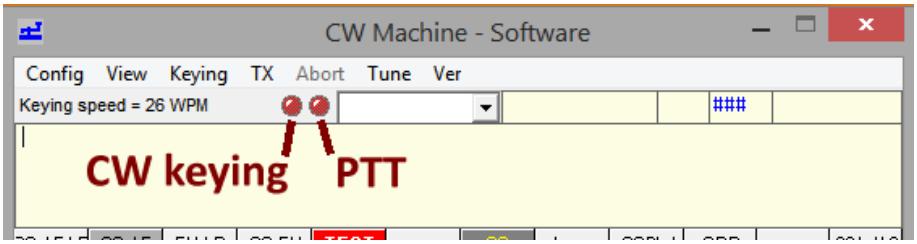
C. Your outbound signal report to the DX station. It populates the $**RST**$ macro and the sent report field of the log entry pane.

D. The serial number you receive from the DX during a contest. It populates the $**SRX**$ macro (useful in a contest situation where you need to confirm receipt of the serial number you just received by repeating it back to the sender).

E. This is a 'free field' meaning that, once configured, it can be filled with any ADIF data. By default, it shows the DX station operator's name.

The meaning of the two normally red "LEDs" on the CW Machine's tool bar differs between hardware and software keying.

With **software keying**, the left "LED" goes green during key-down. The right one goes green only while PTT is asserted by the CW Machine ▶



With **hardware keying**, the left one goes green to indicate that Logger32 is connected to and communicating with the keying device. The right "LED" uses different colors to tell you that the WinKeyer is:

- Generating and sending CW (when green);
- Idle (red);

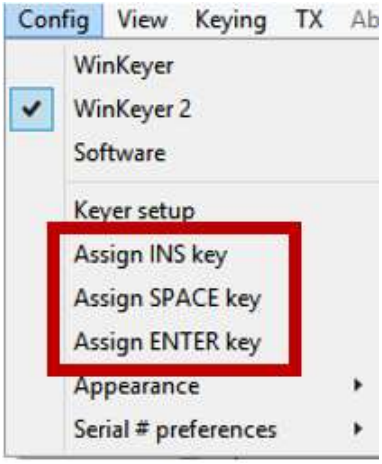
[384] Useful if, say, someone calls you back in a contest for a repeat, or someone from your pileup that you have called unsuccessfully tries again (or whatever). Click their callsign from the list to pop it into field B.

Copyright © 2026 Robert C. Furzer K4CY

- Being keyed with the paddle attached to the WinKeyer (blue); or
- Sounding an operator message on its sidetone (orange).

25.3.1 Assign INS | SPACE | ENTER keys

The CW Machine <**Config**> menu has options to 'assign' the INSert, SPACE and ENTER keys ▶



When you are using your PC keyboard as a CW sender, you can hit the <**Insert**> key, <**Space bar**> or <**Enter**> key as if you had pressed the assigned function keys, triggering the relevant CW macros.

You could simply tap the corresponding function keys as normal, but you *may* find it easier to configure the CW Machine so you can hit the <**Insert**>, <**Space bar**> or <**Enter**> instead when you are, for instance, in the thick of a CW pileup or contest.

25.3.2 Mode/band-switching automation

Imagine you're using Quick Switch, operating JTDX. You decide "Enough of this racket already! I'm changing to CW …". You simply click a CW DX spot. As if by magic, JTDX, the UDP BandMap and the JTDX Control Panel are all closed and the CW Machine opens instead. The radio changes to CW with whatever CW-specific settings you use. The log entry pane shows the callsign of the DX spot you clicked. Any previous QSOs with that station are shown in the previous QSOs pane, and the DXCC entity is looked up for the worked/confirmed table … and off you go.

25.4 Keying speed

Just below the menu bar, the CW Machine displays the current keying speed in **W**ords **P**er **M**inute.

- If you are using the **software keyer**, click the text "Keying speed = NN WPM" to *reduce* the speed by 1 WPM per click, or right-click it to *increase* it. Alternatively, with the focus on the CW Machine, the <**Pgup**> and <**PgDn**> keys speed *up* and slow *down*, respectively (similar to N1MM+).
- If you are using a **WinKey** or **WinKeyer2**, and *if* you have ticked analog speed control in the keyer configuration, turn the speed knob on the front of the WinKeyer. It reports its current speed setting to the CW Machine to update the WPM display accordingly. The 'speed pot max and min' sliders on the WinKeyer configuration form determine the speed range of the knob. Otherwise, *if* analog speed control is un-ticked, left- or right-click the CW Machine's speed text in the same way to reduce or increase the WPM.

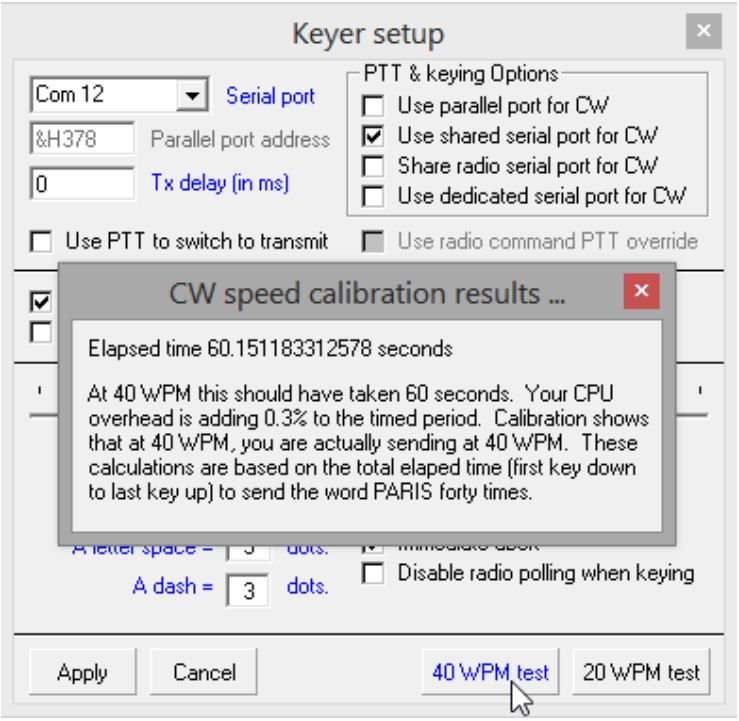
Copyright © 2026 Robert C. Furzer K4CY

25.4.1 Software keyer speed check

How fast are you *really* sending?

Turn your radio off or disable the PTT before you try this!

Click either of the two WPM test buttons at the bottom of the software keyer's **Config ⇨ Keyer setup** form loads the word "PARIS" into the TX buffer either 20 or 40 times, then sends it at 20 or 40 WPM respectively, timing the process to determine the *actual* WPM speed[385] ▶



Hinson tip: there is no such check with hardware keying since the CW timing is left entirely to the box of tricks. Logger32 has no clue how fast or slow it is actually sending, and really doesn't care. It simply tells the keyer "Here you go, magic Morse box, kindly convert this text message to dots and dashes: …" and gets on with other More Important Things.

25.5 CW Machine and CwGet (CW decoder)

Logger32 integrates with the CW decoding program CwGet, allowing received text to be captured from CwGet and placed into the log entry pane, simply by clicking it.

However, CwGet isa separate program and *not* part of the Logger32 distributed package. You'll need to download and install it separately *e.g.* from www.dxsoft.com/en/products/cwget/

By the way, if your Logger32 is configured to "Run as Administrator" (which we don't recommend), CwGet will *also* have to be "Run as Administrator".

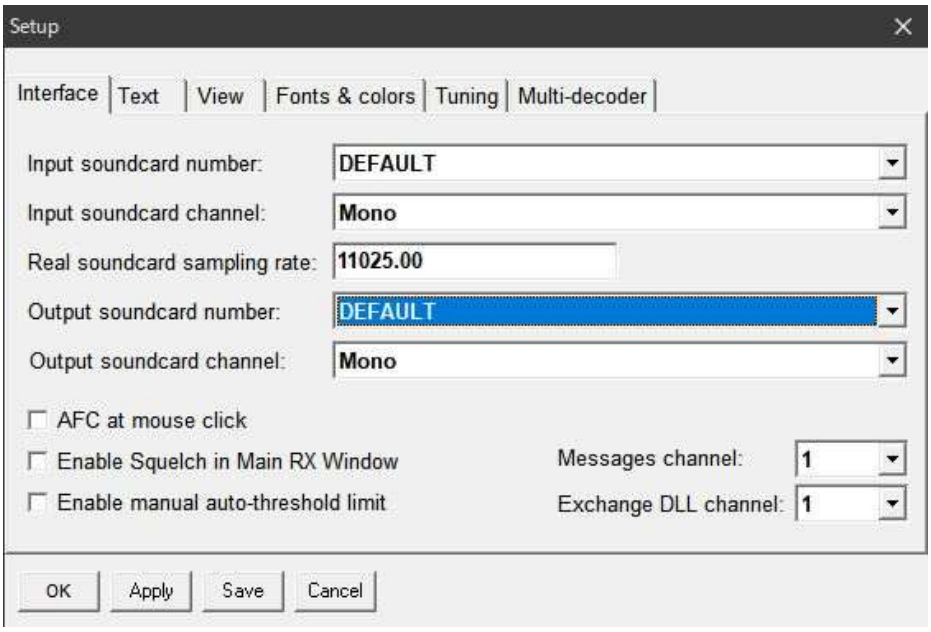
[385] Don't get too hung up on this: the timer depends largely on the accuracy of your PC's clock. It may run fast or slow, or the rate may vary for no apparent reason. Even a recently-calibrated timer in a physics lab would struggle to reach 12 digits of precision, consistently. On top of that, if the PC is doing anything else at the time (which it undoubtedly is), all bets are off.

Copyright © 2026 Robert C. Furzer K4CY

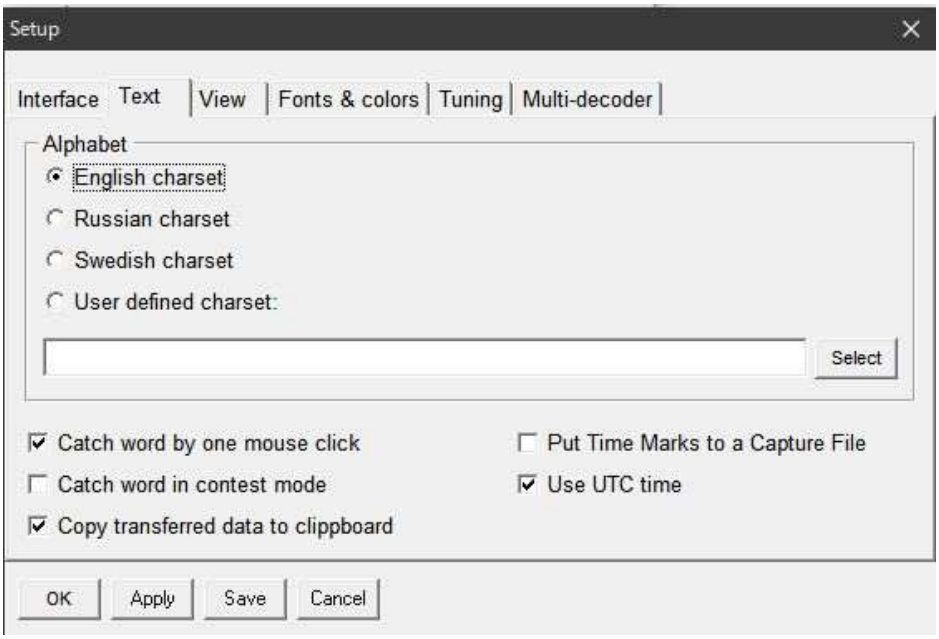
To set up CwGet:

- Click <**Setup**> in CwGet.
- Click <**Apply**> to enact any changes (then try them out if you wish), and <**Save**> them to make them stick.

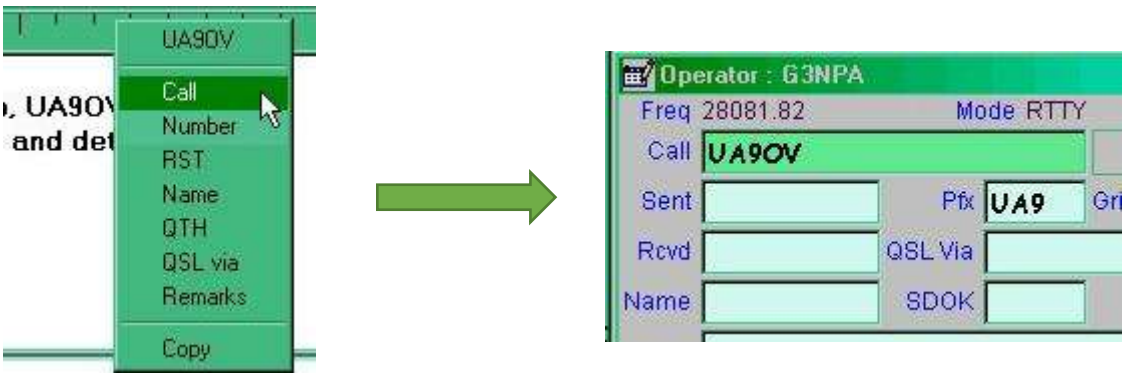
On the <**Interface**> tab select the Input and Output soundcards ▶



On the <**Text**> tab, select <**Catch word by one mouse click**> and <**Copy transferred data to clipboard**> ▶



To use this, click to select an item of text in the CwGet decode window, then click the relevant option to copy the selected text to the log entry pane ▼



Copyright © 2026 Robert C. Furzer K4CY

25.6 CW Machine macros and scripts

25.6.1 Example scripts using macros

The following macro sends a final message with prosigns, returns the radio to receive, logs the QSO and clears the log entry pane ready to log your next QSO:

73 $**Name**$ TNX QSO A^R $**Call**$ DE $**MyCall**$ S^K $**Log**$

This macro sends the DX station's callsign followed by a word space, then 5NN at the keyer speed + 6 WPM. On completion of the transmission, the keyer speed is reset.

$**Call**$ $**Speed+**$$**Speed+**$$**Speed+**$5NN

Since the speed is reset at the end of a message, it is unnecessary to append the corresponding $speed-$ macros *unless* you wish to reset the keyer speed *within* a single transmission *e.g.*

$**Call**$ $**Speed+**$$**Speed+**$$**Speed+**$5NN$**Speed-**$$**Speed-**$$**Speed-**$ QSL?

This will send the DX station's callsign followed by a word space, 5NN sent 6 WPM faster, then a word space and QSL? at normal keying speed.

A typical final over might be scripted:

$**Call**$ de $**MyCall**$ TNX QSO $**Name**$ 73 CUL A^R $**Call**$ de $**MyCall**$ S^K $**Log**$

The final example sends both callsigns, GM/GA/GE/GD (a greeting appropriate for the DX station's local time – as defined in) and includes the DX station op's name (if available in the log entry pane):

$**Call**$ DE $**MyCall**$ $**Greeting**$ $**Name**$ GUD CUAGN – HOWS THINGS?

Hinson tip: if you have multiple callsigns (*e.g.* for regular DXing and contesting, or personal and club calls), use $**MyCall**$ in place of your actual callsign in the macros so they work properly with the current operator as defined through **File ⇨ Change operator**. And remember, the current operator is shown in the caption to the log entry pane. Don't log QSOs with the wrong operator!

25.6.2 The $Loop$ macro

$**Loop**$ replays the macro in which it is placed after a 5 second delay. If a longer delay is required, tag-on additional $**Loop**$ commands at the end. For example, the script:

CQ CQ DE $**MyCall**$ $**MyCall**$ K $**Loop**$$**Loop**$

sends a series of brief CQ messages, each one separated by 10 seconds' listening gap.

After the first message is sent, the CW Machine's text area background color goes red (or rather the color configured in **Config ⇨ Appearance**) - until/unless the looping script is aborted by hitting <**Esc**> or a shortcut, clicking the mouse in the text area or typing any text character, whereupon sending stops instantly and the CW Machine's text area reverts to its original background color.

25.6.3 Chaining CW messages together

If you click a button to send a message and click again (either the same button or a different one) *while the first message is being sent*, the CW Machine automatically inserts a word space between the messages, hence there is no need to include a terminal space at the end of your messages if you plan to chain them together. However, subsequent messages are sent immediately, *without* preceding word spaces, if the second click is even a fraction after the current message ends.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: ending chainable messages with the vertical bar character puts a shortened word space between the repeats as a minimum, and is barely noticeable.

25.6.4 *Software* keyer prosigns[386]

As well as correctly sending various punctuation marks, the CW Machine's software keyer can generate common prosigns as follows:

= sends BT	/ sends DN	(sends KN
+ sends AR	. sends RK	) sends KK
< sends AR	, sends GW	; sends AA
> sends SK	@ sends AC	: sends OS
& sends AS	- sends DU	‘ sends WG
! sends SN	$ sends VU	“ sends RR

25.6.5 *WinKeyer* and *WinKeyer2* prosigns[387]

WinKeyers support a longer list of preconfigured prosigns and, if you need them, additional prosigns can be generated using the *merge character* command (a hex 1B followed by the two characters *e.g.* <1B>AS will send the 'wait' prosign - see the WinKeyer documentation):

= sends BT	. sends RK	; sends KR
+ sends AR	, sends GW	: sends OS
< sends AR	@ sends AC	‘ sends WG
> sends SK	- sends DU	“ sends RR
& is ignored	$ sends VU	* sends MM
! is ignored	(sends KN	# sends UI
/ sends DN	) sends KK	

25.6.6 Regional characters (*software* keyer only)

If the language ID of your PC is Danish, Finnish, Icelandic, Norwegian(Bokmal), Norwegian(Nynorsk), Swedish or Swedish(Finland), the following keyboard characters are supported:

- ASCII 197 (Å) sends AK
- ASCII 196 (Ä) sends AA
- ASCII 214 (Ö) sends OE
- ASCII 198 (Æ) sends AA
- ASCII 216 (Ø) sends OE
- ASCII 222 (Þ) sends a plain L

[386] Although *represented* in this manual as green character-pairs linked with a wavy underline, prosigns are sent as one contiguous Morse sequence *without* the inter-character space.

[387] Notice a few differences between the software keyer and WinKeyer: strings containing these characters will be sent differently depending on which keyer is in use, so are best avoided for compatibility … unless you only ever use one type of keyer anyway.

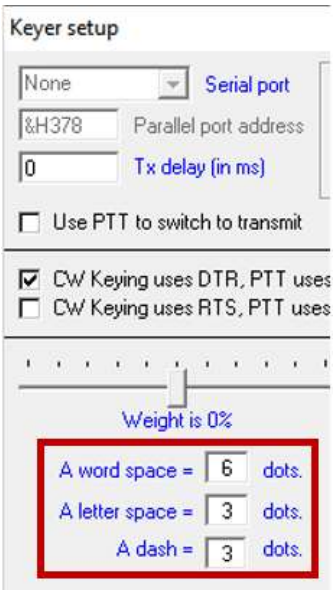
Copyright © 2026 Robert C. Furzer K4CY

25.6.7 Ligatures

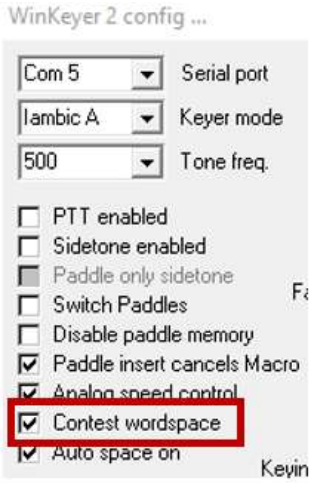
In addition to the prosigns and regional characters that are pre-configured, it is simple to generate your own using the top hat ^ character to link ordinary Morse characters into one contiguous string, transmitted *without* the normal inter-character space. S^K, for instance, would send the end of QSO prosign, not the letter S followed by a letter K.

Hinson tip: personally, I find ligatures easier to comprehend than the obscure prosign symbols. In a macro, A^R seems more intuitive than **+** and S^K is more obvious than **>** Maybe that's just me.

25.6.8 Modified CW spacing



Using software keying, you can specify the lengths of the gaps between words and letters, and the length of a dash, in units of 1 dot-length▶



◂ With a Winkeyer, **Config ⇨ Keyer setup ⇨ Contest wordspace** reduces the normal spacing between words by one dot-length to save precious milliseconds. The effect is quite subtle, barely noticeable in fact, but it does slightly reduce copiability compared to the conventional Morse spacing.

Whereas the above settings affect the timing of every CW message you send, the carat, asterisk and vertical bar characters affect the inter-character spacing only within individual messages sent by the software keyer. With the speed set such that a dot Morse dot (an **E**) takes 200 ms:

- The conventional inter-character space (*e.g.* between **EE**) is 600 ms (3 dots);
- A carat (*e.g.* **E^E**) cuts the inter-character space to just 200 ms (1 dot), so that **E^E** is identical to a normal **I**;
- An asterisk (*e.g.* **E*E**) halves the inter-character space to 300 ms (1½ dots);
- A vertical bar (*e.g.* **E|E**) *extends* the inter-character to 700 ms (3½ dots) - still considerably less than the full 7-dot inter-word space (w h i c h w o u l d m a k e y o u r C W s o u n d c h o p p y). This is a form of Farnsworth spacing described by Don Farnsworth W6TTB in the 1950's, where individual Morse characters are formed and sent at the normal speed but the gaps *between* them are e x p a n d e d, providing additional thinking time for those learning Morse code.

The spacing characters can be concatenated as you wish *e.g.* **E^^E** leaves two dot-lengths gap between the E's, whereas **E^*E** leaves a space of 2½ dot-lengths.

Copyright © 2026 Bob Furzer K4CY

Hinson tip: longer-than-normal gaps can emphasize particular words or distinguish awkward character sequences in your messages, giving listeners' brains just a fraction longer to separate and process each character *e.g.* "CQ CQ VP8EE VP8**|E|E**" or "NAME IS BOB **B|O|B**". Think of it as **bold Morse** if that helps. The effect is more subtle than inserting full spaces between characters.

25.6.9 Other keyboard keys

All log entry pane shortcuts are also active from the CW Machine.

- <**Left arrow**> and <**Right arrow**> move the cursor/edit point within a given field.
- <**Down arrow**> or <**Tab**> jumps to the next field (in a user-definable tab sequence).
- <**Up arrow**> or <**Shift+Tab**> jumps back a field.
- <**PgUp**> and <**PgDn**> speeds up or slows down the CW keying speed, respectively.

25.6.10 CW callsign field

The CW Machine's tool bar field A replicates the Call field in the log entry pane. Type, say, ZL into the CW Machine's callsign field and hit the relevant button to start sending a macro containing "$**Call**$ 5NN" … and *while that message is still being sent*, continue typing the remainder of the callsign 2IFB. Provided you typed ahead of the CW being transmitted, the callsign and message will be sent as one properly-spaced CW string *i.e.* ZL2IFB<space>5NN.

Likewise, a callsign entered into the Call field of the log entry pane in the normal way is transferred to the CW Machine pane … however, because the CW Machine can be CPU-intensive when operating as a software keyer, this *may* not work properly if the full callsign doesn't reach the CW Machine in time for the macro to be interpreted and sent as a contiguous, properly timed string.

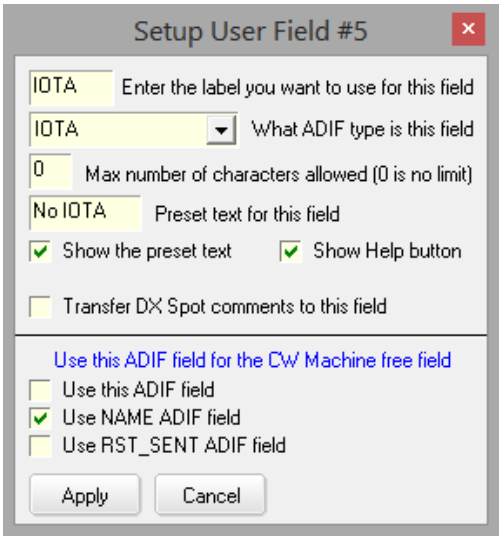
25.6.11 Previous calls

Alongside the callsign field, the "previous calls" pane is automatically populated once a QSO is logged through the CW Machine. Use the small down arrow to see more previously worked callsigns.

'Previous' here refers specifically to stations logged *during the current CW Machine session*, not all time. The CW Machine maintains a temporary working list of stations logged only while it remains open. The list is lost when the CW Machine is closed, hence when you first open the CW Machine, the enigmatic previous calls box is empty apart from its selector arrow, which has nothing to select.

Copyright © 2026 Robert C. Furzer K4CY

25.6.12 Free field



By default, the free field (field E in the CW Machine's toolbar) shows your QSO partner's name. If you worked the station before and logged the person's name, and provided the QSO mask is set to transfer the name, this field will populate automatically.

However, this field can be used for *any* ADIF data.

For example, you might want to use IOTA references logged in the IOTA contest. To do this, configure the log entry pane to include the IOTA reference as one of the user fields, and select <**Use this ADIF field**> in the CW Machine free field section in the lower half of the user field setup window – or use the NAME field as shown ▶

Like the name, the ADIF RST_SENT field can be configured by selecting (ticking) the option. If you want this field to be collected from the CW Machine, tick the box. This option is available in the setup for any user field.

If RST_SENT is selected as the free field, it can be sent by the $**SentRST**$ or $**Name**$ macros.

25.6.13 SRX received serial number

A serial number received during a contest can be entered here.

By popular demand (but in violation of the ADIF specification!), the CW Machine's SRX field accepts non-numeric characters for contest exchanges that include letters. If you *do* enter non-numeric data into this field, however, don't be surprised if when you export the log and import it into some strictly ADIF-compliant system, it objects, rejects it, bursts into tears, throws a tantrum … or quietly ignores and skips these non-compliant ADIF records.

25.7 CW Machine FAQs

Q. Why does the keyer in my SO2R-mini truncate my CW with the CW Machine?

A. There's a lot going on with the CW Machine generating the commands and text, sending them to the keyer, the keyer converting them to Morse keying and passing that to the radio, the radio entering transmit mode and sending the CW. Any processing delays along the way can mess up the CW, especially at the start … so try adding some PTT lead and tail delays - maybe 40ms – to give the system more time to keep up.

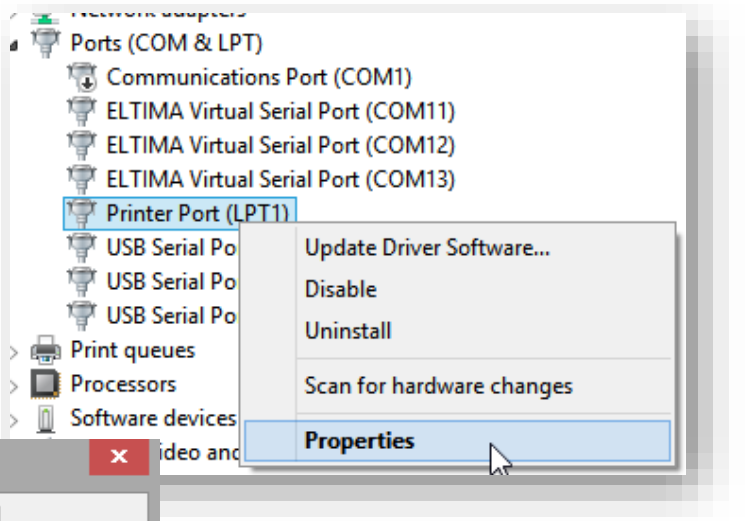
Q. What number COM port or which address LPT port is my keyer interface or WinKeyer using?

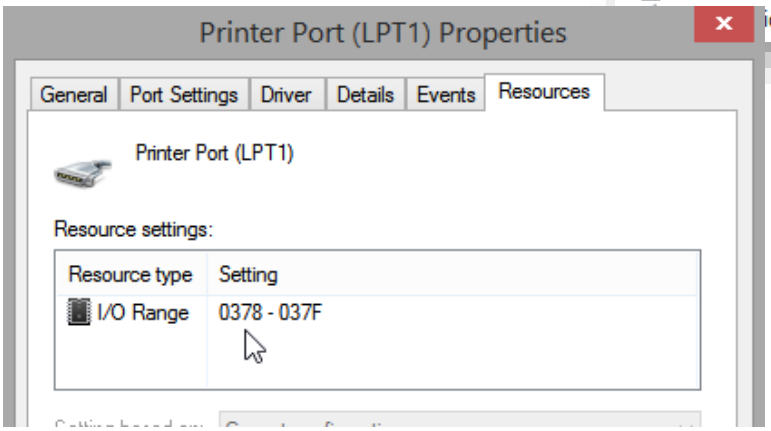
A. PC serial data communications (COM) ports are automatically numbered (*e.g.* COM1) and described generically (*e.g.* "USB Serial Port") by Windows. The standard descriptions/names make it tough to distinguish serial ports if you have several in use, for example for CAT connections to your radios and controls for shack peripherals such as ATUs, rotators, amplifiers *etc*. as well as keyers.

Copyright © 2026 Robert C. Furzer K4CY

To find out the hex address of a parallel LPT port to be used for CW keying:

1. Open the Device manager from the **Win**dows settings using <**Win+i**> or control panel ▶
2. Click the ">" to expand the **Ports (COM & LPT)** section.
3. Right-click the Printer Port (LPT1). Usually there is only one.
4. Click **Properties** to see the settings.





◀ Click to open the **Resources** tab.

The port address you require is the first of two numbers under the I/O Range. Drop the leading zero and enter the remainder in the form &Hxxx.

Hinson tip: the address of LPT1 is *usually* &H378 … so you could simply try that and hope that it works!

Q. My callsign is often busted when I am spotted on DX cluster but – honestly – my Morse is not *that* bad. What's going on?

A. Even if your sending is perfect, it is possible that your transmitted CW signal is malformed due to timing issues such as:

- Slow TX/RX change-over relays – especially old-style open-frame relays – which may truncate the first bit of your transmissions, even slightly truncating all the CW bits in supposedly QSK-capable radios;
- Sequencing delays *deliberately* introduced to avoid hot-switching the PA, linear amplifiers, antenna relays or feed-point preamplifiers;
- Software issues causing the generation of malformed/mis-timed CW characters *e.g.* due to the computer processing higher priority interrupts and other time-critical events, or excessively 'smoothed' waveforms;
- *Deliberate* mis-timing *e.g.* excessive "swing" caused by users of mechanical bugs or hand keys consciously making their CW sound distinctive, but overdoing it;
- RF feedback messing up the CW transmission;
- Hardware issues such as loose antenna connectors, intermittent coax cable or antenna faults, low battery voltage and faulty/failing transmitters introducing hum, chirp or drift;
- Operator error. Typos, spelling and CW errors become more frequent when we are distracted, tired, stressed, inebriated, old or ill, while our capability to notice and correct them simultaneously decreases. We naturally become more irritable and careless.

Copyright © 2026 Robert C. Furzer K4CY

- Your callsign may be confusing people whose Morse is not so good. Calls with lots of dots can be challenging to copy correctly, while any call ending with a K can be problematic if listeners think you are sending the procedural signal K (omitting the preceding word space) meaning “Go ahead”.

On top of all that, errors are possible at the receiving end, while poor propagation, multipath effects (*e.g.* simultaneous short path *and* long path openings, or auroral QSOs) and QRM can make accurate copy challenging, hence *occasional* mistakes are almost inevitable in practice, even if the sending is perfect. Morse code is a raw mode, lacking the data integrity controls such as parity bits and fancy **F**orward **E**rror **C**orrection coding schemes – and the associated overheads – of more modern digimodes.

That said, it is possible to check or analyze your CW transmissions by various methods *e.g.:*

- Simply listen carefully to your own transmitted RF using another receiver, such as a web-based SDR (note: the PC, keyer or rig’s sidetone may not be a reliable guide to your RF transmissions);
- Invite an experienced, trustworthy Morser to critique your sending;
- Monitor the RF waveform using an RF probe and oscilloscope – preferably a storage scope to capture the very start of your transmissions;
- Display the demodulated audio from a receiver on an audio scope or waterfall;
- Send a string of dots at high speed such that fixed delays eat a greater proportion of the CW characters, the truncation becoming ever more noticeable until the dots are no longer recognizable as such;
- Build WA5BDU’s cool Arduino-based CW analyzer.

Depending on the causes, improving your CW may involve:

- Increasing the delay between grounding the PTT line and generating CW;
- Finding and fixing hardware faults;
- Reducing the load on your computer by closing or suspending other programs;
- Changing from software to a hardware-based keyer, such as a WinKeyer;
- Upgrading or replacing your computer and/or bug, radio, amplifier *etc*.;
- Sending more carefully! Take a break. Try concentrating more and practicing *much* more. On CW, quality trumps speed.

With some troublesome callsigns, it may help to:

- Slow down. Give others more thinking time. You can simply reduce your CW speed overall, or just slow down the critical bits of your messages *e.g.* sending your callsign once at a few WPM *below* the selected speed, then again at the selected speed for the rest of your message.
- Use extended spacing to emphasize the separation between the characters of your callsign. “K|4|C|Y” for instance has slightly longer gaps between the characters than “K4CY” but less than “K 4 C Y” using plain spaces.

  Hinson tip: you probably don’t need the emphasis *every single time* you send your callsign – perhaps just once in each CQ message *e.g.* “CQ CQ CQ K4CY K|4|C|Y K4CY A^R K”

- Apply for a different callsign that is easier to copy on CW – one without too many dots and no terminal K.

Copyright © 2026 Robert C. Furzer K4CY

Q. I moved my Logg32 installation to a new computer, including the CW.exe and CW.ini files … but when Logger32 is running, the little Morse key is grayed out and there is no CW Machine window listed in Find lost windows. Where has it gone? How do I get it back?

A. Congrats on the reinstallation! You're nearly done.

From what you've described, it seems the CW Machine is opening and running automatically when Logger32 starts up, but the window is positioned off-screen somewhere so you can't see it.

The **View ⇨ Find lost windows** feature in Logger32 will not find the CW Machine because it is not a Logger32 window: CW.exe is a separate, standalone program.

With Logger32 running:

1. Press <**Ctrl+Alt+Del**>, then click to open the Task Manager.
2. Scroll down the list of running programs (possibly under the <**Apps**> section, more likely under <**Background processes**>) to find the entry for <**CW Machine (32 bit)**>, complete with its blue Morse key.
   - Right-click the CW Machine entry.
   - Click <**End task**> to terminate the invisible CW Machine.
   - Click the corner X to close Task Manager, its job done.

Since Logger32's Morse key icon will have magically changed back to blue, you could manually start the CW Machine … but wait, it will start off-screen somewhere again, and you'll be right back at square one! So keep going:

3. Stop Logger 32.
4. Edit *C:\Logger32\CW.INI* in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad.
5. Look for these two lines giving the window coordinates:
   - Top=*[an integer]*
   - Left=*[another integer]*

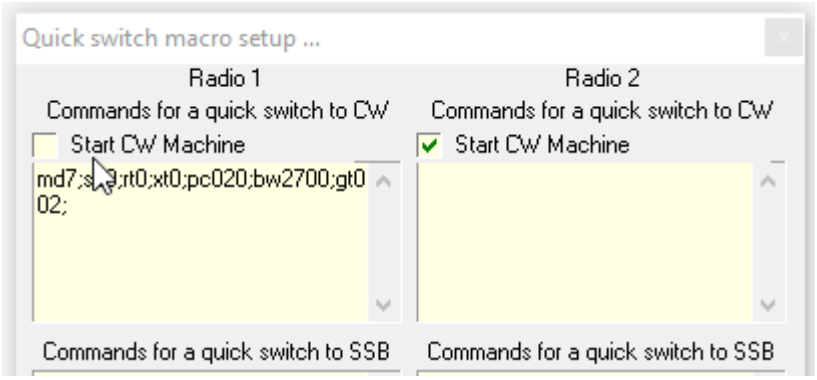
   Change both those numbers to zeroes.
6. Save the edited file and exit the editor.
7. Restart Logger32.
8. Click the Morse key icon to launch the CW Machine. It will start off at the top left corner of your screen.
9. Drag the CW Machine window wherever you want it to be. It will start off there the *next* time you launch the CW Machine.

That's quite a few steps. Alternatively, you could simply delete *C:\Logger32\CW.INI* then close and re-open Logger32. However, by deleting *CW.INI* you will have lost all its settings and macros so you will have to reconfigure the CW Machine from scratch … which probably takes even more steps! You could retrieve the settings from a backup of your original *CW.INI* file, pasting them into the new *CW.INI*, but it's easier to edit just those two window coordinates. If you're short of time, skipping steps 1 and 2 *might* work for you: give it a go.

Copyright © 2026 Robert C. Furzer K4CY

Q. I rarely if ever use the CW Machine. Can I stop it launching automatically when I click a CW spot with Quick Switch enabled?

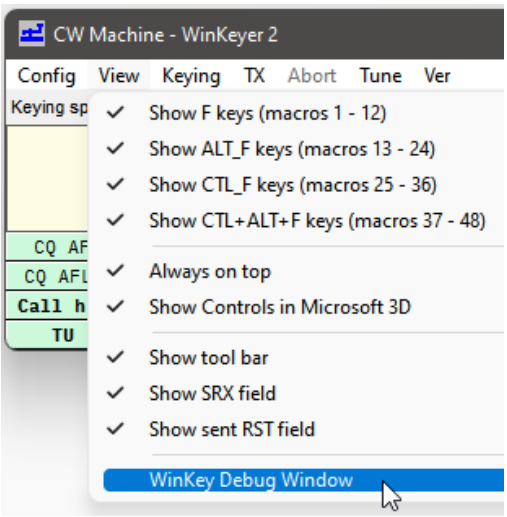
A. Yes, thanks to a pair of options at the top of the Quick Switch configuration pane ▼



Hinson tip: since you can enable/disable the CW Machine separately for each radio in an SO2R setup, you might try sending CW manually using a paddle on radio 1 on one band while using the CW Machine macros and type-ahead buffer simultaneously on radio 2 on another band (unless forbidden by your license or contest rules). Practice hard though before contesting!

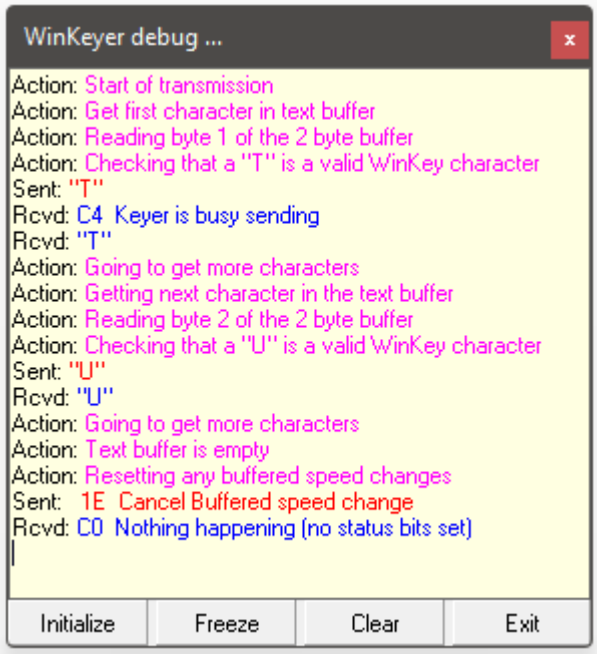
Q. How come the CW Machine appears to be sending characters to my WinKeyer2, but the radio isn't actually transmitting them?

A. WinKeyers and the like are usually black boxes – literally – with no displays and few if any LEDs to indicate visually what they are doing … but they do (usually) offer an *audio* sidetone function. So, having enabled the sidetone using the WinKeyer configuration program, you *should* hear it generating CW when the CW Machine in Logger32 sends characters to the port.



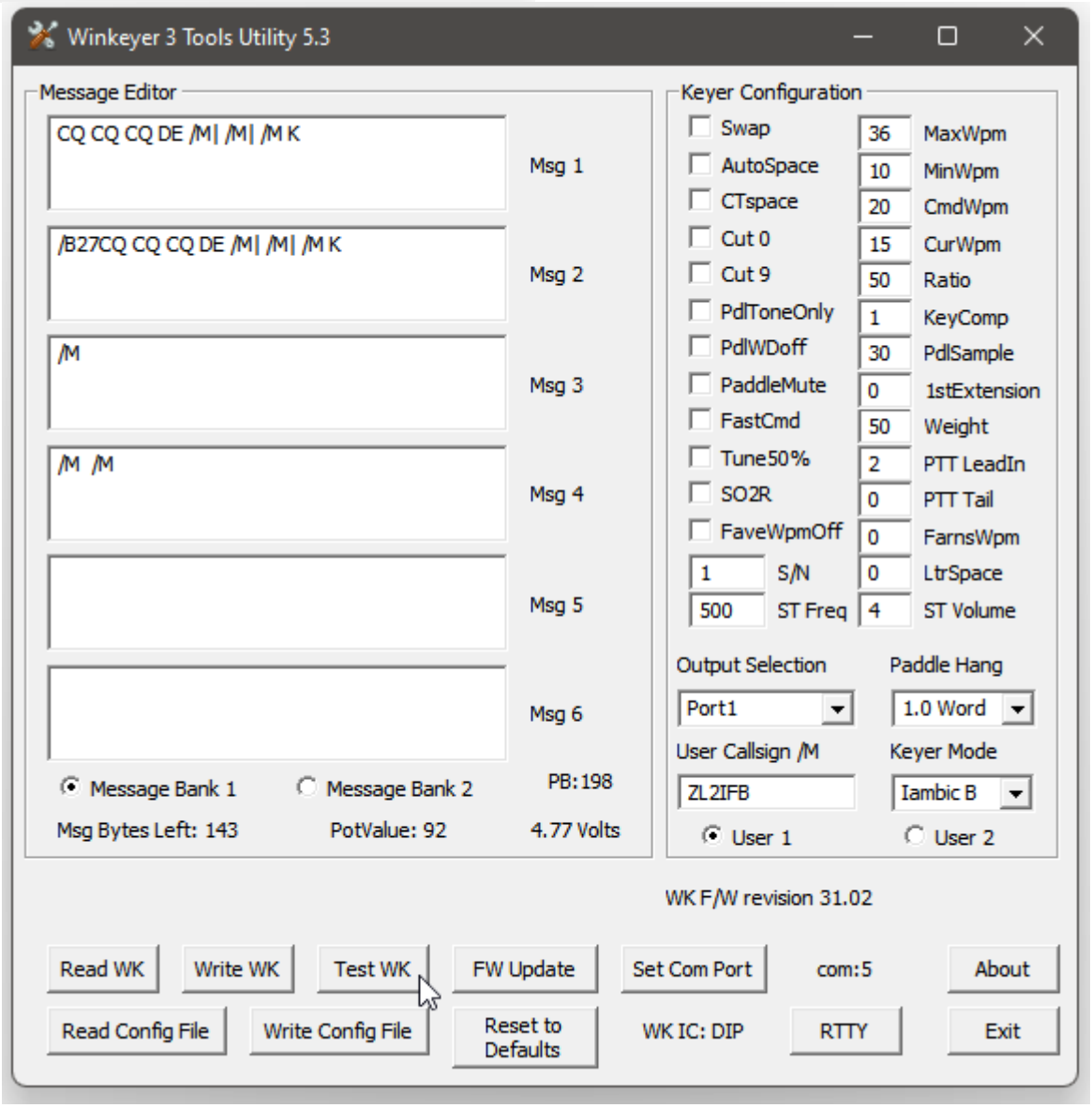
If not, from the CW Machine menu, use **View ⇨ WinKey Debug Window** ▶ to monitor the serial comms flowing between your PC and the WinKeyer for clues about what is going on when you attempt to send messages, change the WinKeyer CW speed *etc*.

Copyright © 2026 Robert C. Furzer K4CY



◀ The debug window shows the CW Machine's activities (in pink), the text or commands sent to the WinKeyer (in red) and its responses flowing back to the CW Machine (in blue).

If you do not see information flowing back and forth through the debug window when you type a message or click a macro button in the CW Machine, check the cable linking the PC to the WinKeyer. Using the same serial port and cable, try to connect, read and test or reconfigure the WinKeyer using the WinKeyer 3 Tools Utility (after having closed Logger32 to release the port) ▼



Copyright © 2026 Robert C. Furzer K4CY

Does the WinKeyer respond? Are you using the correct COM port? If you can't get it working with the WinKeyer Tools Utility, you evidently have a problematic WinKeyer, PC, serial port or cable – and that's not a Logger32 issue.

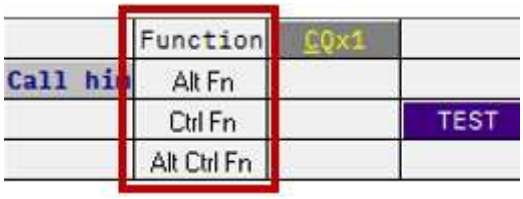
If you hear the WinKeyer or radio sidetone sounding the CW but the radio is still not actually transmitting, the problem apparently lies further downstream, between the WinKeyer and the radio. Check that the radio is in CW mode (!). Watch the CW Machine's "LEDs": the left one should be green indicating that the WinKeyer is correctly initialized. If you have PTT enabled in your CW Machine config, and **Auto TX (VOX)** ticked on the CW Machine <**Keying**> menu, the right LED should go green while sending CW. Put the radio manually into transmit (*e.g*. by pressing the PTT or front panel 'transmit' or MOX button) and send some more CW through the CW Machine: does it transmit OK now? If so, the problem appears to be that the radio is not going into transmit when CW is being generated – a PTT issue. Most radios will go into transmit automatically when keyed using the radio's VOX function, so try that. If VOX isn't working or isn't available, you may need to connect the WinKeyer's PTT output to the radio's PTT line, using the WinKeyer's PTT function. PTT is implemented differently across various WinKeyer devices *e.g*.:

- WinKeyer USB has two pairs of phono sockets for CW keying and PTT outputs to two radios, with a command to select either pair (or to use the PTT to control an amp *etc*.).
- WKMini has one stereo jack using the tip for CW keying and the ring can be either another CW keying output or PTT - a configuration option.
- The G4ZLP CW Keyer has one stereo jack: tip for CW keying and ring for PTT.

Consult the device manual and check the CW Machine and WinKeyer settings, and the cables.

Q. Which function key combinations trigger the macros? I forget.

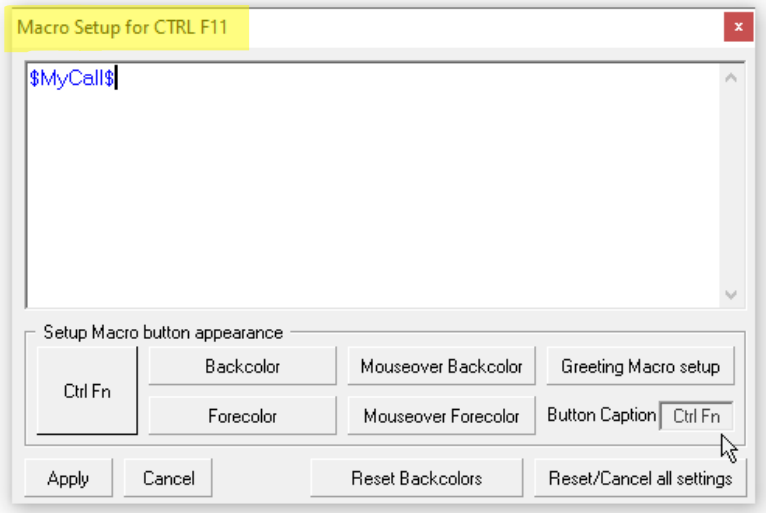
A.



◀ Unless you actually *need* to use all 48 macros, consider using some buttons purely as reminders for each row, labeling a spare column in the macro grid.

Right-click a button then …

… taking your cue from the caption to the configuration form that opens … ▶

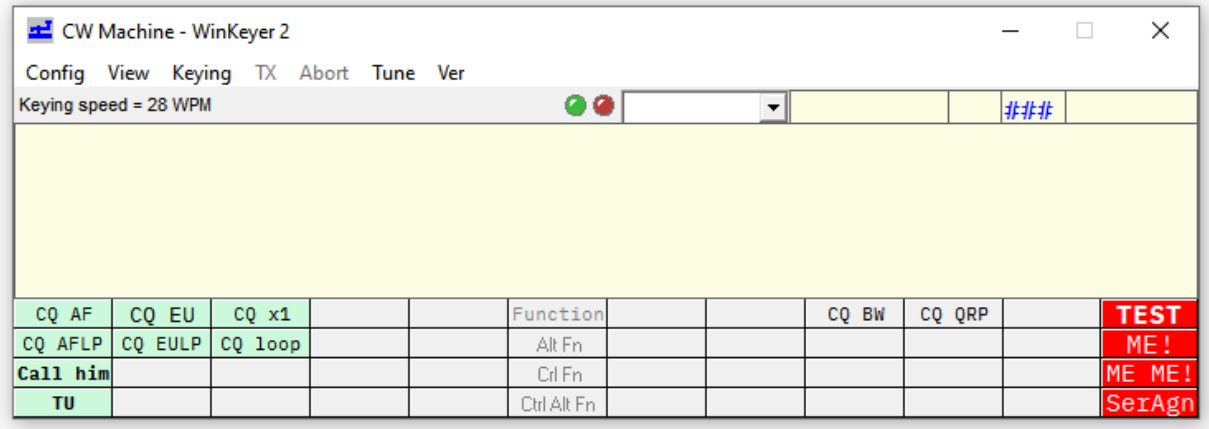


… type your reminder label into the <**Button caption**> field. Optionally pick a subtler shade for the forecolor (text). Leave the macro area empty or type a space (to send nothing if you absentmindedly click the button) or perhaps a short message such as your callsign.

Click <**Apply**> when done. Repeat *ad lib.*

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: since I normally click buttons on the grid rather than using the function keys to trigger CW macros, I prefer to group related macros together on the grid with distinctive button background and text coloring *e.g.* for CQing, casual CW contesting and DX pileupping.



If I normally used the function keys rather than clicking cells, I would probably put the most commonly used macros on the top row, requiring just a press of the function key alone to trigger them. The Alt, Ctrl and Alt+Ctrl combinations would probably not get used as much.

If you find configuring/reconfiguring the macro buttons too tedious using the CW Machine's right-click configurator, it may help to edit *C:\Logger32\CW.INI* directly using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad. Start with a button or two configured as normal to figure out the syntax (oh and take a backup of the original *CW.INI* file before making changes), then copy and edit their definitions as you wish. The macro buttons are numbered in sequence from left to right, one row of 12 at a time. Since *CW.INI* is read when the CW Machine launches and gets over-written if configuration changes are made through the CW Machine, close the CW Machine first before manually editing the file, save it, then open the CW Machine to check out your handiwork (with your TX/PTT disabled or power turned right down unless you are happy for listeners to hear the results of any typos!).

By the way, those red buttons labelled with ME! send my callsign using a macro, rather than the actual letters of my call, since I use the same settings when DXing with my usual callsign or when contesting with my contest call. Since I can't change the button label on the fly, I use "ME!".

Q. If I chain CW messages/macros together, my WinKeyer-emulating device locks up after the first one and has to be power-cycled. Is there anything I can do to avoid this?

A. Try this: add a non-transmitting command or two to the start of each macro *e.g.* $speed+$$speed-$ to speed up and slow back down. Hopefully, the WinKeyer-thing will interpret the command/s instantly, then send the messages at the normal speed without locking up if macros are concatenated.

So, a macro to send Bob's callsign would be like: $speed+$$speed-$K4CY

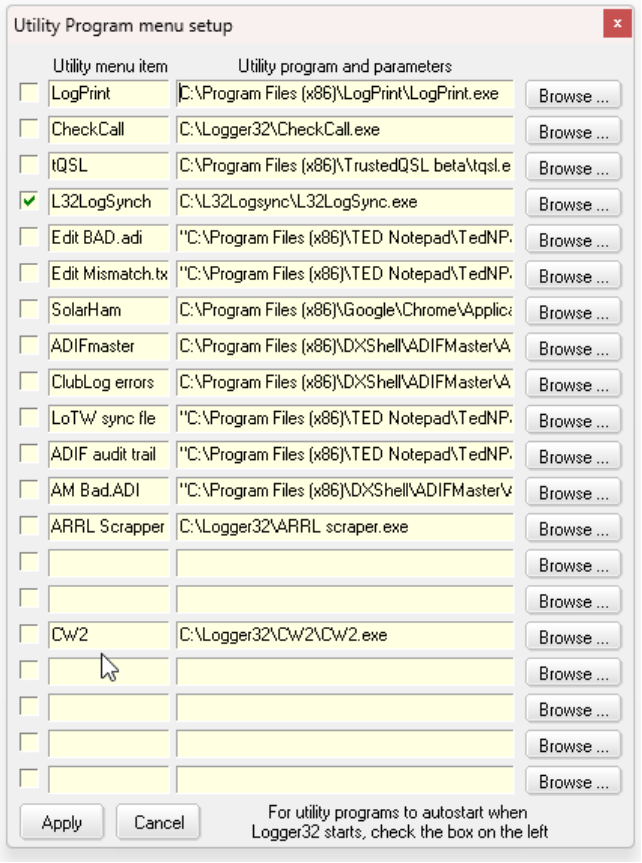
Alternatively, you might contact the supplier of your WinKeyer-thing to determine why the firmware *appears* incapable of correctly handling concatenated messages sent by Logger32's CW Machine in the correct format for true WinKeyers.

Copyright © 2026 Robert C. Furzer K4CY

Q. Can I run the CW Machine twice, once for each of my 2 radios?

A. Probably … but it takes a bit of setting-up, and may not go to plan[388]. This works for Dave N8DC:

- Configure and connect each radio to Logger32 individually, using software, hardware or CAT keying, with the appropriate CW keying interfaces or devices for each.
- Configure the CW Machine to your liking for the first radio, with the appropriate port and the CW timing, memories/macros *etc*. Get it running sweetly … because shortly you'll be copying the configuration for the second radio, except for its port.
- Now, through Windows Explorer:
  - Create a new folder in *C:\Logger32* called *cw2*[389]
  - Copy *C:\Logger32\cw.exe* into the *cw2* folder, then rename it *cw2.exe*
  - Copy *C:\Logger32\cw.ini* into the *cw2* folder. No need to rename it.
- Using **Tools ⇨ Utility program setup**, add a "CW2" entry to your Utilities menu to run the program *C:\Logger32\CW2\CW2.exe* ▶



- Click to run CW2 from the Utilities menu and configure it for the second radio: you will need to change the port and you *may* need to change other settings.
- The little blue Morse key icon #13 ▼ will still open the CW Machine for the first radio as before …



… but now you can open the CW Machine for the second radio from your Utilities menu, instead or possibly even at the same time[390] (for example, using a repeating memory to call CQ on CW on a deserted band in the hope of raising the dead on your VHF radio, *while* busily working loads of CW DX on 20m using your HF radio).

---

[388] **This is an experimental approach, with various drawbacks.** By all means give it a go. If you can't get it to work to your satisfaction, Logger32's usual no-questions-asked full money-back guarantee applies.

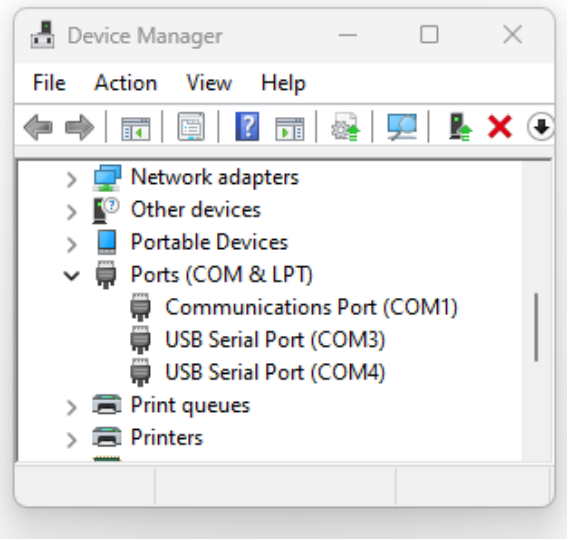
[389] The folder and program names are arbitrary. "CW2" seems intuitive to Dave and me. YMMV.

[390] To run both CW Machines at the same time, I gather you have to open the CW Machine from the Morse key icon *first*, and only *then* open the second instance from the Utilities menu. If you do it the other way around, the universe silently implodes. Puppies get sick. Stock markets crash. That sorta thing.

Copyright © 2026 Robert C. Furzer K4CY

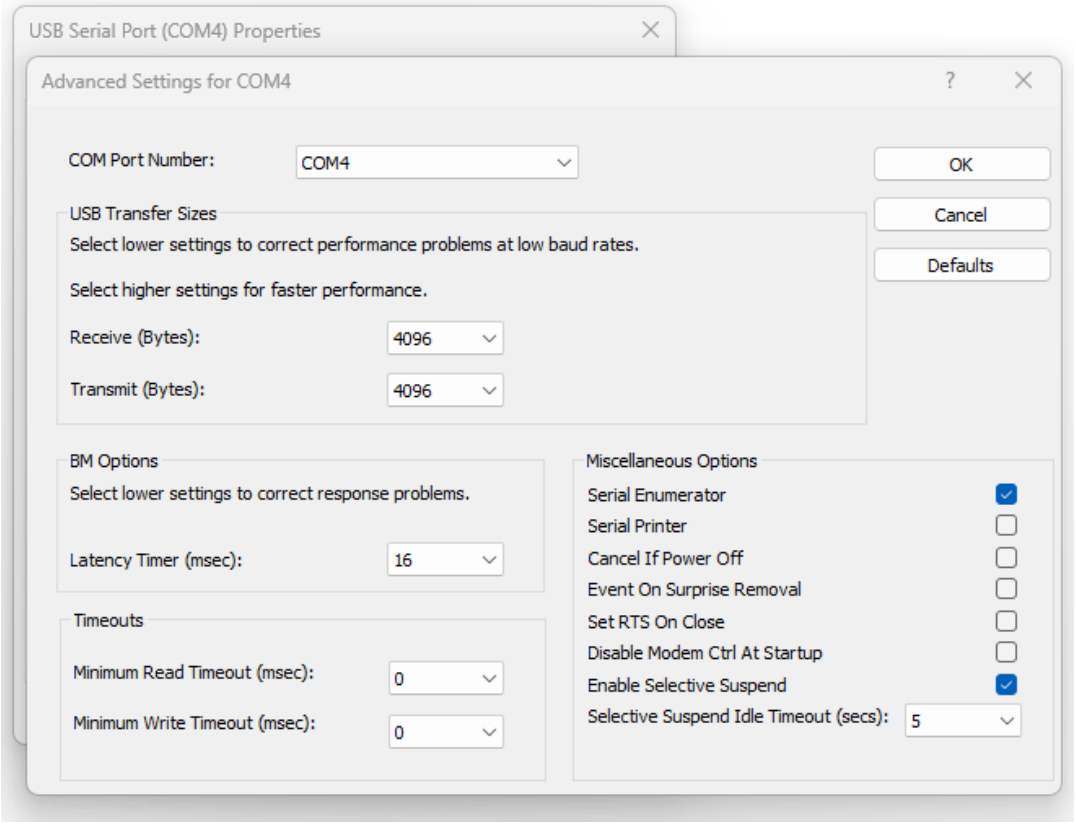
Q. How come the WinKeyer USB speed pot is sometimes unresponsive and it sometimes skips the first character from my CW macros?

A. These symptoms suggest the USB port may be falling asleep when inactive. It takes a moment to wake up when prodded by the CW Machine. The culprit is a Windows port setting called "Selective Suspend", designed to save the planet by automatically powering-down inactive serial ports. Even if you have previously disabled it, it sometimes gets re-enabled by (some) naughty Windows updates.



To disable it (again!):

1. Open **Windows Device Manager**.
2. If not already expanded, click the ">" to expand the **Ports (COM & LPT)** branch ▶
3. Right-click whichever USB serial port you are using for CW[391] and click **Properties**.
4. Click to open the **Port Settings** tab.
5. Click the <**Advanced …**> button ▼



6. Near the lower right corner, click to *un*-tick <**Enable Selective Suspend**>.

---

[391] Or simply work your way through *all* your USB serial ports, disabling Selective Suspend in the same manner. If your CW port is not listed, make sure that <**Show hidden devices**> is ticked on the Device Manager **View** menu … and plug your WinKeyer or CW interface into your PC.

Copyright © 2026 Robert C. Furzer K4CY

7. <**OK**> or your way out of Device Manager.
8. Windows will tell you to reboot: the changed Selective Suspend setting may work without rebooting but to be *sure* it has changed, reboot your PC at some point – probably quite soon given that you are running Windows.

[Thanks for the tip, Darren G0TSM]

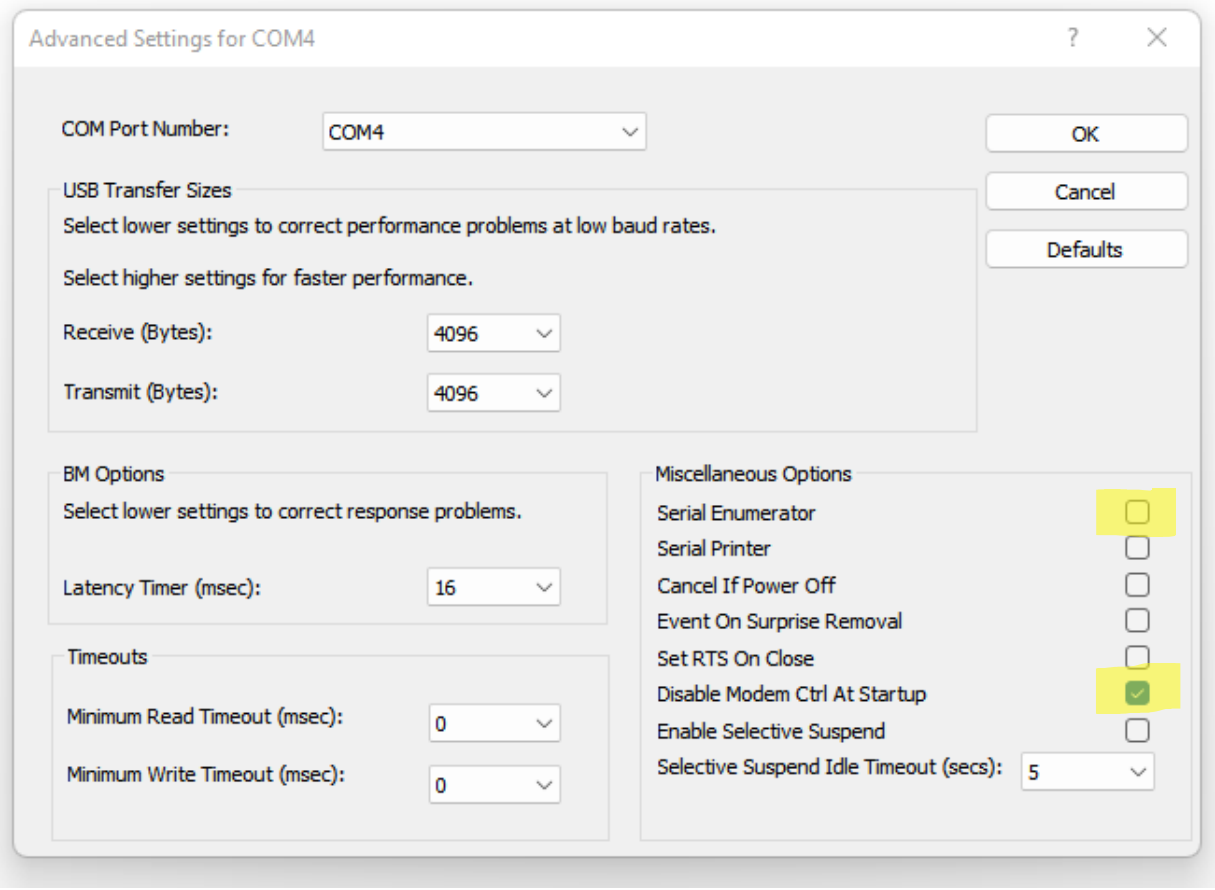
If that *still* doesn't solve your problem, check that your USB root hubs are not also powering-down when inactive.

Q. Why does my radio transmit a series of dashes as the computer boots?

A. This is a consequence of Windows toggling the state of whichever comms port pin you are evidently using to key your radio – part of its boot sequence.

To prevent this annoyance, you can use the "STROBE" line (pin #1 on a parallel port) as a ground return to the CW keying circuit, since this pin is normally held high during the boot. When Logger32 launches, it sets this pin low, hence it provides a CW keying ground return that is ungrounded during the boot.

Under the Advanced settings for a serial port in Windows Device Manager, you can change the defaults *i.e*. *un*tick <**Serial Enumerator**> and tick <**Disable Modem Ctrl At Startup**> ▼



Alternatively you can physically disconnect/disable the keying line or PTT function, set the radio to a non-CW mode, or simply leave the radio turned off until your shack computer has booted.

Copyright © 2026 Robert C. Furzer K4CY

26 Digital Voice Keyer (DVK)

> "Parrots have gone a bit quiet since pirates have gone"
>
> *Karl Pilkington*

Logger32's DVK can transmit pre-recorded audio messages such as a CQ call or your callsign.

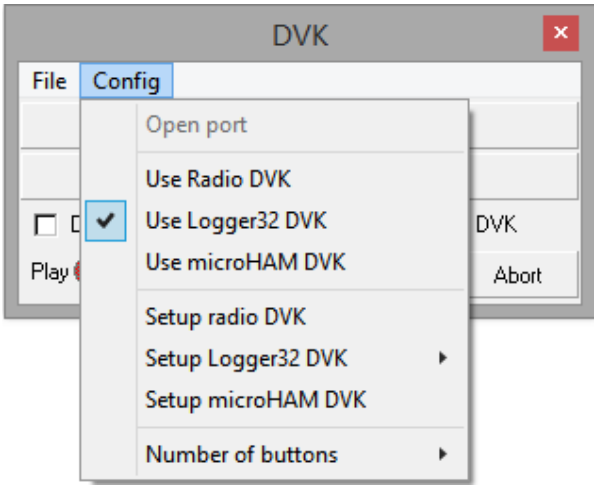
Open the DVK by clicking the toolbar calculator icon #14 ▶



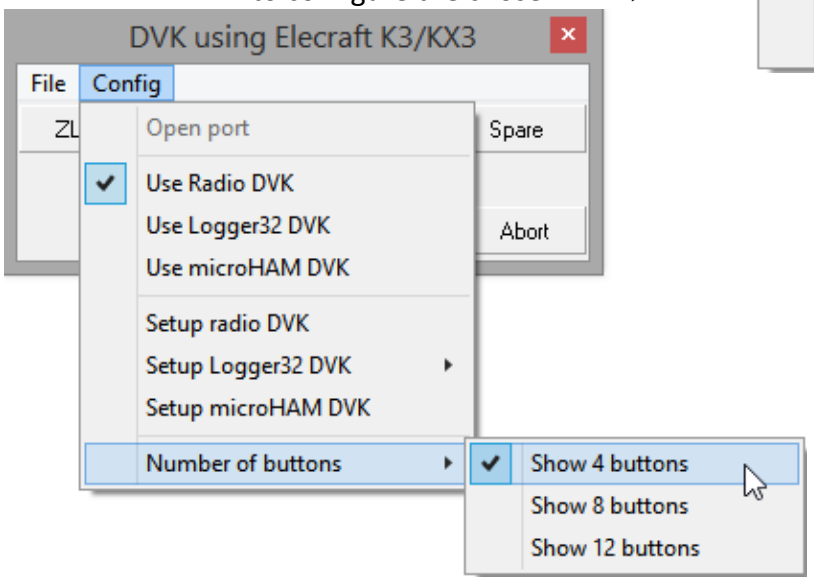
26.1 Setup the DVK

From the DVK <**Config**> menu, first select the type of DVK you wish to use[392]:

1. **Use radio DVK**: trigger a [hardware] DVK built-in to some radios to transmit messages that have been recorded on the radio.
2. **Use Logger32 DVK:** play pre-recorded *.wav* files through the computer sound card.
3. **Use microHAM DVK**: trigger *microHAM* products to send their pre-recorded messages.



Then use the appropriate setup option to configure the chosen DVK ▶



◀ At the bottom of the same config menu, choose 1, 2 or 3 rows of 4 DVK buttons.

[392] For once, it is safe to ignore any port error messages at this point while the port is changed.

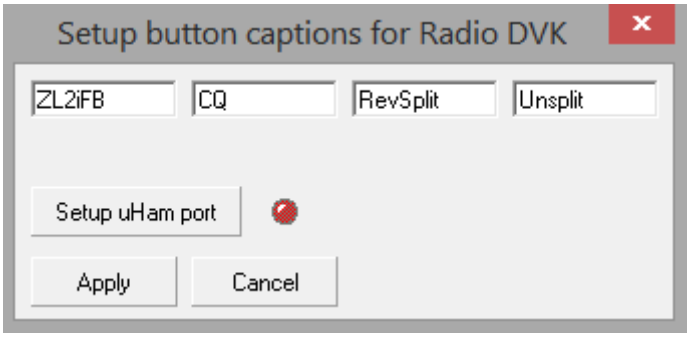
Copyright © 2026 Robert C. Furzer K4CY

26.2 Setup radio DVK

Some radios (such as the Elecraft K3[393], K4 and KX3, and the ICOM IC-7300) have built-in or optional hardware DVKs. Logger32's radio DVK function can simply trigger them to replay stored messages by sending the appropriate CAT command to the radio when you click the DVK button. There is no need to put the radio into transmit first as that happens automatically when the DVK replay is triggered.

First, though, you need to program/record the DVK memories on the radio: see your radio's instruction manual for instructions on how to do that.

Having already set up the type of radio and established CAT control through Logger32, the radio DVK configuration may be as simple as typing labels for the 4, 8 or 12 DVK buttons you have elected to use ▶

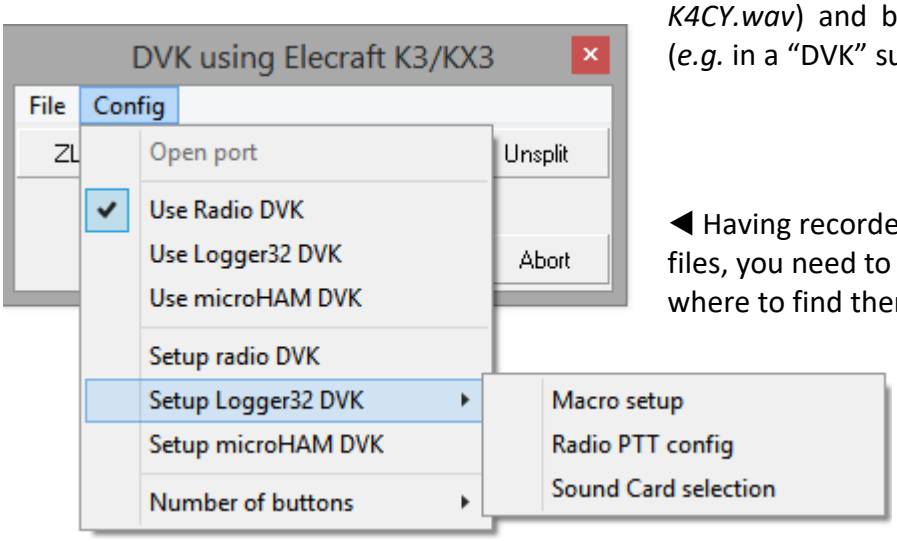


Hinson tip: hardware-based DVKs built-in to some modern radios generally work better than software DVKs running on the shack PC, avoiding timing glitches, stuttuttutteruttering and other annoyances.

26.3 Setup Logger32 DVK

To use the Logger32 [software] DVK, you must first record your voice as *.wav* files with a suitable audio program such as the Microsoft Sound Recorder (provided as a menu option within the DVK) or Audacity or almost any other (most of them handle *.wav* files).

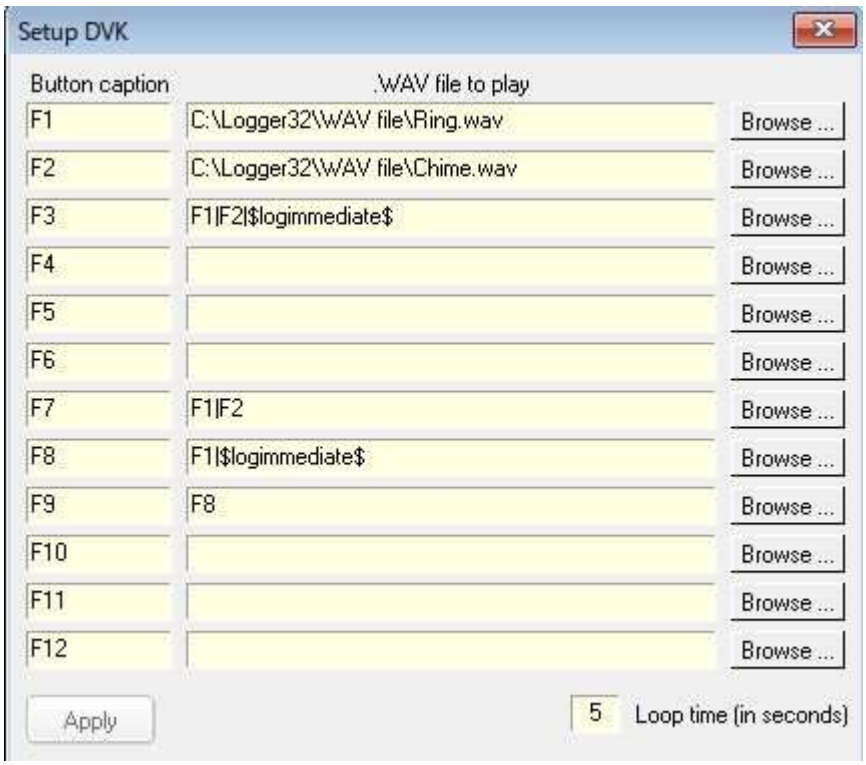
The sound files can have any name (typically relating to the message content *e.g. CQ.wav*, *K4CY.wav*) and be stored wherever you like (*e.g.* in a "DVK" subfolder under *C:\Logger32*).



◀ Having recorded your *.wav* files, you need to tell the DVK where to find them using **Config ⇨ Setup Logger32 DVK ⇨ Macro setup**

[393] On my K3, the DVK can record and replay voice messages for voice modes, data/CW messages for data/CW modes, and radio macros for any mode. I have recorded my callsign into M1 and a CQ message into M2 for both SSB and CW, and macros for split operation on M3 and M4. I can trigger them from the K3 front panel … or using Logger32's radio DVK.

Copyright © 2026 Robert C. Furzer K4CY



◄ Give the button a sensible caption/label and specify which *.wav* file to play (using the <**Browse**> button if you are not sure of the folder or file name).

It is possible to combine *.wav* files, Radio Control Panel macros and even some other function keys (F1, F2 *etc.*), separating each element with a "|" vertical bar. If you are invoking another function key, it can *only* call a *.wav* file: and it *must* be another F key (self-referential or complex definitions are ignored).

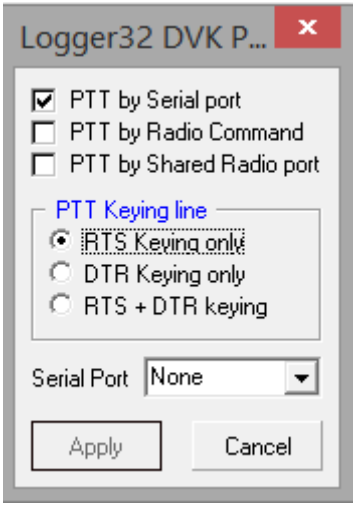
The example above works like this:

- F1 plays the *C:\Logger32\WAV file\Ring.wav* file
- F2 plays *C:\Logger32\WAV file\Chime.wav*
- F3 plays *C:\Logger32\WAV file\Ring.wav*, then *C:\Logger32\WAV file\Chime.wav* and then logs the QSO.
- F7 plays *C:\Logger32\WAV file\Ring.wav*, then *C:\Logger32\WAV file\Chime.wav*
- F8 triggers F1 to play *C:\Logger32\WAV file\Ring.wav* and then logs the QSO.
- F9 does nothing … because the F8 key it invokes would do more than just play a *.wav* file

Loop time defined at the bottom sets the interval between the end of a message and the start of the repeat if you put the <**Loop**> at the end of a macro. There can only be one loop time setting so use multiple <**Loop**>s for longer delays *e.g.* if the loop time is set to 1 second, each <**Loop**> macro adds a second's delay before repeating the message. Six <**Loop**>s give a six second delay. 210 <**Loop**>s give you a surprise 3½ minutes later.

To save your setup, click <**Apply**>.

The Logger32 DVK can put the transmitter intro transmit either using VOX or PTT via the RTS|DTR line on an RS232 port, or a radio command via a CAT connection.



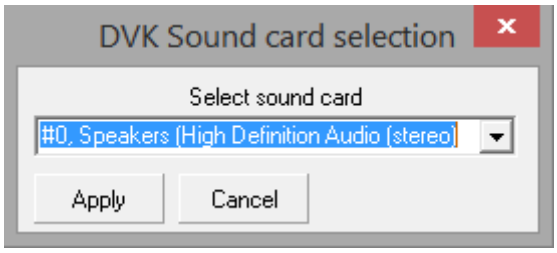
The DVK menu option
Config ⇨ Setup Logger32 DVK
⇨ **Radio PTT** opens the DVK Port setup ▶

Select the PTT method and PTT keying required, including the serial port if applicable, then <**Apply**>.

Copyright © 2026 Robert C. Furzer K4CY

26.3.1 Sound card selection

Finally, specify the sound card[394] on which to play the *.wav* file.



◀ **Config ⇨ Setup Logger32 DVK ⇨ Sound card selection** lists the available sound cards under the drop-down selector button.

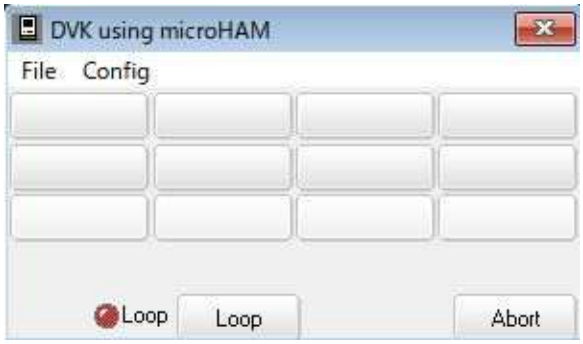
Click to select whichever sound card is connected to your radio, then click <**Apply**>.

Hinson tip: dedicate a sound card *exclusively* for each radio, separate from the sound card used by Windows to place bleeps and other alert/alarm sounds, music, video soundtracks *etc*. Don't broadcast them! Monitor your transmissions to be sure, especially when first setting things up.

26.4 Setup the *micro*HAM DVK

Logger32's DVK function can trigger playback of any of the first eight pre-recorded messages as set up in the *micro*HAM router DVK tab. It is assumed that these messages have already been recorded using the *micro*HAM facilities as the Logger32 DVK function can only play them.

DVK messages are available in the *micro*HAM *micro*KEYER, *micro*KEYER II and MK2R/MK2R+ units only, but *not* the DigiKeyer, CW Keyer or USB Interface II.



Open the DVK menu option **Config ⇨ Setup microHAM DVK** to set up the button definitions, the loop time delay and the control port details ▶

Simply enter the required captions into the appropriate edit boxes, select the Serial Port, enter the loop time (delay) and click <**Apply**>.

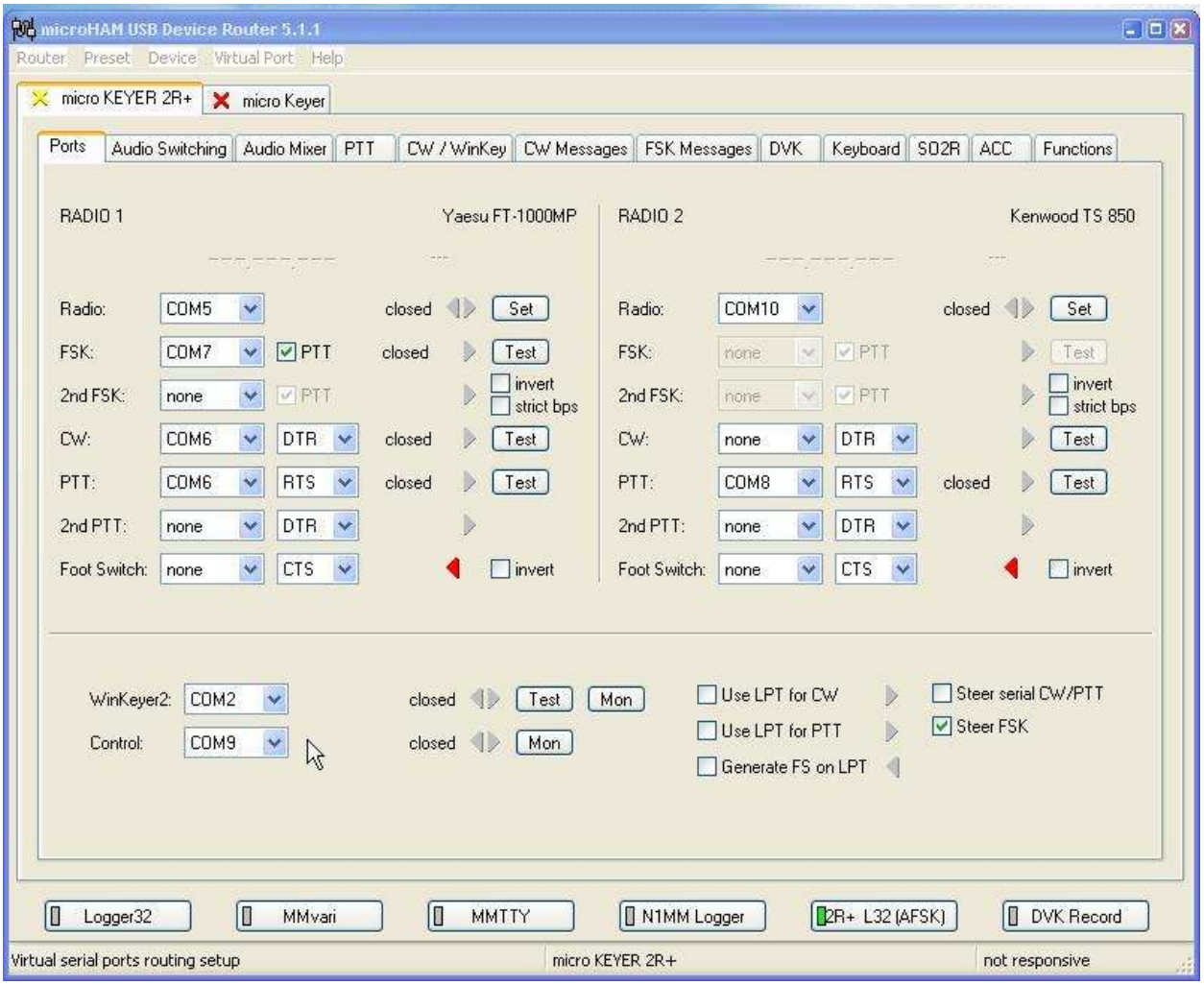
The DVK communicates with the *micro*HAM unit via a virtual control port within the *micro*HAM router software and described below. There is no need to define any more than the port number.

You need a virtual port to be used for control purposes in the device router. Use the *micro*HAM menu option **Virtual Port ⇨ Create**. Once done, add the port number into the Control Port pane on the PORTS tab. Since the virtual port is only going to be used for DVK triggering, check that <**microHAM control protocol on Com port**> in the SO2R tab (MK2R/MK2R+ only) is *not* activated.

[394] A 'sound card' may be literally a card inserted into one of the PC motherboard slots, an audio device built-in to the motherboard itself, or an external audio device/dongle plugged into a USB port.

Copyright © 2026 Robert C. Furzer K4CY

In this example, COM9 is the control port in the *micro*HAM router menu ▼

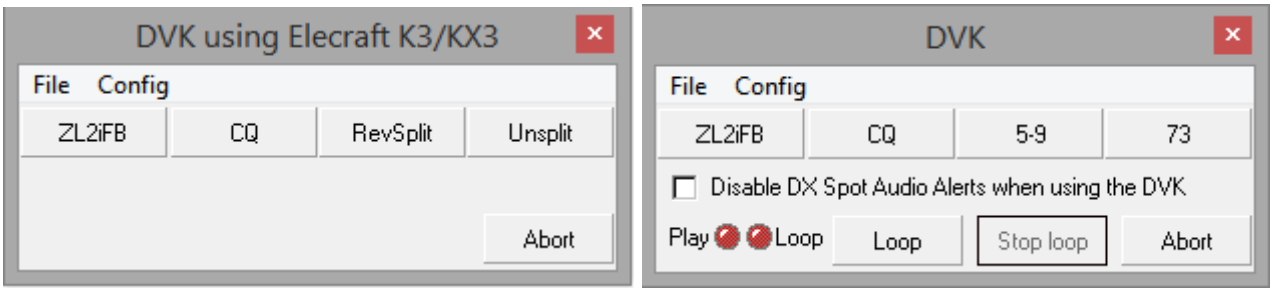


26.5 Using the DVK



◀ Click toolbar icon #14 to open the DVK, then simply click one of the DVK buttons to play the message.

The options ▼ vary according to your configuration.



There are additional options on the Logger32 (software) DVK ▲

With <**Disable DX spot Audio Alerts when using the DVK**>, you can silence DX spot audio alerts while the Logger32 DVK is playing. For instance, you may be calling CQ endlessly on an apparently dead band. Any audio alerts for 'new band country', 'country not confirmed' *etc*. if the band springs

Copyright © 2026 Robert C. Furzer K4CY

to life will not be played (nor will they be transmitted over the radio if you only have the one sound card!) … but interesting DX spots will still be highlighted visually as usual.

The "LEDs" on the Logger32 DVK window indicate the current status.

26.5.1 Repeating and chaining DVK messages

To repeat a message, click <**Loop**> then the message button to be repeated. <**Stop loop**> lets the current message transmission complete but stops it from repeating, while <**Abort**> immediately terminates any message currently playing.

DVK messages can be chained together (queued) *e.g.* if you click button 1, then button 2, then button 3 without waiting, all three DVK messages play in sequence without gaps. Deliberately recording short pauses at the start and/or end of each recorded message stops them runningtogether unnaturally when chained together like this.

26.5.2 DVK hotkeys

The DVK responds to the following hotkeys if the focus is on the DVK window:

- <**F1**> to <**F12**> play the corresponding DVK button.
- <**Ctrl**+**L**> Turns **Loop** on (repeats the message – see above).
- <**Ctrl**+**S**> **S**tops play.
- <**Ctrl**+**A**> or <**Esc**> **A**borts play. <**Esc**> works even if the focus is on the log entry pane …

26.5.3 Triggering DVK messages from the log entry pane

While the focus is on the log entry pane, your F-key (function key) clicks get passed through to:

1. The CW Machine if it is open.
2. Otherwise, the Sound card data window if it is open.
3. Otherwise, the DVK if it is open.
4. Otherwise, nothing. The F-key has nowhere to go and is rudely ignored, a waif and stray.

Imagine, for example, that you have just come across a *massive* pileup for P5DX on 14195 kHz.

Having hastily put your radio into split and found a hole in the pileup, you are busy entering P5DX's details into the log entry pane in the hope of working him. You have neither the CW Machine nor the Sound card data window open at this point … because P5DX is on SSB.

You press (say) <**F1**> … which gets passed through, bypassing the closed CW Machine and Sound card data window, ending up at the DVK. There it triggers the DVK to put the radio into transmit, replay message #1 containing the pre-recorded audio of your spoken callsign to the radio's mike input (or trigger the radio to send your pre-recorded callsign message from its built-in DVK), then release the PTT, returning to receive. You just called him!

Trust me, that one-button-press message transmission really comes into its own after a solid half an hour or more, calling him in vain. Better to wear out your fingerprint than your voice! Meanwhile, you concentrate on listening and tuning around in the pileup in the hope of figuring out where P5DX is listening, beaming the right way, searching the Web for information about the legitimacy of this operation, and excitedly messaging your radio pals to join the fun.

Copyright © 2026 Robert C. Furzer K4CY

Becoming ever more desperate, you decide to call him on CW in the hope of cutting through his SSB pileup, so you click the toolbar Morse key icon #13 to launch the CW Machine. Now, those same <**F1**> keypresses are passed to the CW Machine instead of the DVK, triggering the CW Machine to send its message #1, which you have previously programmed to send your callsign, this time of course in CW.

Moments later, *finally*, you catch his ear, complete the QSO and triumphantly hit <**Enter**> to log the cross-mode P5DX QSO, putting you well on course for #1 Honor Roll. Job done! Time for a little victory dance!

Copyright © 2026 Robert C. Furzer K4CY

27 Configurations

> "To configure is to create,
> to design a masterpiece
> from the raw materials
> of imagination and possibility"
>
> *Leonardo da Vinci*

'Configurations' are discrete setups for Logger32, each with a bunch of settings. Why might you want to do this? …

- Aside from regular operating, you might prefer different settings for, say, digital mode operating, casual contesting, remote operating *etc*. For instance, there's little point showing BandMaps for the WARC bands during contests, so simply turn them off in your CONTEST configuration and arrange the non-WARC BandMaps neatly to keep an eye out for multipliers on other contest bands.
- If the shack and PC is shared between operators (*e.g*. family or club members), they can each set up Logger32 as they wish, with their callsigns, screen and log settings *etc.* If little Jimmy happens to like dark green text on a canary yellow background for his FT# QSOs, that's no problem for mum who prefers more muted pastel shades for her DXing, and pop who needs large fonts with plenty of contrast and despises this new-fangled digital nonsense. Selecting personal configurations is similar to logging in to individual Windows accounts using personal usernames, with different desktops, apps, files, screensavers *etc.*
- Use a DEMO configuration to check out new features, try different screen or logbook layouts, practice and demonstrate various program functions using a DEMO log containing fake QSOs, without messing up your regular Logger32 configuration and risking damage to your real log.

Here are just *some* of the settings typically associated with a given Logger32 configuration:

- The operator's callsign and home QTH.
- The folders and names of various files, not least the logbook.
- The statistics *e.g*. how many countries have been worked, confirmed, submitted and granted for DXCC and other awards.
- The screen layout *i.e*. which panes are visible, whereabouts they are located on the screen and what size they are, plus details within each pane *e.g.* which bands are shown on the worked/confirmed table and the BandMaps, which columns are shown in which order in the logbook and previous QSOs panes, and which apps appear on the <**Utilities**> menu.
- Various colors and fonts.
- Highlighting and alerting if used *e.g*. which sound plays when a DXpedition from a rare DXCC entity appears in your FT# decodes, and what color indicates, say, 'new mode' on DX spots.
- DX cluster settings *e.g*. whether to auto-connect when Logger32 starts up, and if so to which default DX cluster/s.

Copyright © 2026 Robert C. Furzer K4CY

While you are using Logger32, changing various settings to suit your preferences, the *.INI* files for the *current* configuration are updated accordingly. The next time you run Logger32 with a given configuration, it starts up with the same settings as when it was last closed in that configuration.

Hinson tip: if this chapter doesn't quite hit the mark, Lenz DL8RDL has published a YouTube video on Logger32's configurations. Thanks Lenz!

27.1 Defining and using a different configuration

Swapping between configurations is straightforward, although it cannot be done without closing and restarting Logger32. Logger32 reads the configuration and applies most of the settings as it starts up, and when subsidiary functions (such as the CW Machine) are started.

As initially installed, Logger32 uses its default "Logger32" configuration. The settings are saved in *C:\Logger32\Logger32.INI* and other *.INI* files in that folder. However, you can **open a different configuration by adding an argument to the command line that starts Logger32**.

The command line to start Logger32 with a specific (non-default) configuration is:

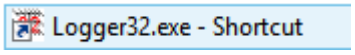
"*C:\Logger32\Logger32.exe*"[space]\[CONFIG]

… where [space] is replaced by a space character and [CONFIG] is replaced by the name of the configuration to load. For example, to run Logger32 with the CONTEST configuration, the command line on my PC is:

"*C:\Logger32\Logger32.exe*" \CONTEST

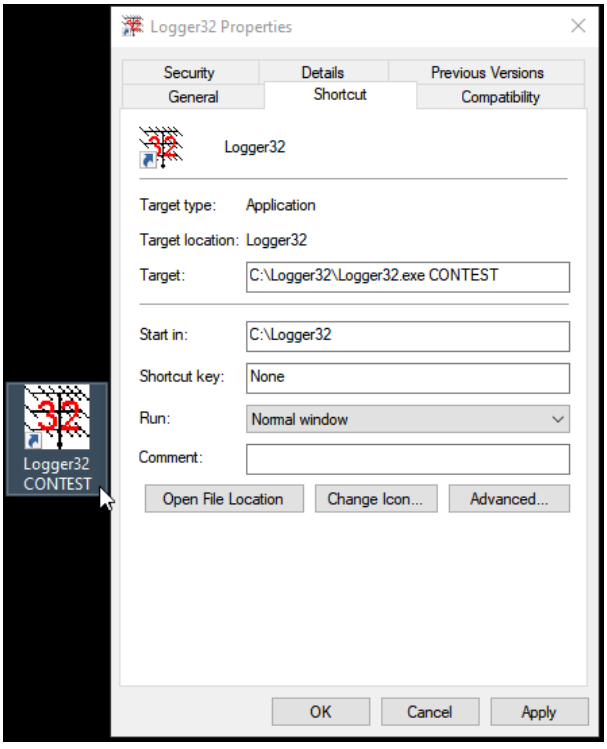
The simplest way is to create an additional Logger32 shortcut/icon on your Windows desktop for each configuration:

1. Press <**Win**+**E**> to open File **E**xplorer.
2. Navigate to your *C:\Logger32* folder or wherever you installed Logger32 on your PC.
3. Right-click the file *Logger32.exe* and click <**Create shortcut**> to create a new shortcut ▼

4. Right-click the new shortcut and select <**Properties**> to open the Properties form.
5. On the <**Shortcut**> tab, in the <**Target**> field, after the path and filename in speech marks, type a space followed by a backslash then the name of your new configuration. The name must be a single word. It does *not* need to be all capitals: we use caps in this manual for clarity. MiXeD cAsE iS oK tOo.

   Here, double-clicking the shortcut opens the \CONTEST configuration ▶



6. Click <**OK**>.

Copyright © 2026 Robert C. Furzer K4CY

7. Single-click the shortcut to select its label (or right-click it and select <**Rename**>), and type a more appropriate name *e.g.* "Logger32 CONTEST", then hit <**Enter**> to save it.
8. Copy, drag or cut-and-paste the shortcut to your desktop.
9. Either double-click the new shortcut to run Logger32 with the new configuration, or double-click the original Logger32 shortcut for the original (default) configuration.

The first time you use a new configuration, Logger32 'clones' its current "Logger32" configuration for the new one. It creates new files in the *C:\Logger32* folder with the configuration as part of the file names *e.g*. *CONTEST.INI* and *CONTESTMyBandMode32.db*. These files are updated as you use the new configuration, changing the settings. If you open the Sound card data window with MMVARI or MMTTY, or the CW Machine window or the Data Terminal under a new configuration, Logger32 creates named *.INI* files for these too, cloning their initial settings from your "Logger32" configuration.

27.2 Swapping configurations using a batch file

A Windows batch (.BAT) file lets you choose between the available Logger32 configurations, and can be programmed to do other things, such as saving backups to a suitable location. Here's an example inspired by Pete N5KD, for you to customize and play with:

```
@ECHO OFF
TITLE Logger32 config selector
:TOP
CLS
ECHO       -------------------------------
ECHO       Select a Logger32 configuration
ECHO       -------------------------------
ECHO.
ECHO       1. Original Logger32 configuration (avoid!)
ECHO       2. Jimmy's personal configuration
ECHO       3. Mum's personal configuration
ECHO       4. Pop's personal configuration
ECHO.
ECHO       0. Quit without launching Logger32
ECHO.
set choice=
set /p choice=.       Enter your choice . . .
if not '%choice%'=='' set choice=%choice:~0,1%
if '%choice%'=='1' goto CONFIG1
if '%choice%'=='2' goto CONFIG2
if '%choice%'=='3' goto CONFIG3
if '%choice%'=='4' goto CONFIG4
if '%choice%'=='0' goto END
ECHO.
```

Copyright © 2026 Robert C. Furzer K4CY

```
ECHO       "%choice%" is invalid. Pick a number from the list, or
zero to quit. Try again.
ECHO.
PAUSE
goto TOP

:CONFIG1
REM Default Logger32 config, used as the basis for any new
configs. Be careful about amending this one.
cd C:\Logger32\
start Logger32.exe
goto END

:CONFIG2
REM Little Jimmys L32 config
cd C:\Logger32
start Logger32.exe \JIMMY
goto END

:CONFIG3
REM Mums special config uses above-normal priority
cd C:\Logger32
start /ABOVENORMAL Logger32.exe \MUM
goto END

:CONFIG4
REM Pops config
CLS
ECHO Use Radio 1 and check IC-7300 is working
ECHO Ensure Slave Port is Open
ECHO Start SDRuno
ECHO On SDRuno - "RX CONTROL" window - SETT. - CAT - Uncheck
"ENABLE & CONNECT" - Check it again
ECHO Press "PLAY" on SDRuno
ECHO.
PAUSE
cd C:\Logger32
start Logger32.exe \POP
goto END

:END
exit
```

Copyright © 2026 Robert C. Furzer K4CY

27.3 Editing Logger32's *.INI* files manually

Rather than storing its settings as obscure hexadecimal parameters in the Windows registry, Logger32 saves its configuration settings in plain ASCII *.INI* or *.ini*[395] (initialization) files in the program folder - normally *C:\Logger32*:

- *Logger32.INI* is the main Logger32 configuration, holding most of the program's default settings.
- *CW.INI* holds the settings and macros for the CW Machine.
- *DataTerminal.ini* holds the settings and macros for the data terminal (TNC).
- *MMVARISoundCardMacros.ini* does what it says on the tin.
- *RadioPanel.ini* holds the settings and macros for the **R**adio **C**ontrol **P**anel.
- *SoundCardMacros.ini* is where the MMVARI macros are stored.
- *UDPPanel.ini* holds the settings for the JTDX Control Panel.
- *UserPara.ini* holds a bunch of RTTY settings.
- *2Tone.ini, FreqPad.ini*, *Mmtty.ini* and *QRZLookup.ini* are examples of configuration files for third-party Logger32 add-on applications providing supplementary functions[396].

Any additional configurations will have the configuration name as part of the *.INI* filenames *e.g*. *CONTEST.INI* and *CONTESTRadioPanel.ini* store the \CONTEST configuration on my PC.

The *.INI* files are human-readable and *can* be edited using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad, providing a direct way to change the settings for a given configuration (*e.g*. to copy the logbook columns or screen layout and assorted color options from one configuration to another) *if* you know what you are doing.

Hinson tip: although it may be tedious, it is safer (less error-prone) to change the settings one-by-one in Logger32 which updates the relevant *.INI* files for you.

Hinson tip: after you have made and proven a bunch of changes, it is worth making a backup of the *.INI* files at that point so you can easily recover if the settings get messed up or lost, somehow.

If you make changes to an *.INI* file, be sure to:

- Shut down Logger32 first to stop it attempting to update its *.INI* files at the same time as you are editing them.
- Decide which *.INI* you intend to edit and make a backup copy of the original, unedited version of the file with a suitable name (*e.g*. *Logger32 original.INI*) just in case you change your mind about the changes, or it all turns to custard.
- Carefully edit the file, following the correct syntax using existing parameters in the *.INI* file as a guide:
    - Lengthier *.INI*s have sections delineated by [Section heading]. Future navigation and editing is a bit easier if you add a single blank spacer line before each section.

[395] *.INI* and *.ini* are functionally equivalent. The case is not significant.

[396] Some third-party apps store their settings elsewhere *e.g*. JTDX uses the *%LocalAppData%\JTDX* folder.

Copyright © 2026 Robert C. Furzer K4CY

- Simple parameters generally use either =true or =false[397], or just = followed by nothing (meaning the parameter is as-yet undefined).
- More complex settings use arrays … and are best left alone! Let Logger32 handle these, unless you simply want to copy arrays between configurations.

- Save the edited version in "text file" format.
- Launch Logger32 with the relevant configuration and check it out before you forget what changes you've made.
  - If it didn't go to plan, revert to the backup (delete the edited file, then make a copy of the backup file with its original name) … and maybe try again later.
  - If it works nicely, now is a good time to make a new backup!

> *What a wonderful job! both, the Logger32 program, as well as its always updated manual. My sincere and eternal thanks to the entire team that maintains Logger32.*
>
> *Tnx Edwin YV5EN*

27.4 Configuration FAQs

Q. Where can I find the CONTEST configuration mentioned in this manual?

A. You won't. You need to create and customize it yourself, following the instructions above.

[397] These are from the English language version of Windows. You may see equivalents in other languages *e.g.* =vrai ou =faux en Français. Users occasionally report configuration issues due to the wrong language being used in the *.INI* files: we're not sure how or why that happens but it is worth checking if you and your PC are multilingual.

Copyright © 2026 Robert C. Furzer K4CY

Q. What about the DEMO log containing fake QSOs? Where is that?

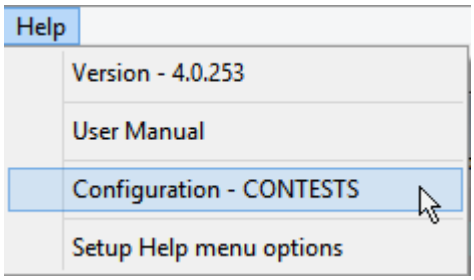
A. You'll need to create one from scratch:

- Create and run a DEMO configuration as described above.
- Use **File ⇨ Change operator** to select a fake callsign (such as TE5TING) so that any fake QSOs you log in this configuration can be readily distinguished from your genuine QSOs logged in your genuine logbook using your genuine operator/station callsign/s.
- Use change logbook to open a new log called DEMO or TE5TING or FAKE or some other distinctive name.
- Have fun inventing and logging a bunch of fake QSOs with various made-up callsigns on various bands and modes, trying out and showing off various program functions, awards *etc.*
- When you've finished playing, demonstrating the system to friends, testing things *etc.,* close Logger32. The DEMO configuration remains available for future fun …
- Feel free to set up a handy desktop "L32 DEMO" shortcut to run Logger32 with the DEMO configuration and log, following the instructions above.
- Revert to your normal configurations with genuine logs to carry on DXing.

Q. Which configuration am I using?

A. If you forget or get confused, <**Help**> on Logger32's main menu tells you which configuration is currently active.

In this case, I was using the "CONTESTS" configuration ▶



Hinson tip: guess what the configuration is called if you start Logger32 without a configuration parameter. Go on, guess. Now check. Nailed it?

Hinson tip: to reduce the risk of messing up your real logbook by logging a bunch of fake QSOs for testing and training purposes, or by accidentally logging beacons as real QSOs, I find it helps to make *obvious* changes to the screen background colors, fonts or layouts for any non-default configurations. For the same reason, you might like to drag the desktop icons for various special Logger32 configurations away from your normal Logger32 (default) icon, and maybe change the icon images to something distinctive using **Properties ⇨ Change icon**

Copyright © 2026 Robert C. Furzer K4CY

Q. I want to set up Logger32 for logging beacons. There are a lot of columns in my logbook I don't need to see and won't use while I'm logging beacons, but when I switch back to regular operating I'll need those columns again. Is there a way to customize which columns are shown?

A. Yes: configurations let you do that. See above!

You can, if you wish, log beacons in your normal logbook as informational entries. Alternatively, you might prefer to set up a fresh logbook *just* to record beacon reception reports – in which case with your BEACON configuration open, use change logbook to create a new logbook solely for logging the beacons you hear. The logbook that is opened when Logger32 starts up is one of many settings saved in the relevant configuration, so you'll get the right one for whichever configuration you are using at the time, as you'll see from the filename shown in the logbook caption.

Q. What *else* can I change using configurations?

A. Almost anything! Open *C:\Logger32\Logger32.INI* in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad to see all your current settings that were saved to disk when you last configured various things. These are recalled for you when you launch Logger32 without a configuration parameter (in other words, using the "default" configuration). **All** those settings can be customized for each separate configuration *if* you really want to do that … but if it is *you* who is using different configurations, you should probably keep at least some settings the same for the sake of your sanity - the audio alerts and highlighting colors, for instance.

> **Hinson tip**: a TRIAL configuration is a handy way to develop and experiment with different screen and logbook layouts, without destroying your original configuration in case the trial doesn't work out. If you *don't* change the operator and logbook in the TRIAL configuration, you will be logging QSOs in your normal logbook under your regular callsign cloned from the regular "Logger32" (default) configuration, so don't go bonkers logging weird stuff.

Copyright © 2026 Robert C. Furzer K4CY

28 Casual contesting and special events

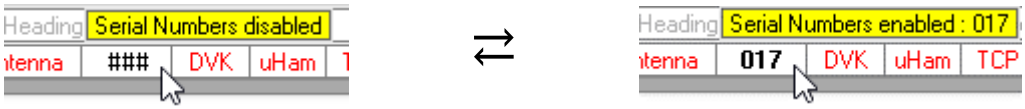
> "Contests are a way
> to measure progress
> and to set new goals"
>
> *Usain Bolt*

Although Logger32 is unsuited for logging a major multi-multi CQ WW DX contest entry from Bhutan, it is perfectly adequate for *casual* small-scale contesting. If you simply intend to have a play, giving away a few points, hunting around for new ones during a contest and maybe clocking up a few dozen QSOs in the process, Logger32 is probably sufficient.

28.1 Contesting

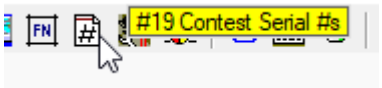
Contest QSO serial numbers may be generated and sent automatically through the Sound card data window, Data Terminal or CW Machine using the $**SerialNum**$ macro. The serial number increments each time a QSO is logged.

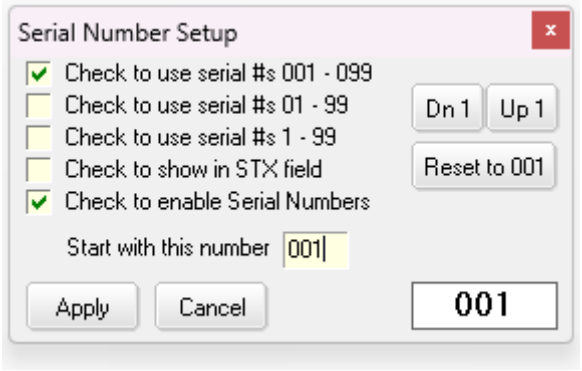
When enabled, the current serial number is shown in the status bar ▼



▲ If serial numbering is not presently active, the panel shows ###

Right-click the panel to enable serial numbers and set the starting value. If you need to change the settings, click <**Serial # setup**> or click the hash toolbar icon #19 ▼ to open the same menu ▶





For low serial numbers, you can choose whether to add zero, one or two leading zeros (the contest rules may specify this).

Hinson tip: contest serial numbering is distinct from routine QSO numbering.

The previous QSOs pane and worked/confirmed table offer basic dupe-checking and 'needed multiplier' capabilities – especially if you open a new log just for the contest which zeroes all your statistics. Conversely, if you aren't specifically chasing contest multipliers, continuing to log QSOs in your normal log means the highlighting and audio alerts for 'new ones' works as it always does: nicely.

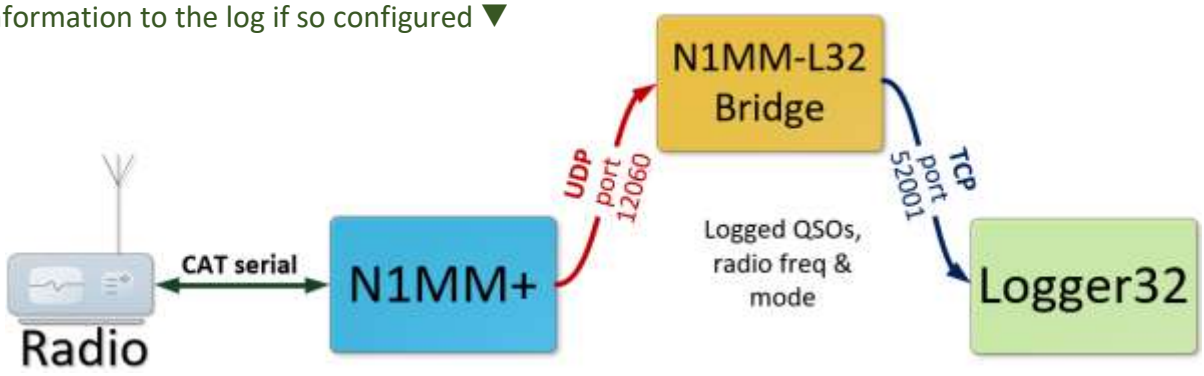
See below if you use a special/short contest callsign.

Copyright © 2026 Robert C. Furzer K4CY

Logger32 cannot score the contest for you and display pretty rate graphs, nor can it generate Cabrillo output if you decide to submit an entry. It can however generate an ADIF from which you can generate Cabrillo using third party utilities, such as real contest loggers (N1MM+ for instance).

If playing in contests becomes a regular and more serious pastime, you might like to generate a CONTEST configuration with all the Logger32 settings you prefer, as opposed to your normal everyday operating. Simply follow the instructions in the configurations chapter.

Hinson tip: better still, use N1MM+ (or another dedicated contest logger) to operate the contest and log and score your QSOs, then either export your log as an ADIF file at the end of the contest and import it to update your consolidated station log in Logger32, or use N1AMG's cool N1MM-to-Logger32 'bridge' utility: it receives logged QSOs and other information (such as the current VFO frequency) broadcast as TCP messages by N1MM+, converting them to UDP messages passed to Logger32 to log as well, using the usual QSO mask and callsign lookups to add supplementary information to the log if so configured ▼



Read all about this in the TCP server chapter.

28.2 Special events and callsigns

Logger32 is perfectly capable of logging QSOs made with special callsigns, both personal calls (*e.g.* ZL25NZ) and club/shared calls (*e.g.* ZL6YOTA). Special, distinctive, interesting callsigns are typically issued to commemorate special events such as exhibitions, ham fairs, for significant anniversaries, or for contesting (*e.g.* ZM4T).

Use **File ⇨ Change operator** to configure the special event callsign being used and to log QSOs appropriately in Logger32.

Hinson tip: remember to change back to your normal, boring callsign after the event. The top line of the log entry pane shows the current operator callsign as a reminder. If you forget, you can correct the operator field recorded against QSO in your log later on, *provided* you know for sure which QSOs to correct.

Use **File ⇨ Change Logbook** to open a separate log for the duration of the special event if you wish, especially for multi-operator events or shared use of special callsigns.

Hinson tip: provided you are the licensee and the *only* person using a special call, you may continue to log QSOs in your regular log *and* submit those QSOs for awards such as DXCC – but check the award rules for details. For DXCC, you will need to apply for a new Callsign Certificate with the special callsign, in order to sign and upload your special-call QSOs to LoTW. If you simply use your regular Callsign Certificate, none of the signed and uploaded special-call QSOs will be matched by the people you contacted hence they will remain unconfirmed in LoTW, essentially lost forever.

Copyright © 2026 Robert C. Furzer K4CY

28.3 Contesting and special event FAQs

Q. Setting up Logger32 for a contest or special event is quite involved. Any tips?

A. Yes! Use separate configurations so that any changes you make to Logger32 during a contest or special event are saved for the next time you use those configurations, while your normal everyday configuration remains exactly as it was.

Q. Logger32 doesn't have all the functions I need for dupe-checking, mult-hunting, scoring, hand-washing and compiling my entry for the COVID Lockdown Contest. The callsign lookups and DX spot processing slow everything down so I can't possibly reach my peak rate of >500 QSOs per hour. What can I do?

A. Either be realistic and make do with what you have … or use a different logger, ideally one that is specifically designed for contesting and supports whichever contests you intend to operate.

Google knows of several - N1MM+, WinTest and others. Explore the Web, download, install, configure and try them out for yourself.

Hinson tip: choose a contest logger that can export your contest logs in ADIF format. That way, you can readily import your single-operator contest logs into your master/consolidated shack log in Logger32, and your club logs into the club's master/consolidated log, making it much easier to manage QSLs, update statistics, search for specific QSOs, track progress towards awards *etc*. accordingly. Contest loggers that *only* export logs in Cabrillo format would require an additional file format conversion step using third party utilities to generate ADIF files that Logger32 can import. Old, unsupported contest loggers may not offer ADIF at all, or may not comply with the current ADIF version, hence you may not be able to import your contest logs into Logger32, at least not without a lot of extra work and support.

Copyright © 2026 Robert C. Furzer K4CY

Q. If I use a 'proper' contesting logger, can I still make use of Logger32's fantastic DXing support functions?

A. Some, yes, particularly if you run Logger32 on a separate screen (possibly even a separate PC with a separate radio – basically a parallel station to avoid any performance hit on your slick contesting setup) *e.g.*:

- Although the contest software may highlight DX spots that qualify as multipliers in the contest, Logger32's BandMaps will highlight any 'new ones' that are spotted regardless of their multiplier status. As such you may want to work them during the contest even if they don't qualify as scoring QSOs, especially if your QSO rate is slow and you aren't obsessed about winning.
- With both contest logger and Logger32 running in parallel, it is easy to log QSOs in the most appropriate log – perhaps both if you are using a contest callsign *i.e*. contact and log DX in the contest logger using the contest callsign for contest points, then contact and log 'new ones' separately using your normal callsign and log for DXCC or other awards[398].
- Showing Logger32's BandMaps for multiple bands on a separate screen may alert you to openings and DX on other bands that you may have missed if the contest software only shows spots for the band you are using at the time.
- While you are patiently waiting to work a multiplier in the contest, entering the DX call into Logger32's log entry pane shows you any previous QSOs and your notes (such as "Blocked short path – try long path") that might help you catch them and bag the points.
- Using a multi-screen or multi-computer setup (*e.g*. running the contest logger on the main shack PC plus Logger32 on a laptop to one side), you can read and respond to DX cluster talk messages, make FT# QSOs, check the countries and run lookups for unfamiliar callsigns, use other handy utilities in Logger32, or simply keep up with emails and browse the web on the second screen during slow periods in the contest, while continuing to work the contest using the contest logger.
- At the end of the contest, or part way through if you like[399], you can save your contest log as an ADIF file and import it into your main station log in Logger32, updating your consolidated log.

> **Hinson tip**: if you import your contest or special event log into your consolidated shack log, QSOs made with your contest, special event and normal callsigns are meshed together. Display the OPERATOR column in your logbook to see which is which. Logger32 lets you select which operators to show in the awards and statistics tables, export specific operators' QSOs *etc.* It is very flexible. However, it can get confusing if you mix up your personal QSOs with QSOs made by you and other people such as shack visitors or friends in a multi-operator contest. I find it better to maintain separate logs for multi-op events.

[398] Please submit a checklog showing any QSOs made and logged under a personal callsign during a contest in which you are using a contest callsign, in order to avoid the DX stations being penalized for uniques.

[399] It may be against the rules of the contest to QSL online *during* the event: check the rules. You may need to suspend any automatic QSO uploading to Club Log, LoTW or QRZ.com until the contest is over.

Copyright © 2026 Robert C. Furzer K4CY

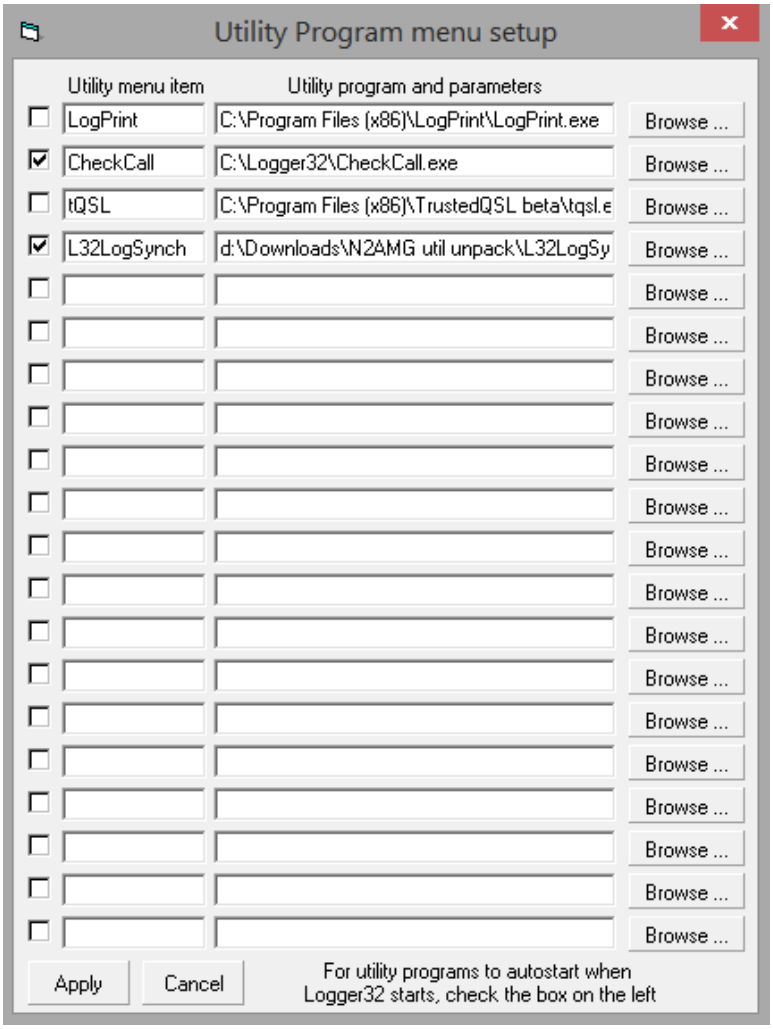
29 Printing

> "Printing is a way to
> bridge the gap between
> the digital and physical worlds"
>
> *John Maeda*

Although Logger32 does not contain any built-in print routines, it can provide the data for other programs such as LogPrint or MS Word, or for online QSL card printing services.

29.1 Using external print programs

Whether you rarely print your log, QSL information *etc*., or do so frequently, it is well worth setting up the print program so that it appears on the <**Utilities**> menu. To do so, click **Tools** ⇨ **Utility Program Setup**, then type in the path and program name (or click the <**Browse**> button to open File **E**xplorer so you can find and select it), add any necessary parameters, and provide a suitable label to be used on the <**Utilities**> menu.



Up to 20 of your favorite printing programs and other utilities can be set up here ▶

Logger32 does not *automatically* generate data for printing. You need to generate the required data in the appropriate format (typically ADIF using the **File** ⇨ **Export logs** or **File** ⇨ **Export files** menus), then save it as a file in a convenient folder, ready to check/edit, format and print.

Copyright © 2026 Robert C. Furzer K4CY

29.2 Printing QSL labels using Microsoft Word

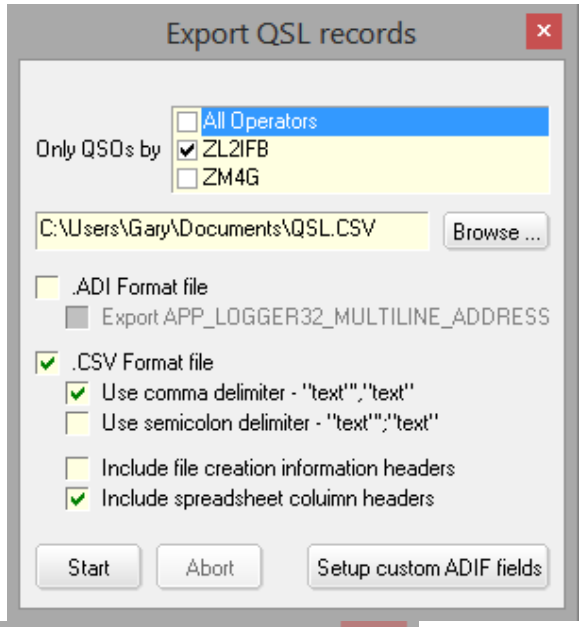
Included with the Logger32 installation package are five Microsoft Word files that will merge QSL data into to pre-formatted labels and are given as examples of what can be achieved with a little effort. An example of a QSL card is also given. The files are:

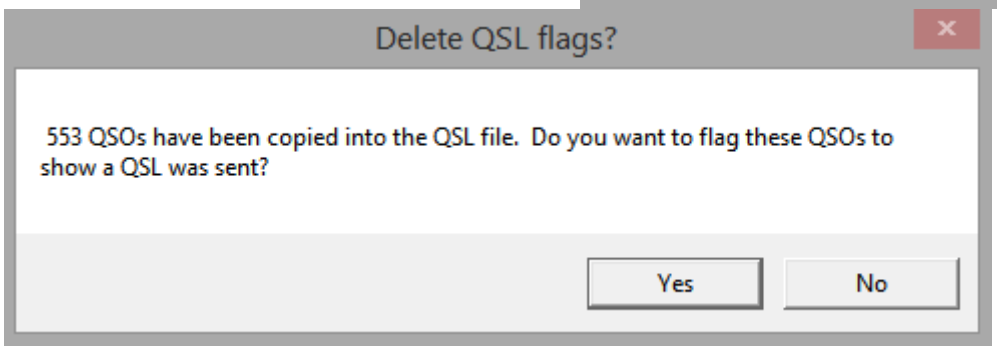
- *QSL_Label 8160 (3by10).doc* for Avery 8160 labels
- *QSL_Label 8163 (2by7).doc* for Avery 8163 labels
- *QSL_Label L7159 (3by8).doc* for Avery L7159 labels
- *QSL_Label L8160 (3by7).doc* for Avery L8160 labels
- *QSL_Label 7421(2by2).doc* for Avery 7421 labels or cards for QSL cards.

29.2.1 How to use the example files

All of the above files expect to find QSL data in a file called *QSL.CSV* in your Documents folder. To generate the required file:

1. Click or **File ⇨ Export files ⇨ Export QSL file** to open the QSL export function.
2. Click <**Browse**> and find your Documents folder.
3. Select <**.CSV Format file**>, <**Use comma delimiter…**> and <**Include spreadsheet column headers**> ▶
4. Click <**Start**> to run the export, generating and saving the data file as specified.
5. Decide whether to update the flags for those QSOs to show <**QSL sent**> rather than <**Send QSL**> ▼





Hinson tip: if, like me, you save up your outbound QSLs until you have enough for a sizeable batch, you may prefer to wait until the entire batch is completed before updating the flags. Click <**No**> now. When the batch is complete, run the QSL export again, but this time click <**Yes**>. That way, if something goes wrong (*e.g.* you run out of labels or time), you can regenerate the QSL information to have another go, updating the flags only when complete: there's no need to hunt through your log for the QSOs flagged "QSL Sent" on the date you ran the function but were unable to complete the process.

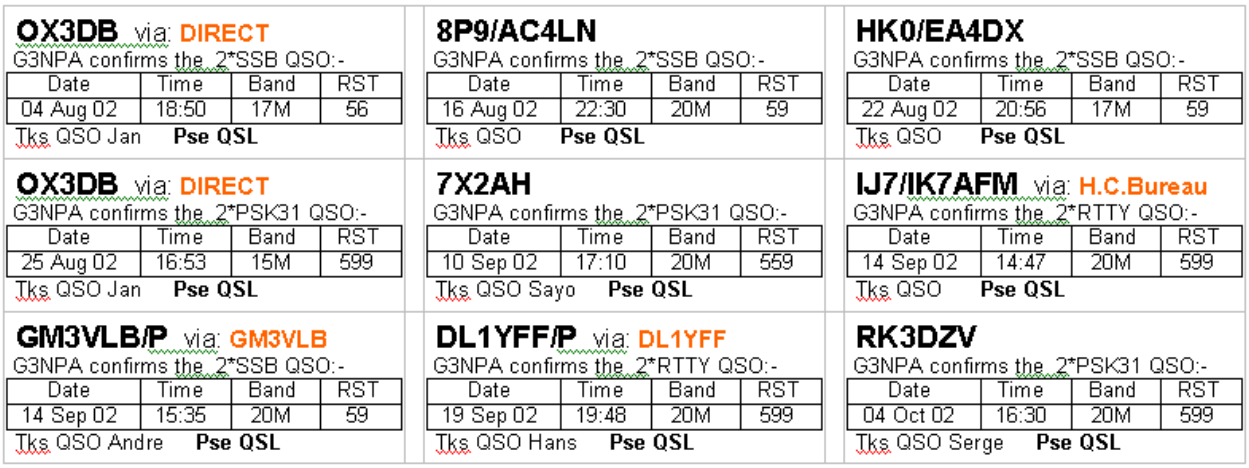
Copyright © 2026 Robert C. Furzer K4CY

29.2.2 Modifying or creating a custom QSL label template

The templates provided with Logger32 have been designed for dates in the format dd mmm yy (*e.g.* 24 Aug 02) and time in the format hh:mm (*e.g.* 18:50). If other formats are used or preferred, the label spacing/layout may need to be altered in Microsoft Word with the mail merge toolbar visible.

On that toolbar you will find a <<>> symbol with REC written underneath. Click the button to reveal the mail merge mask with things like <<Call>> and <<QSL_via>> in the labels. These are the fields from the QSL.CSV data that will be merged and all of these are available from the mail merge toolbar. If you already see this format, then click the same button to see the mail merged results.

One of the example layouts (in part) is shown below.



OX3DB via: DIRECT
G3NPA confirms the 2*SSB QSO:-

Date	Time	Band	RST
04 Aug 02	18:50	17M	56

Tks QSO Jan **Pse QSL**

8P9/AC4LN
G3NPA confirms the 2*SSB QSO:-

Date	Time	Band	RST
16 Aug 02	22:30	20M	59

Tks QSO **Pse QSL**

HK0/EA4DX
G3NPA confirms the 2*SSB QSO:-

Date	Time	Band	RST
22 Aug 02	20:56	17M	59

Tks QSO **Pse QSL**

OX3DB via: DIRECT
G3NPA confirms the 2*PSK31 QSO:-

Date	Time	Band	RST
25 Aug 02	16:53	15M	599

Tks QSO Jan **Pse QSL**

7X2AH
G3NPA confirms the 2*PSK31 QSO:-

Date	Time	Band	RST
10 Sep 02	17:10	20M	559

Tks QSO Sayo **Pse QSL**

IJ7/IK7AFM via: H.C.Bureau
G3NPA confirms the 2*RTTY QSO:-

Date	Time	Band	RST
14 Sep 02	14:47	20M	599

Tks QSO **Pse QSL**

GM3VLB/P via: GM3VLB
G3NPA confirms the 2*SSB QSO:-

Date	Time	Band	RST
14 Sep 02	15:35	20M	59

Tks QSO Andre **Pse QSL**

DL1YFF/P via: DL1YFF
G3NPA confirms the 2*RTTY QSO:-

Date	Time	Band	RST
19 Sep 02	19:48	20M	599

Tks QSO Hans **Pse QSL**

RK3DZV
G3NPA confirms the 2*PSK31 QSO:-

Date	Time	Band	RST
04 Oct 02	16:30	20M	599

Tks QSO Serge **Pse QSL**

Bob, sincere thanks for your prompt reply and especially for the fantastic logging program and support.

Jim EI4HH

29.3 Printing award tables

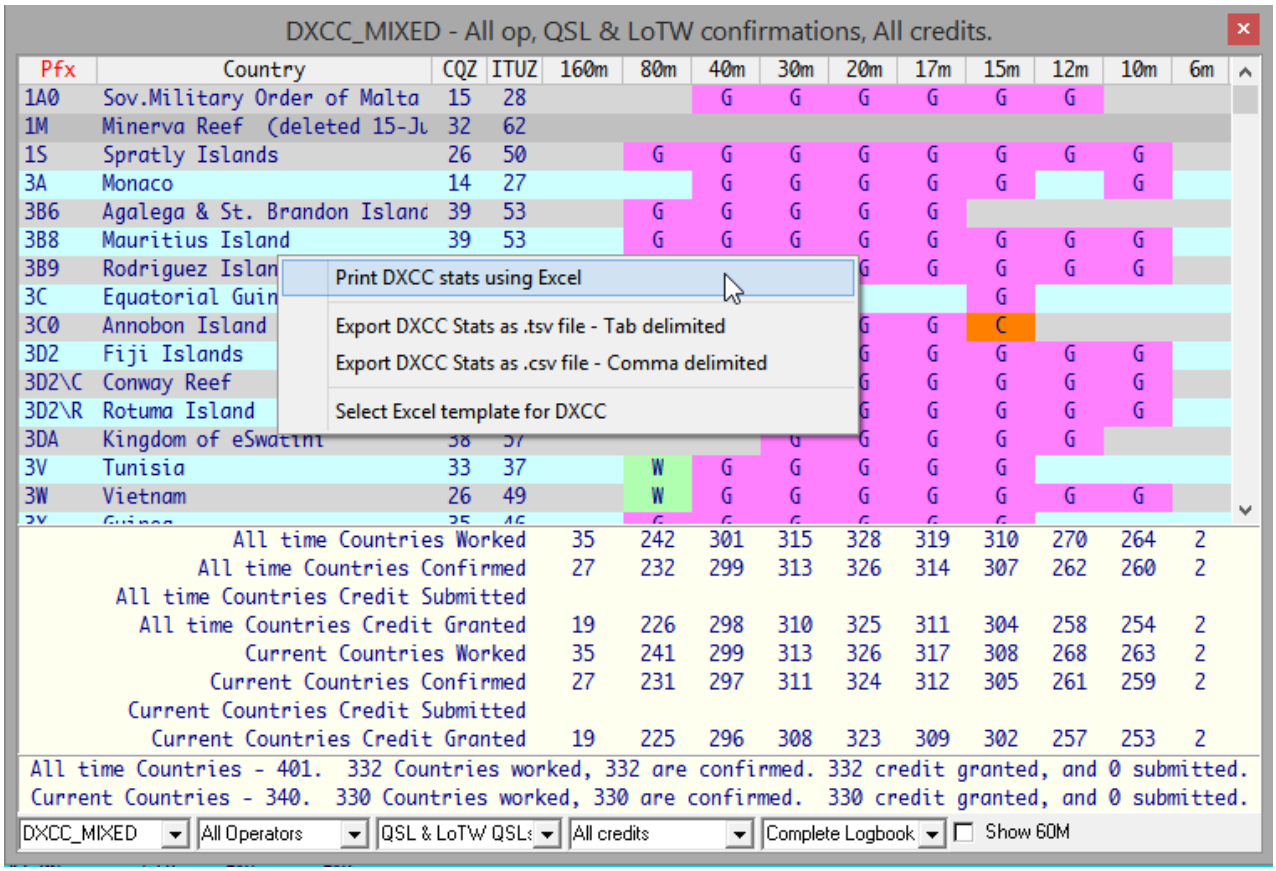
You can print a hardcopy of the awards tables from Logger32 using Microsoft Excel or some other program that accepts tab- or comma-separated data files.

29.3.1 Process

1. Open the awards table you wish to print from the <**Awards**> menu.
2. Select the specific award, operator/s, QSL types and date range at the bottom of the table, including the 60m band if you wish.

Copyright © 2026 Robert C. Furzer K4CY

3. Right-click anywhere among the colored blobs to open a menu ▼



29.3.2 Print using Excel

1. Click <**Print [**award**] stats using Excel**> to generate a print preview. Look it over to confirm that it shows what you expected (*e.g.* data for the correct award!). If not, return to Logger32 by clicking <**Close**> in the preview screen or pressing <**Esc**>.

> **Hinson tip**: if the <**Print [**award**] stats using Excel**> menu line is grayed out and unavailable, remove the line **Check For Excel=No** from your *C:\Logger32\Logger32.INI* file.

2. To print it, click the print button in the preview screen.
3. It is a good idea to save the Excel spreadsheet using **File ⇨ Save as** in case you want to review, edit or print it later.
4. Close the spreadsheet using the X in the top right-hand corner.

> **Hinson tip**: by default, Logger32 uses the *donotdelete.xls* spreadsheet template supplied with Logger32's installation files. You can select different customized Excel templates for the WAC, CQ Zones, DXCC, DXCC Challenge and IOTA awards, using <**Select Excel template for [**award**]**>. Logger32 remembers your choice here by saving the info to *C:\Logger32\Logger32.ini* as usual.

29.3.3 Generating a CSV for another program

Click either the tab-delimited or comma-delimited option to produce a *.txt* file in your Logger32 folder. The file name for each award is *[award]Stats.txt e.g. IOTAStats.txt*.

Copyright © 2026 Robert C. Furzer K4CY

If you choose to use the text file format for DXCC information, you will find additional columns to the right of the main data.

29.3.4 DXCC-4L32f.xls and IOTA-4L32f.xls

Two customized variants of the default *doNotDelete.xls* template were written by Javier EA1AUS (SK) and kind shared via the Files area of the Hamlogger group at Groups.IO. These two Excel files are designed to review or print DXCC and/or IOTA awards (specifically) from Logger32, with the ability to filter the data by specified criteria using the buttons in the top right of the form:

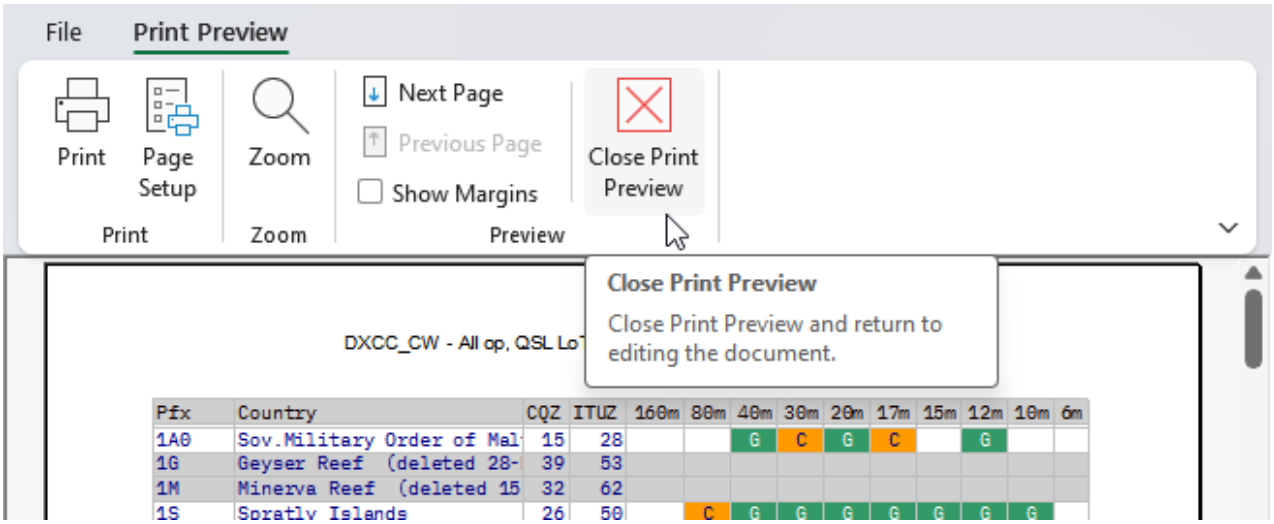
- **W** = Worked
- **nW** = Not worked
- **W/N** = Worked but not yet confirmed, or never worked
- **C** = Confirmed
- **G** = Granted (for the DXCC or IOTA awards)
- **nG** = Not granted
- **S** = Submitted
- **D** = Deleted
- **Blank button**: reset (remove all the filters)

For the IOTA award, there are additional filters for the award's continental areas.

To use these files, download them from the Hamlogger group's Files area, then unpack the zip into *C:\Logger32*.

To setup as default template for your award, open <**Awards**> from the main menu and select DXCC or IOTA. Right-click anywhere around the colored blobs then click <**Select Excel template for [**award**]**> at the bottom of the menu. Find, select and <**Open**> the relevant template.

Now click <**Print [**award**] Stats using Excel**> from the same right-click menu. Logger32 opens the template, populates it with the relevant award's statistics data, then selects, sorts and colors the information, and finally opens Excel's print preview. If you don't simply want to print it out as-is, click the <**Close Print Preview**> button ▼



Now you can adjust the filtering, sorting, color-coding *etc.*, save the result, print it or whatever in Excel.

Copyright © 2026 Robert C. Furzer K4CY

29.4 Printing FAQs

Q. Can I print QSL labels or cards from my log?

A. No, at least not directly. Logger32 does not have any print routines or functions built-in, but it can generate the data to be printed by other programs or services. You could use LogPrint or the MS Word mail merge templates supplied with Logger32, or maybe send your QSL records to a reputable online card-printing service.

Have you considered electronic QSLs such as LoTW, eQSL or Club Log? They are greener, cheaper and quicker than paper cards.

Q. I don't have MS Excel. Can I still use the Excel functions and templates?

A. Probably, provided you have a spreadsheet-type app that understands the simple *.CSV* (**C**omma **S**eparated **V**ariable) data format. I am not aware of such apps any that don't … but then I haven't exactly been hunting for them.

Q. Why aren't DX spots being processed and added to my band maps while I am working on my awards?

A. Logger32 temporarily puts the DX spots processing on hold to concentrate your PC's feeble mind on the awards stuff. When you close the award report screen, the spots will flood in much like a tsunami at a coastal JA nuclear power station – only without the sea water. Or the radiation.

Copyright © 2026 Robert C. Furzer K4CY

30 Multi-language support

> "My hovercraft is full of eels"
>
> *Monty Python*

Logger32 supports some non-English languages for its menus, error and warning messages, and some data-entry field names ▼



Internally, Logger32 program supports Arabic, Baltic, Eastern and Western European, Greek, Hebrew, Italian, Japanese, Korean, Russian, simplified and traditional Chinese and Turkish character sets. This does *not* mean you can enter data into Logger32 in these character sets however.

30.1 Using 'foreign' languages in Logger32

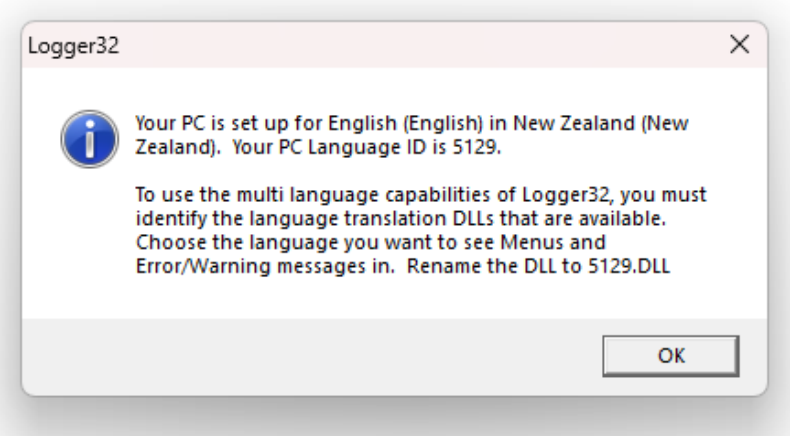
Menu and message multi-language support has been implemented in Logger32 using 'satellite' **D**ynamic **L**ink **L**ibraries – compiled DLLs containing localized resources, specifically the text for items on menus and messages. There is only one version of *Logger32.exe* itself: everyone uses the same executable program … that calls the relevant satellite DLL for the US English original or 'foreign' translated text.

By default, Logger32 calls the English (US) DLL in the Logger32 folder (normally *C:\Logger32*). However, we can replace the English (US) DLL with any of those available for other languages, using the following procedure.

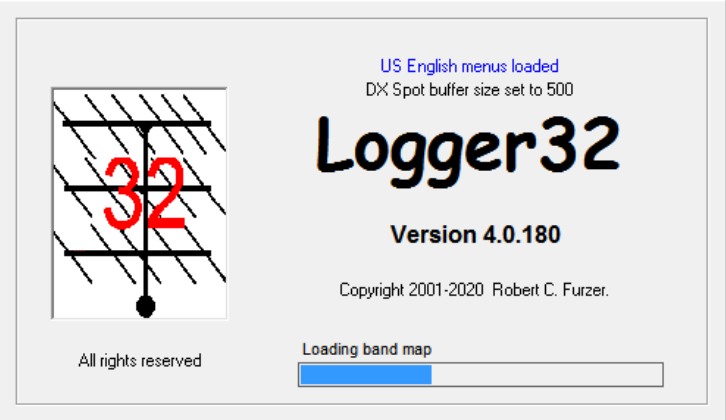
Copyright © 2026 Robert C. Furzer K4CY

1. With your computer set to your normal language and region settings, run Logger32 and click **Setup** ⇨ **Language** to display your "PC Language ID" ▶

   Make a note of the ID number, then click <**OK**> and close Logger32.



2. Unzip the distribution file called *DLL.ZIP* and look for your own language DLL file. Extract this file to your *C:\Logger32* folder.
3. Rename your language DLL to *nnnn.dll* where *nnnn* is the PC Language ID number from step 1.
4. Run Logger32: it opens the specified DLL for the language used on your PC.



◀ You may not have even noticed until now … but the language is mentioned in blue text up at the top of the splash screen message as Logger32 starts up.

30.2 Generating (translating) language DLLs

If you have the knowledge and time to help your compatriots, you are encouraged to make a Logger32 language DLL for *your* language, following the instructions below.

Logger32 menus, warning/error messages and some field names can be displayed in languages other than US English. If the required language DLL is not already available, you 'simply' need to translate the original English text and compile the result to produce your own language DLL.

Hinson tip: read right through the following instructions before you start. Visual Basic or Delphi is needed, for example, to compile the translated files, generating the new language DLL. Also, please note that the text gets updated by Bob from time to time, so your language DLL will gradually fall behind unless you maintain it – a simple process but worth bearing in mind.

Phase 1: translation

In the Support Files area of the Hamlogger group, under the Coding Support area, click the PC icon to download "**English text for translation to make other language DLLs – 4.0**". It is *.zip* archive containing two files:

- *ErrorsAndWarnings.txt* has source errors and warning messages.

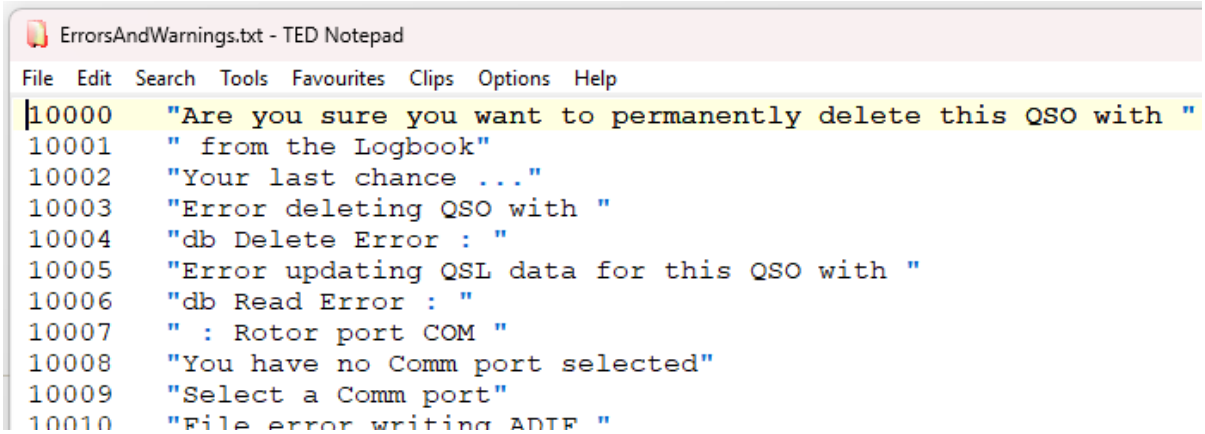
Copyright © 2026 Robert C. Furzer K4CY

- *Menus.txt* has menu items.

Both files have the same structure:

- First comes the text identifier number, followed by a <**Tab**>.
- Next is the text for this text ID.

1. In a plain text editor such as Notepad or Ted, insert the content of *ErrorsAndWarnings.txt* after item 9836 of *Menus.txt*
2. Now translate the content of *Menus.txt*:
   - Items marked "hidden" need not be translated as they are not visible in Logger32.
   - There are menus of the form "This && that". Wen the "&" character is used as part of the text content in a menu it must be doubled-up as "&&" to prevent Logger32 identifying it as a hotkey/shortcut …
   - Logger32 uses <**Alt**+**letter**> shortcuts to menu items, generally using the first unique letter of the menu text *e.g.* <**Alt**+**V**> opens the **V**iew menu. If your translated words clash with other shortcuts (including any global hotkeys), strange things may happen. For example, the Polish word for View is Widok: using the Polish translation, when a user presses <**Alt**+**W**> hoping to **W**ipe the log entry pane, the <**View**> menu opens instead. If menu items share the same first letter, you must identify different shortcut letters. To do this, find the menu item in the source RES file, select a different letter from the item to be used for the shortcut, and precede it with "&" character *e.g.* Wid&ok instead of Widok. This menu item will be visible in Logger32 as Wid**o**k, with <**Alt**+**O**> being the shortcut.
3. Translating the *ErrorsAndWarnings* messages is a little more involved …



The text is placed between pairs of double quotation marks ▲. These quotation marks are specifically used to indicate the start/end of text and **must be deleted during translation, *leaving* the beginning/ending spaces in place**. The example below shows the reason for the spaces. As seen above, the first two *ErrorsAndWarnings* messages are:

10000 "Are you sure you want to permanently delete this QSO with "

10001 " from the Logbook?"

These two messages are displayed in sequence by Logger32 when the user is deleting a QSO from the log. For example, if it was a QSO with SP7DQR, the following message will be displayed:

Are you sure you want to permanently delete this QSO with SP7DQR from the Logbook?

Between message 10000 and message 10001, Logger32 inserts the callsign SP7DQR. If spaces at the end of 10000 and the beginning of 10001 were missing, the message would be:

Copyright © 2026 Robert C. Furzer K4CY

Are you sure you want to permanently delete this QSO withSP7DQRfrom the Logbook?

If your language's grammar requires a different word sequence, you can freely manipulate the contents of the lines but **no line can be totally empty**. If you have no text for a line, follow the line number and tab with at least one character (*e.g*. just a space).

4. Add two lines to the top of the file, above *40 Data Terminal*, in this order:
   - The language name *e.g*. Polish
   - The language ID for your PC (see above) *e.g*. 2057.

Phase 2: create a *.res* resource file

Aki JA1NLX has kindly published a utility for this:

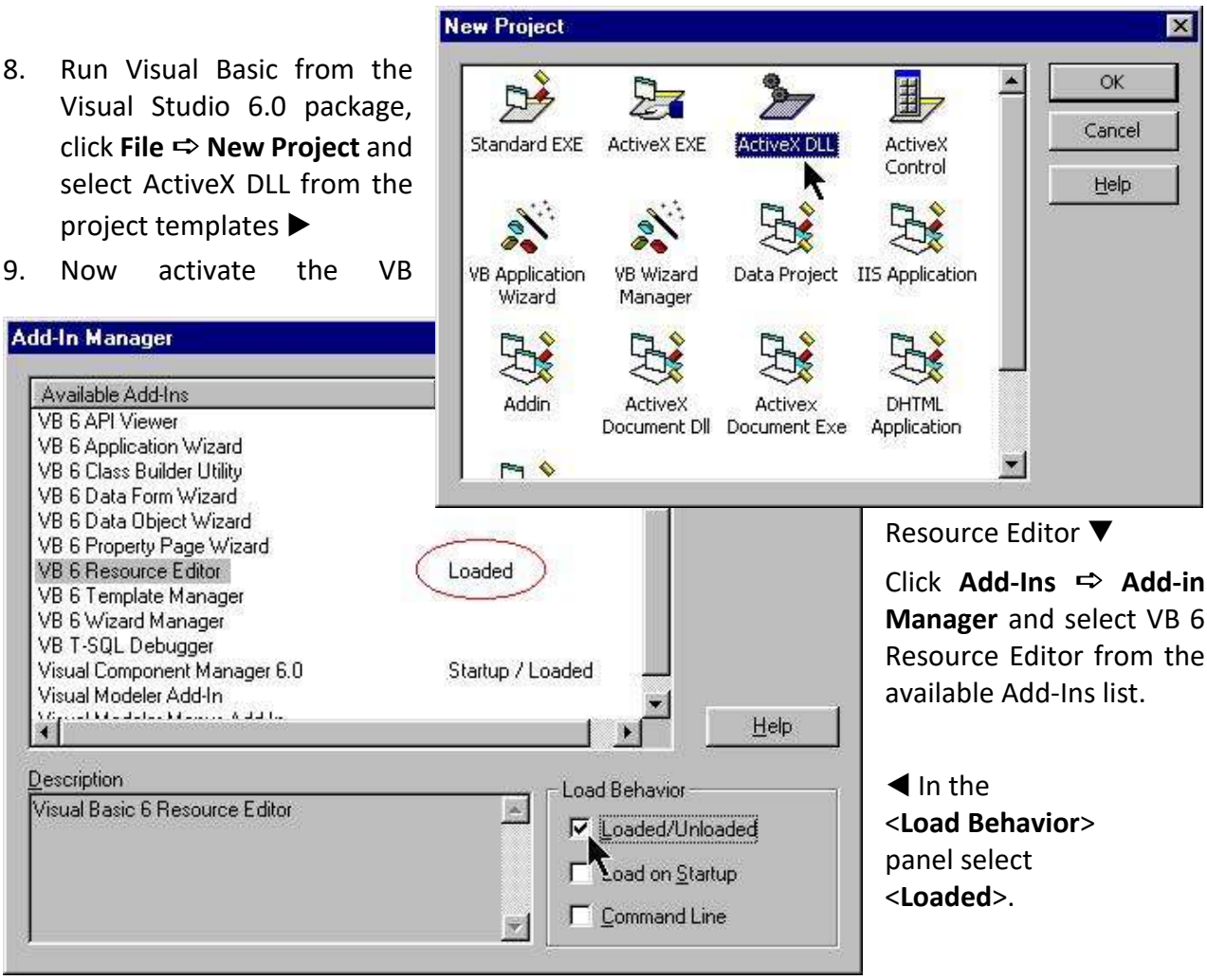
5. Download, unpack and then run ResConv from
   https://ja1nlx.web.fc2.com/logger32/UtilityProgram/ResConv.zip
6. Unzip ResConv and run it.
7. Follow the prompts to convert your translated *.txt* file to a *.res* file.

Phase 3: compile DLL

The *.res* file needs to be compiled into a *.DLL*. There are three possible methods …

Method A: compile DLL using Microsoft Visual Basic

8. Run Visual Basic from the Visual Studio 6.0 package, click **File ⇨ New Project** and select ActiveX DLL from the project templates ▶
9. Now activate the VB Resource Editor ▼



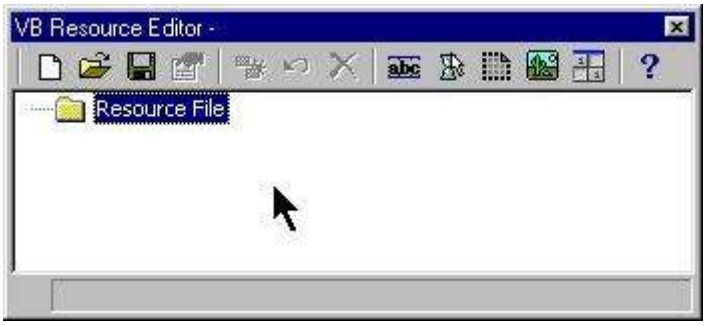
Click **Add-Ins ⇨ Add-in Manager** and select VB 6 Resource Editor from the available Add-Ins list.

◀ In the <**Load Behavior**> panel select <**Loaded**>.

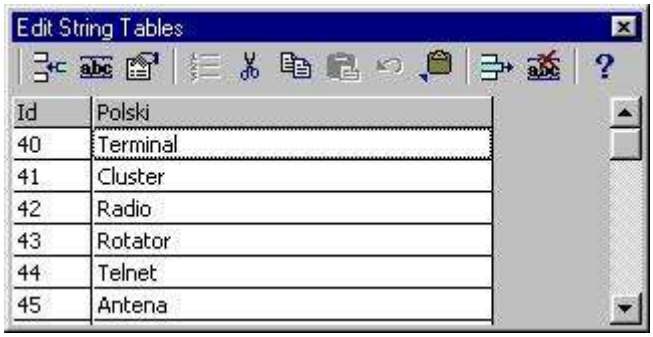
Copyright © 2026 Robert C. Furzer K4CY

Click <**OK**>

If the VB Resource Editor is not already visible, click **Tools ⇨ Resource Editor** to open it ▶



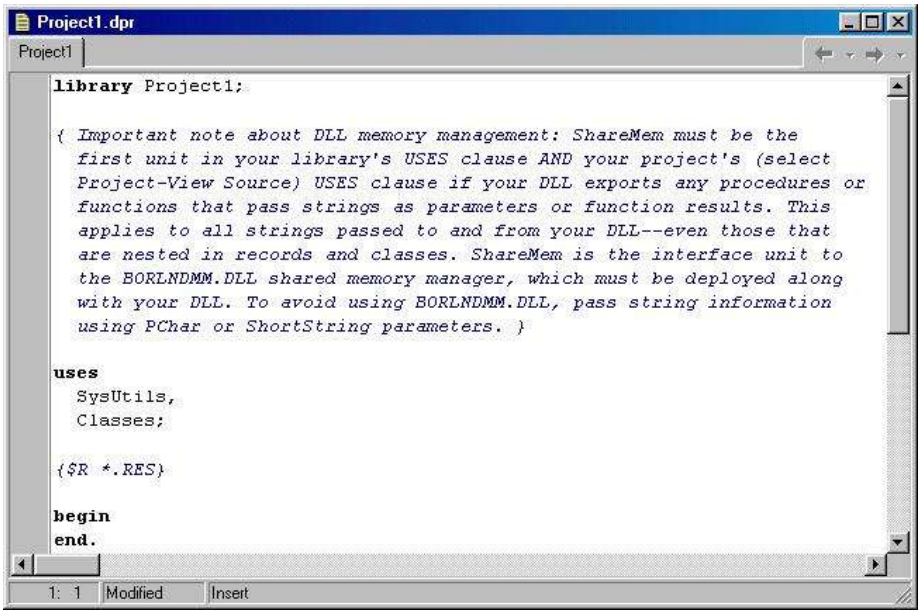
10. Click the Open Resource File icon or select <**Open**> from the pop-up menu. Select your *.res* file to load and load it. The String Table will be added to your resource.
11. Click the Edit String Tables icon to open the editor ▶
12. Check one last time and make any necessary changes.
13. If you decide to manually insert all items into the String Table (without the *.res* file created by the ResMan program), don't select <**Open Resource File**> but instead click the Edit String Tables icon. An empty Edit String Tables window is displayed. All text ID and items can now be typed directly to the String Table.



14. Click to open the <**File**> menu. Select the folder where the DLL must be written and type in name such as *1045.dll*. Click <**OK**>: The "Compiling" and "Writing" messages will be displayed on the toolbar.
15. Copy your new Language DLL to your *C:\Logger32* folder.
16. Run Logger32 and gasp excitedly in amazement at seeing the menu items and messages displayed in your chosen language.

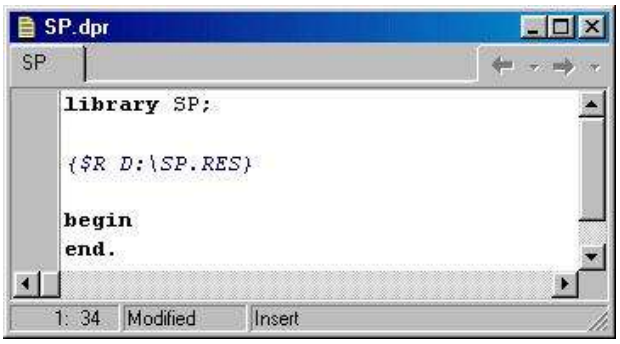
Method B: compile DLL using Borland Delphi 5.0 Standard

8. Run Delphi 5.0, click **File ⇨ New** and click the DLL icon to open an empty Library Project ▶
9. Only four lines are required, so feel free to delete the comment lines from the project to shrink the DLL in half.



Copyright © 2026 Robert C. Furzer K4CY

10. Replace **.RES* with your RES resource file name and location, and save your project. Mine was called *D:\SP.RES* and I saved the project as SP.



◀ After these operations, my project looked like this, with just 4 lines.

11. Click **Project ⇨ Build Project (Build SP for my file)** to create *SP.DLL* in the SP project folder (in my case).

Method C: phone a friend

8. If you don't have access to VB or Delphi, ask around among your ham friends for someone who does, then help them work through methods A or B as appropriate.
9. If you have no friends, or at least none with VB or Delphi, beg for help on the Hamlogger reflector.

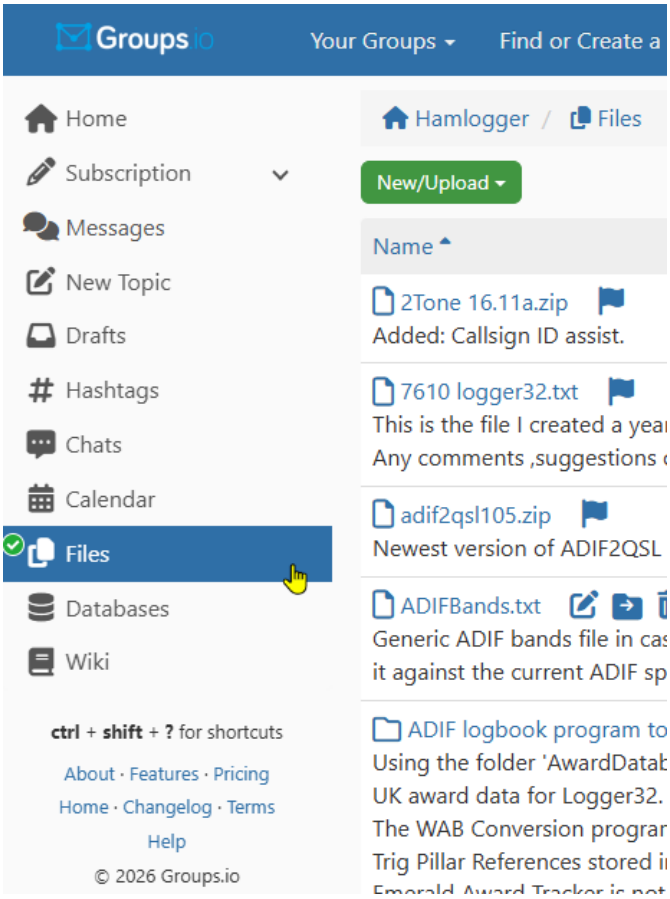
Phase 4: rename and test the translated DLL

Use your PC Language ID as the name for the DLL (*e.g. 5129.dll*), then load it into Logger32 using the instructions above and check that it works OK.

Phase 5: share it

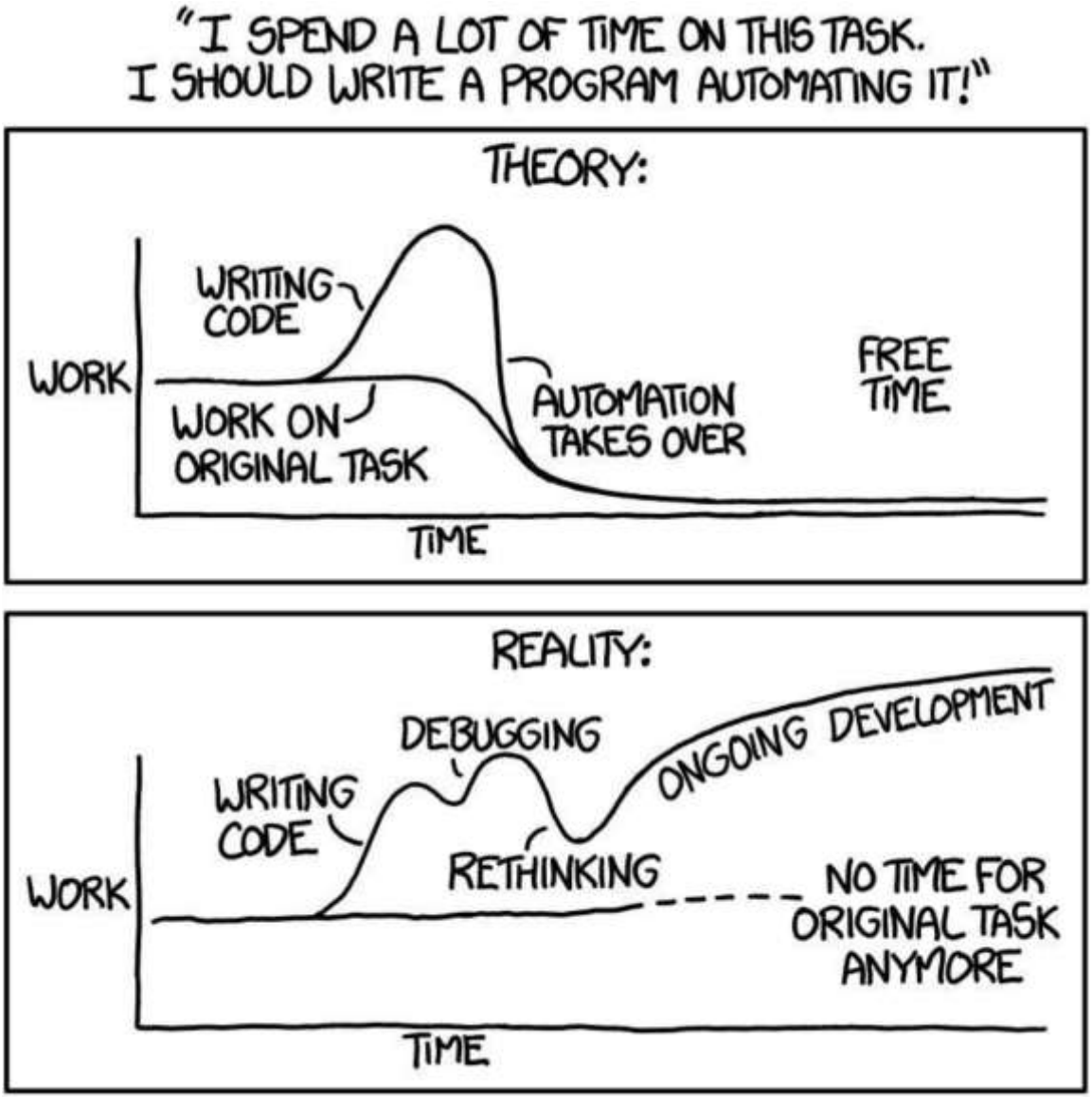
Login then copy the DLL to the Hamlogger group's **Files** area ▶

Finally, announce it on the Hamlogger reflector so your countrymen can give it a go too, and bask in the glory of their eternal gratitude.



Copyright © 2026 Robert C. Furzer K4CY

31 Application Program Interfaces (APIs)



A series of Windows messages are available to exchange data between Logger32 and external applications (apps), using ADI-formatted text strings.



A skeleton Visual Basic application (with source code) is available from the Support Files area of Logger32.net as a starting point for those writing applications to use Logger32 APIs ▶

Copyright © 2026 Robert C. Furzer K4CY

31.1 Establishing communication

The first step in establishing communication between your app and Logger32 is for your app to identify the hWnd of the Logger32 MDI:

```
Dim L32hWnd as long

L32hWnd = FindWindow(vbNullString, “Logger32”)
```

If Logger32 is not running, your app can check periodically until it *is* running.

31.1.1 Initial Windows messages to Logger32

Secondly, since Logger32 supports up to 5 simultaneous external apps, your app must register/identify an unused Windows message (called “Logger32 1” to “Logger32 5”).

31.1.2 Message 1

```
Dim L32Msg as long

L32Msg = RegisterWindowMessage(“Logger32 1”)

PostMessage L32hWnd, L32Msg, 1, GhW ‘ GhW is the hWnd of your main form
```

Logger32 replies with message number 0. If the L32Msg you send is in use by another external app, the lParam will be set to 0 (no go). If the L32Msg is unused, then the lParam will be set to 1 (go ahead).

If your request for a given L32Msg was rejected (lParam is 0), your app should retry using a different L32Msg:

```
L32Msg = RegisterWindowMessage(“Logger32 2”)

PostMessage L32hWnd, L32Msg, 1, GhW ‘ GhW is the hWnd of your main form
```

Your application can retry the connection using a RegisteredWindowMessage between “Logger32 1” and “Logger32 5”. In the unlikely event that *all 5* message numbers are already assigned to or tied up by other external apps, yours is out of luck.

Hinson tip: with only 5 RegisteredWindowMessage streams available, external apps that fail to release a stream when they terminate normally or crash can soon exhaust the supply, forcing users to restart Logger32 or reboot their PCs so that Logger32 can communicate with an external app without generating an error message. When your app re-starts following a crash, can it somehow locate and pick up the stream it used previously rather than grabbing and hogging another? Failing that, how about releasing the stream in your app’s error-handler? You do trap and handle errors, right? See Message 4 below.

31.1.3 Message 2

When your application has successfully connected to Logger32 (you receive a message number 0 with an lParam of 1), send a message number 2 to Logger32 with an lParam of the TextBox.hWnd to which Logger32 is to send text strings (if any). If your app does not want unsolicited data from Logger32, set the lParam to 0:

```
PostMessage L32hWnd, L32Msg, 2, TextBox.hWnd
```

Copyright © 2026 Robert C. Furzer K4CY

31.1.4 Message 3

Immediately following the message 2 to Logger32 you have the *option* to send a message 3. Setting the lParam to 0 (or simply not sending a message 3) will tell Logger32 to not send any DX spot information to your app (most apps will not want to receive DX spots from Logger32). Setting the lParam to 1:

PostMessage L32hWnd, L32Msg, 3, 1

tells Logger32 to send ADI-formatted text to your app for each DX spot received by Logger32[400].

Logger32 responds by doing three things:

1. It sends your app a message number 3. The lParam of this message is the hWnd of a TextBox in Logger32 to which your app sends ADI formatted text strings.
2. It sends your app a message 99. The lParam of this message identifies the radio currently in use (either "1"or "2").
3. It sends your app a message 100. The lParam of this message is the radio frequency in Hz.

If your app has OK'd the receipt of unsolicited text strings from Logger32 (you sent a TextBox hWnd as the lParam of your message number 2), Logger32 sends your app a WM_SETTEXT message, putting the current radio mode in the TextBox you have specified. The message text is of the form <APP_RADIO_MODE:3>SSB

31.2 Additional messages *to* Logger32

At this point, the basic connection between Logger32 and your app is complete. Logger32 will respond to the following additional messages from your app:

Message 4. lParam is 0. Your app tells Logger32 it has stopped. Logger32 releases the RegisteredWindowMessage your app had been using, ready for re-use. Nice!

Message 5. lParam is 0. Your app tells Logger32 to shut down.

Message 6. lParam is 0. Your app relinquishes PTT control. Logger32 assumes PTT control and initializes the PTT ports/keying lines.

Message 7. lParam is 0. Your app assumes control of the PTT.

Message 8. lParam is 0. Your app tells Logger32 to assert the PTT.

Message 9. lParam is 0. Your app tells Logger32 to release the PTT.

Message 10. lParam is the hWnd of a TextBox in your app to write to. Logger32 will respond with an ADI-formatted text string in the following format <APP_SET_FREQ_MODE:27>18132.012|CW The frequency is in kHz with the decimal separator as specified in the PCs language & region settings.

Message 11. lParam is 0. Your app tells Logger32 to release CAT control of the radio.

Message 12. lParam is 0. Your app tells Logger32 to take CAT control of the radio.

---

[400] If you have previously set the message 2 lParam to 0 (no TextBox hWnd is offered to Logger32) and you set the message 3 lParam to 1 (telling Logger32 to send your app DX spot information), Logger32 will complain, generate a warning message and turn off the request.

Copyright © 2026 Robert C. Furzer K4CY

Message 13. lParam is 0. If Logger32's <**Mode from Sound Card**> is enabled, <APP_FORCE_MODE:x> messages from your app will change Logger32 mode.

Message 14. lParam is 0. Disables the feature turned on by message 13. Remember to turn this off when your app closes.

Message 15. If lParam is 1, a flag is set so that Logged QSOs will be flagged for QSLing. If lParam is 0, the flag is unset.

Message 16. If lParam is 1, a flag is set so that Logged QSOs will be flagged for eQSLing. If lParam is 0, the flag is unset.

Message 17. If lParam is 1, a flag is set so that Logged QSOs will be flagged for LoTW. If lParam is 0, the flag is unset.

Message 18. If Lparam is 1, Logger32 radio polling is turned off. If Lparam is 0, Logger32 radio polling is turned on. If your app disables Logger32 radio polling, remember to turn it back on when your app closes.

Message 19. If lParam is 0, Logger32's Mode from bandplan option is disabled. If lParam is 1, it is enabled. If lParam is 2, Logger32 responds with a message 103 indicating the Mode from bandplan in the lParam. If your app uses message 19, remember to set the value back to the original setting when your app closes.

Message 20. lParam is ignored. An ExternalInternetCallsignLookup module sends this message on lookup completion. On receipt of this message, Logger32 looks for and processes the data file written by the external lookup module.

Message 21. lParam is ignored. When Logger32 receives this message, the export LoTW file process is started. On completion, Logger32 sends unsolicited messages to the app: <APP_LoTW_RECORDS:xx> and <APP_LoTW_FILENAME:xx>.

Message 22. lParam is ignored. When Logger32 receives this message, the export eQSL file process is started. On completion, Logger32 sends unsolicited messages to the app: <APP_eQSL_RECORDS:xx> and <APP_eQSL_FILENAME:xx>.

Message 23. lParam is ignored. When Logger32 receives this message, the sync LoTW process is started. Before sending this message, the app should send an <APP_IMPORT_SYNC_FILENAME:xx>xxx message to Logger32 with the filename to be imported.

Message 24. lParam is ignored. When Logger32 receives this message, the sync eQSL process is started. Before sending this message, the app should send an <APP_IMPORT_SYNC_FILENAME:xx>xxx message to Logger32 with the filename to be imported.

Message 25. An lParam of 1 tells Logger32 to write an ADI-formatted record to *eQSL Dump File.txt* in the app folder, for each QSO logged. If the lParam is 2, the ADIF record is written and the EQSL_QSL_SENT field in the logbook is set to 'Y'. If the lParam is 0, the function is turned off and no ADI-formatted records are written. Logger32 remembers any previous setting of this function. It is not necessary to send this message every time the external app is executed, only when the user wishes to change the setting. Informational logbook entries such as G3NPA= are not classed as QSOs and so are not written to the file.

Message 26. An lParam of 1 tells Logger32 to write an ADI-formatted record to *C:\Logger32\ADIF audit trail.txt* as each QSO is logged or modified. An lParam of 0 tells Logger32 *not* to write the record. It is not necessary to send this message every time the external app is executed, only when the user wishes to change the setting. New QSOs added to the logbook, and written to the *ADIF*

Copyright © 2026 Robert C. Furzer K4CY

audit trail.txt file, are prefixed with the text <APP_QSO_LOGGED:0>. QSOs deleted from the logbook, and written to *ADIF audit trail.txt*, are prefixed with the text <APP_QSO_DELETED:0>. QSOs that are modified in the logbook are written to *ADIF audit trail.txt* as record pairs. First comes the unmodified QSO prefixed with the text <APP_QSO_DELETED:0>. Next comes the modified QSO prefixed with <APP_QSO_LOGGED:0>. It is the responsibility of the external app to do whatever file maintenance/cleanup of the *ADIF audit trail.txt* as necessary. All logbook entries (*including* informational logbook entries such as G3NPA=) are written. Remove them on receipt if they are not required by your app.

Message 27. Almost the same as message 26, message 27 enables/disables writing to *C:\Logger32\ClubLog Dump File.txt*. Informational logbook entries are *not* written.

Message 28. Sending this message to Logger32 (params are ignored) opens the External Interface Debug window allowing message sequences to be monitored. This window can also be turned on by adding an entry to *C:\Logger32\Logger32.INI*:

[Globals]

External Interface Debug=1

Message 29. Sending this message to Logger32 (params are ignored) makes Logger32 send all current QSO information from the log entry pane.

Message 30. lParam is ignored. When Logger32 receives this message, the export QSL file process is started. On completion, Logger32 sends unsolicited messages to the app: <APP_QSL_RECORDS:xx> and <APP_QSL_FILENAME:xx>.

Message 120. Send this message to Logger32 if you have written data to a disk file as the result of an Internet lookup.

Message 121. When an external application sends message 121 with lParam *QSO Number* to Logger32, Logger32 deletes the LoTW_SEND flag and sets LoTW_SENT to "Y" for the QSO with this *QSO Number* in the Logbook. If an external application sends message 121 with lParam *0* (zero) to Logger32, Logger32 deletes the LoTW_SEND flags and sets LoTW_SENT to "Y" for *all* QSOs in the Logbook.

31.3 Additional messages *from* Logger32

Logger32 will send the following additional messages to your app:

Message 99. This message is sent prior to message 100. The Lparam contains the number of the radio currently in use (1 or 2).

Message 100. This message is sent whenever radio 1 frequency changes. The Lparam contains the radio frequency in Hz. Message 100 is always preceded by message 99.

Message 101. This message is sent if Logger32 has calculated a valid beam heading and the user types CTL_A or ALT_A. The Lparam contains the Short Path direction (if the user typed CTL_A) or Long Path direction (if the user typed ALT_A). The directions are calculated to include any/all Logger32 corrections the user has set. Logger32 can report directions either side of the range 0˚ to 360˚ so the external app must check and correct the values if necessary.

Message 102. This message is sent when the user clicks a DX spot.

Copyright © 2026 Robert C. Furzer K4CY

Message 103. This message is sent in response to receipt of a message 19 with Lparam of 2. The message 103 has an Lparam of 0 if Logger32's Mode from bandplan option is off, or 1 if the Mode from bandplan is on.

Message 104. This message is sent whenever radio 2 frequency changes. The Lparam contains the radio frequency in Hz. Message 104 is always preceded by message 99.

Message 105. Logger32 sends this message to the app every time a QSO is written to the log and an ADIF-formatted message is written to the eQSL Dump File. When an ADIF file is imported into Logger32 a single message 105 is sent on completion of the import. This function is enabled/disabled when Logger32 receives a message 25. Prior to sending message 105, Logger32 sends a <APP_eQSL_DUMP_FILENAME:xx>filename to the app.

Message 106. Logger32 sends this message every time a record is written to the *ADIF audit trail.txt* file *e.g.* when a QSO is deleted from the logbook and an ADIF-formatted message is added to the file. The record has the additional field of <APP_QSO_LOGGED:0> to indicate why the record was added to the file. Logger32 sends your app a message 106 with an Lparam of 0. When a logged QSO is added, an ADIF-formatted record is added to the file. The record has the additional field of <APP_QSO_LOGGED:0> to indicate why the record was added to the file. Logger32 sends your app a message 106 with an Lparam of 1. When a logged QSO is modified, an ADIF-formatted record is added to the file. The record has the additional fields of <APP_QSO_MODIFIED:0> and <APP_HRD_APIKEY:xx>callsign|date|time (callsign, date and time are the values prior to modification) to indicate why the record was added to the file and to provide the necessary keys for the app to identify the original QSO record that was modified. Logger32 sends your app a message 106 with an Lparam of 2.

Message 107. Logger sends this message to the app every time a QSO is written to the logbook and an ADIF-formatted message is written to the Club Log dump file. When an ADIF file is imported into Logger32 a single message 105 is sent on completion of the import. This function is enabled/disabled when Logger32 receives a message 27. Prior to sending message 107, Logger32 sends a <APP_CLUBLOG_DUMP_FILENAME:xx> filename to the app.

Message 108. If the currently selected rotator is "External Interface" Logger32 sends this poll message to the rotator app to request the rotator's azimuth. The app should reply with a message 108, giving the azimuth in the Lparam.

Message 109. Logger32 sends this message when an automatic callsign lookup is triggered.

Message 110. Logger32 sends this message when the antenna selector is changed. The lParam contains the antenna number (either decimal or BCD).

Message 111. Logger32 sends this message to external apps when a QSO is logged. The message is not sent if QSOs are imported or added from the ADD Window.

31.4 ADIF text strings

Data is exchanged between your app and Logger32 by writing ADIF-formatted text strings to identified TextBoxes. Use the TextBox_Change event as a trigger to process the ADIF data received.

Copyright © 2026 Robert C. Furzer K4CY

31.4.1 Logger32 recognizes the following ADIF strings

<EOR> If received without additional text, Logger32 clears all current entries in the log entry pane.

<APP_TAB> If received without additional text, this simulates <**Tab**>bing or moving the focus away from the Call field of the log entry pane. Automatic callsign lookup functions such as QRZ lookup, previous QSOs lookup, carry-forward of selected data from previous QSOs *etc*. are triggered.

<CALL:x> The callsign is sent to the Call field of the log entry pane.

<RST_RCVD:x> The RST Received is sent to the log entry pane.

<RST_SENT:x> The RST Sent is sent to the log entry pane.

<NAME:x> The person's name is sent to the log entry pane.

<APP_TIME_ON:x> in Microsoft's timestamp format (*e.g.* 39470.6737384259) sets the ADIF QSO_DATE and TIME_ON fields.

<APP_TIME_OFF:x> in Microsoft's timestamp format sets the ADIF TIME_OFF field.

<APP_QSL:1>Y sets the Logger32 QSL flag. Any other character(s) *un*sets (resets) the flag.

<APP_eQSL:1>Y sets the Logger32 eQSL flag. Any other character(s) *un*sets the flag.

<APP_LoTW:1>Y sets the Logger32 LoTW flag. Any other character(s) *un*sets the flag.

<COMMENT:x> sends a comment to the log entry pane.

<QTH:x> sends the QTH to the log entry pane.

<ADDRESS:x> sends the address to the log entry pane.

<STATE:x> if the user has assigned STATE as a user field on the log entry pane, the Primary Administrative Subdivision is sent to the log entry pane.

<CNTY:x> if the user has assigned CNTY as a user field on the log entry pane, the Secondary Administrative Subdivision is sent to the log entry pane.

<GRID SQUARE:x> if the user has assigned GRID SQUARE as a user field on the log entry pane, the grid square is sent to the log entry pane.

<IOTA:x> if the user has assigned IOTA as a user field on the log entry pane, the IOTA reference is sent to the log entry pane.

<STX:x> if the user has assigned STX as a user field on the log entry pane, the transmitted serial number is sent to the log entry pane.

<SRX:x> if the user has assigned SRX as a user field on the log entry pane, the received serial number is sent to the log entry pane.

<QSL_VIA:x> if the user has assigned QSL_VIA as a user field on the log entry pane, the QSL routing information is sent to the log entry pane.

<QSLMSG:x> if the user has assigned QSLMSG as a user field on the log entry pane, a QSL message is sent to the log entry pane.

<SAT_NAME:x> if the user has assigned SAT_NAME as a user field on the log entry pane, the satellite name is sent to the log entry pane.

<SAT_MODE:x> if the user has assigned SAT_MODE as a user field on the log entry pane, the satellite mode is sent to the log entry pane.

Copyright © 2026 Robert C. Furzer K4CY

<PROP_MODE:x> if the user has assigned PROP_MODE as a user field on the log entry pane, the propagation mode is sent to the log entry pane.

<FREQ_RX:x> if the user has assigned FREQ_RX as a user field on the log entry pane, the receive frequency is sent to the log entry pane.

<TEN_TEN:x> if the user has assigned TEN_TEN as a user field on the log entry pane, the TenTen number is sent to the log entry pane.

<USER_1:x> if the user has assigned USER_1 as a user field on the log entry pane, the user defined data is sent to the log entry pane.

<USER_2:x> if the user has assigned USER_2 as a user field on the log entry pane, the user defined data is sent to the log entry pane.

<USER_3:x> if the user has assigned USER_3 as a user field on the log entry pane, the user defined data is sent to the log entry pane.

<APP_LOGQSO:x> logs the data currently in the log entry pane. This may be sent as standalone data, or part of a complete QSO to be logged.

<FREQ:x> if Logger32's radio type is none, this value (in kHz) *simulates* a change of radio frequency.

<MODE:x> if the data received does not match the mode of the log entry pane, a warning message is generated.

<APP_FORCE_FREQ:x> - if a message 18 with Lparam of 1 has been sent to Logger32 (radio polling has been turned off), this changes the frequency in the log entry pane.

<APP_FORCE_MODE:x> changes the mode in the log entry pane. The message is intended to be used when Logger32 does not have control of the radio. Otherwise, the mode you have forced may be overwritten by the radio's mode when next polled.

<APP_SET_FREQ_MODE:x> sets the radio frequency and radio mode to the data received. The format is (say) <APP_SET_FREQ_MODE:14>14003.451|CW-R with the frequency in kHz.

<APP_SET_FREQ:x> sets the radio VFO A frequency in kHz *e.g.* <APP_SET_FREQ:9>14003.451

<APP_SET_MODE:x> sets the radio mode *e.g.* <APP_SET_MODE:3>USB. For this to work, Logger32's <Mode from bandplan> or <Mode from Radio> must be enabled.

<APP_SET_MODE_DIGITAL:x> sets the radio mode for radios that have modes such as USB-D. The format is (say) <APP_SET_MODE_DIGITAL:3>USB. For this to work, Logger32's <Mode from bandplan> or <Mode from Radio> must be enabled.

<APP_CLICK_DXSPOT:x> simulates clicking a DX spot in Logger32. Your app must pass both the frequency (in kHz) and the DX callsign *e.g.* <APP_CLICK_DXSPOT:13>14003.01|K4CY

<APP_IMPORT_SYNC_FILENAME:xx> is sent to Logger32 prior to sending message 23 or 24, giving the file to be imported.

<APP_CLICK_RCP_BUTTON:x> - your app sends this ADIF message to Logger32 to click a **R**adio **C**ontrol **P**anel macro button. x is the button number in the range of 1 to 48. The RCP does not have to be open for this to work.

<APP_QSO_UPDATE:0>*&ADIF field data&*<EOR> Logger32 updates the logbook's ADIF field NAME, ADDRESS, STATE, CNTY, IOTA, GRID SQUARE, QSL_VIA or QTH *e.g.* <IOTA:6>AS-007<GRID SQUARE:4>PM95

Copyright © 2026 Robert C. Furzer K4CY

<APP_OVERWRITE:1> The value 0 means only populate empty fields, whereas 1 means overwrite existing fields with new data.

Logger32 sends the following ADIF fields to your app (assuming unsolicited data are accepted):

- <**CALL:x**>, <**RST_SENT:x**>, <**RST_RCVD:x**>, <**NAME:x**>, <**MODE:x**>, <**BAND:X**>, <**GRIDSQUARE:x**>, <**IOTA:x**>, <**APP_COUNTRY:x**>, <**STATE:x**>, <**APP_CNTY:x**>, <**CONT:x**>, <**CQZ:x**> and <**ITUX:x**> are sent when the user interacts with the log entry pane.
- **<APP_RADIO_MODE:x>** is sent when the radio mode is changed.
- **<APP_RADIO_MODE_SECONDARY:x>** is sent when the secondary radio mode is changed.
- **<APP_DXSPOT_CALLSIGN:x>** is sent when a DX spot is received.
- **<APP_DXSPOT_DXCC:x>** is sent when a DX spot is received.
- **<APP_DXSPOT_FREQ:x>** is sent when a DX spot is received.
- **<APP_DXSPOT_BAND:x>** is sent when a DX spot is received. The band is derived from the Logger32 bandplan.
- **<APP_DXSPOT_MODE:x>** is sent when a DX spot is received. The operation mode (SSB, CW, RTTY, *etc*.) mode is derived from the Logger32 bandplan.
- **<APP_DXSPOT_TIME:x>** is sent when a DX spot is received.
- **<APP_DXSPOT_COLOR:x>** is sent when a DX spot is received *if* the spot is highlighted.
- **<APP_DXSPOT_CLICKED:x>** is sent when a DX spot has been clicked. The format is: <APP_DXSPOT_CLICKED:18>HB9RDE|14225.0|SSB
- **<APP_PTT_STATUS:x>** is sent when Logger32 receives a message 6, 7, 8 or 9. Additional message may be sent if messages are ignored *e.g.* if PTT is OFF but message 9 arrives.
- **<APP_LoTW_RECORDS:x>** is a count of exported QSOs sent when Logger32 completes a LoTW file export.
- **<APP_LoTW_FILENAME:xx>** is the filename of an LoTW export.
- **<APP_eQSL_RECORDS:x>** is a count of exported QSOs sent when Logger32 completes an eQSL file export.
- **<APP_eQSL_FILENAME:xx>** is the filename of an eQSL export.
- **<APP_QSO_UPDATE:x>y <EOR>** is a callsign to lookup *e.g.* <APP_QSO_UPDATE:4>K4CY<EOR>.
- **<EOR>** is sent when Logger32 clears the log entry pane.

31.5 Things to consider



If you write an app or interface to Logger32, *please* ensure your code handles frequency strings with either periods or commas as the decimal separator, depending on the user's PC language & region settings.

Also, please ensure you handle the connect sequence correctly so that you don't tie up more than one RegisteredWindowMessage. The sample code provides a working example.

Finally, please release any resources (including your assigned RegisteredWindowMessage) when your app closes.

Run along and play nicely now.

Copyright © 2026 Robert C. Furzer K4CY

32 The TCP Client-Server

> "In the realm of servers,
> stability and speed are
> the cornerstones of success"
>
> *Linus Torvalds*

A growing number of radio applications today, and indeed radios, are communicating with each other through UDP and TCP messages, sharing information such as callsigns, QSOs, radio frequency and mode *etc*. Thanks to networking, Logger32 and radio apps such as JTDX, JTAlert, N1MM+ may even be running on different computers (not necessarily all Windows machines) but, if they are connected to a TCP/IP network, they can share QSO and radio/CAT data.

Hinson tip: you may be wondering why TCP and UDP *networking* protocols are being used even though, generally, radio apps are all running on the same shack computer. The answer is two-fold. (1) UDP and TCP are popular, reasonably robust and efficient background messaging standards, with support available on most platforms. (2) Networking protocols make it possible to exchange messages between apps running on the same *or* on separate computers on the network (*e.g*. for performance reasons), perhaps even over the Internet (as in remote shacks and mobile ops).

Whereas UDP messaging is described elsewhere (particularly the UDP BandMap), this chapter explains how to set up Logger32 and other apps to use TCP messaging.

32.1 About TCP

Transmission **C**ontrol **P**rotocol **(TCP)** vaguely resembles an amateur QSO in that it involves:

- A CQer (the client) and a caller (the server) with unique callsigns (their IP addresses) sharing the same frequency (a numbered network port such as 52001);
- Modulation (message encoding by the client) and demodulation (message decoding by the server) using the same mode and submode (protocols such as TCP, UDP, IP, ARP, ICMP …);
- Various communications media (*e.g*. network cables, RF, audio or light links) providing a somewhat noisy, unreliable path linking the parties, requiring dynamic adjustments to speeds and message lengths, and automatic re-transmission of parts of a message (packets) that were not QSLd (acknowledged) by the recipient:
    - Unlike UDP which just broadcasts data and hopes it is received OK, TCP is connection-oriented. Every TCP packet is assigned a **sequence number** (starting with a random number) and incremented in successive messages – much like a contest serial number.
    - When Station A sends a packet, it starts a timer. If Station B receives it correctly, it returns an ACK. If Station A doesn't receive that ACK before its timer expires (perhaps due to QRM) it assumes the packet was lost in QRM and automatically retransmits it, repeating the same sequence number. It may try many more times before eventually giving up in disgust.

Copyright © 2026 Robert C. Furzer K4CY

- Meanwhile, it may have sent other packets with their own sequence numbers which allow the receiver to reassemble the message in the correct sequence;

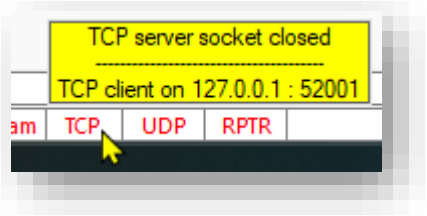
- A conventional, structured process to initiate communications between the parties and pass messages through the link (the three-way handshake):
  1. **SYN** (**SYN**chronize): station A (the **TCP client**) transmits a CQ message (**packet**) on a specific frequency (port or socket) flagged 'SYN' and tagged with an initial message ID (starting sequence number) then listens for a reply on the same frequency.
  2. **SYN-ACK:** station B (the **TCP server**) calls station A on his frequency, adding 1 to acknowledge station A's message ID and providing station B's initial message ID.
  3. **ACK** (**ACK**nowledge): station A responds to station B, establishing a bidirectional QSO.
- Messages are exchanged, back and forth, until either party goes QRT, closing each leg of the connection (client-to-server and server-to-client) with FIN and ACK message pairs.

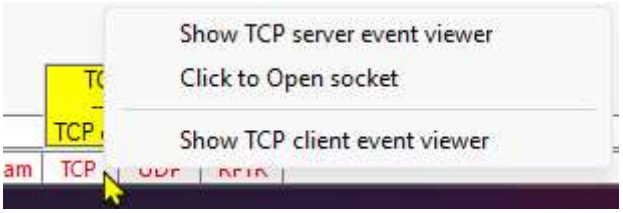
32.2 Configuring and monitoring TCP in Logger32

32.2.1 The TCP client

TCP is configured through the <**TCP**> panel on the status bar at the bottom of Logger32's window. Initially the <**TCP**> text is red, indicating that TCP is closed, dormant, inactive.

Mouse-over it for a yellow pop-up showing the status ▶

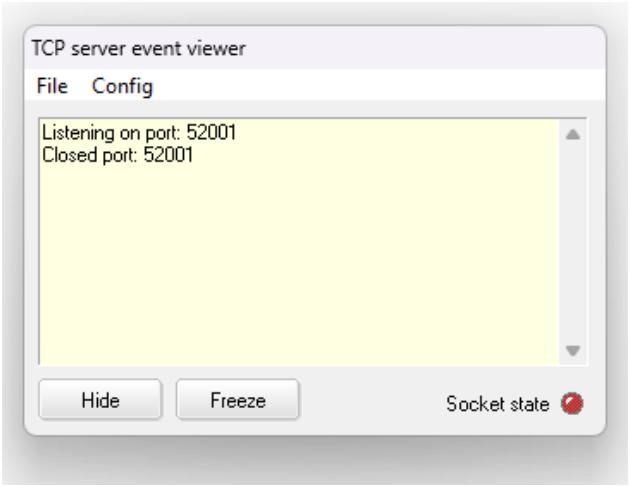




◀ Right-click the <**TCP**> panel for the configuration options.

- <**Show TCP server event viewer**> displays a window and briefly checks the port, decoding and showing any TCP messages being processed by the TCP server ▶ The <**Socket state**> "LED" turns green while the port is checked but almost immediately goes red if there is currently no TCP connection. Click the <**Freeze**> button to freeze the display if loads of messages are flowing through, too fast to read.

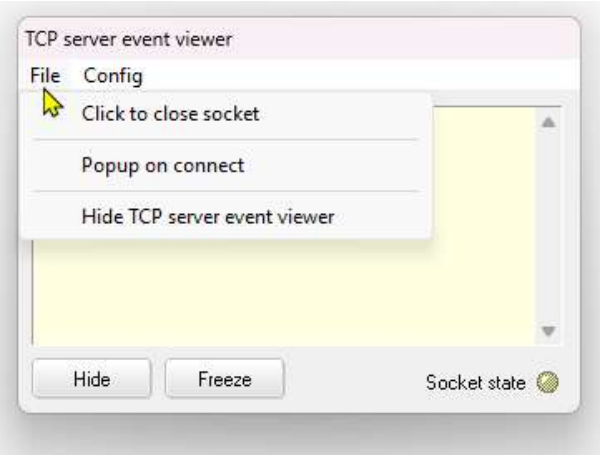
  Notice there is no top right corner X to close the event viewer … but the <**Hide**> button tucks it away out of sight.



Copyright © 2026 Robert C. Furzer K4CY

On the TCP server event viewer's <**File**> menu ▶

- **Click to Open|Close socket**: opens or closes the TCP port (it's a toggle).
- **Popup on connect**: the TCP server event viewer can be set to reappear magically whenever a TCP connection is established, or remain discreetly hidden.
- **Hide TCP server event viewer**: hides the window, the same as clicking the <**Hide**> button.

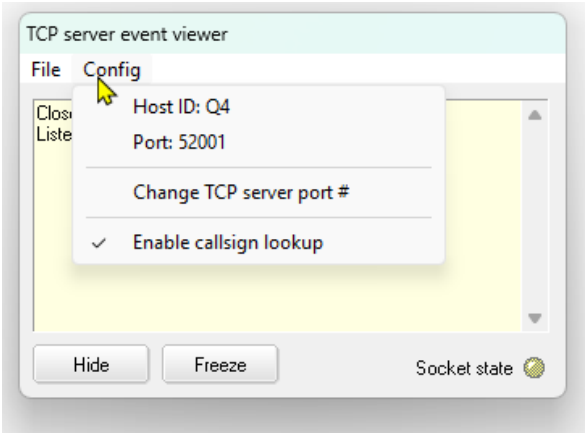


> The socket state "LED" at the bottom shows the TCP connection status in traffic light colors:
>
> - **Red**: the port is closed.
> - **Amber**: the server port is open, listening for connection requests, ready to go … but no TCP connection has been established at this point.
> - **Green**: a TCP connection is established and is passing traffic.

Once opened, Logger32's TCP server port (socket) remains open, patiently waiting for network connections until the port is closed, even if the TCP server event viewer is hidden (well almost: much like the operator, the TCP function is temporarily inactive while Logger32 is busy making backups or rebuilding the logbook. It needs to concentrate on the task at hand. Chillax.)

On the TCP server event viewer's <**Config**> menu ▶

- The top 2 lines show the "host ID " (computer name) and port number.
- <**Change TCP server port #**> lets us pick a different port for TCP in the unlikely event you need to do so *e.g*. to avoid a port conflict with some other networking application running on the same PC. Likewise, you *can* change the Host ID and Local IP through Windows if you want but that is out of scope for this manual.

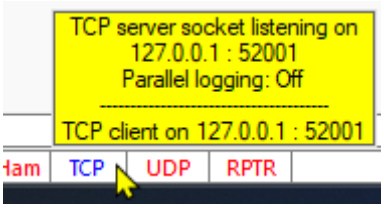


- With <**Enable callsign lookup**> ticked, any additional information (such as the DX operator's name, gridsquare, state and county) that can be retrieved by your preferred callsign lookup service (*e.g*. HamQTH or QRZ) is added to QSOs as they arrive through TCP and can be logged.

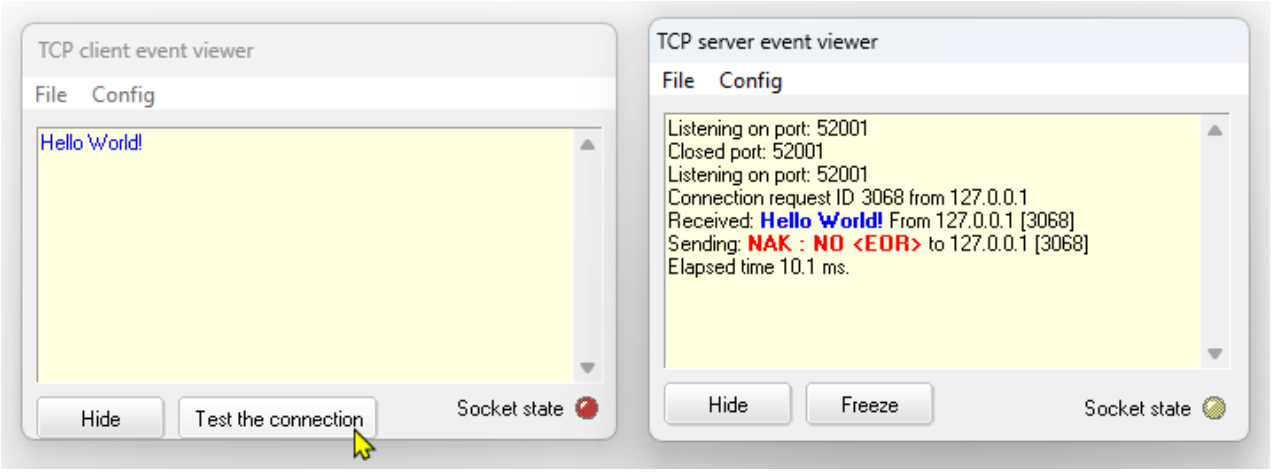
> **Hinson tip**: with several networked computers and applications in the shack, it is worth adopting a sensible naming and numbering approach. I prefer sticky labels for each machine showing the pertinent details (network names and addresses plus motherboard and CPU type, installed memory size, ports *etc*.) as it is quicker to glance at the labels than to look up the information in whichever operating systems they are running, if indeed they are running at all.
>
> **Hinson tip**: static IP addresses (such as the IPv4 address 192.168.1.73) avoid the confusion and miscommunication caused when DHCP suddenly gives your shack devices different addresses for some obscure reason. Remember, though, that each device requires a unique address, just like our callsigns.

Copyright © 2026 Robert C. Furzer K4CY

Returning now to the <**TCP**> panel's right-click menu:



- <**Click to Open socket**> opens the TCP server port and leaves it open, patiently waiting for an inbound connection. The <**TCP**> panel text turns blue showing that the port is open, and the mouse-over pop-up shows updated status info including the IP address and port numbers for the TCP client and server ▶
- <**Show TCP client event viewer**> opens a window showing - guess what - messages processed by the TCP client. With *both* event viewers open side-by-side, we can watch both ends of the TCP 'QSOs' ▼



Clicking <**Test the connection**> ▲ demonstrates that TCP is working:

- A three-way handshake establishes a client-server link using the configured port;
- The TCP server encodes and transmits an iconic message through the link;
- The TCP client receives, decodes and displays the message;
- The link closes a few milliseconds later, its job done.

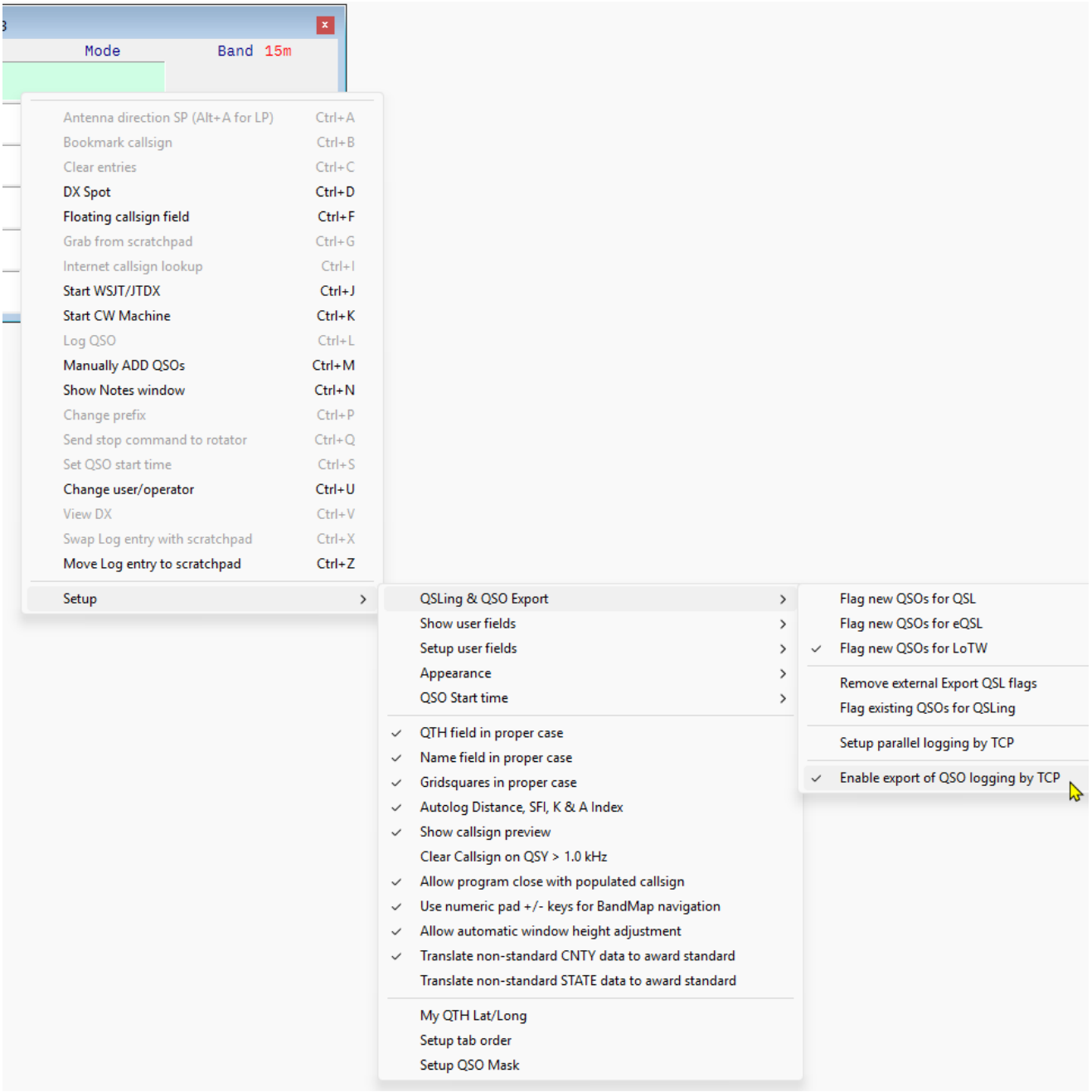
Notice that the TCP client socket ends up closed (red "LED") but the server continues listening for the next connection (amber "LED").

Copyright © 2026 Robert C. Furzer K4CY

32.3 Sending ADI-formatted QSOs to other apps via TCP

The TCP client can connect and send ADI-formatted QSO records to other applications via TCP as QSOs are logged in Logger32[401]. To use this facility:

1. Right-click any data-entry field in the log entry pane.
2. Click **Setup ⇨ QSLing & QSO Export** then click to tick **Enable export of QSO logging by TCP**.

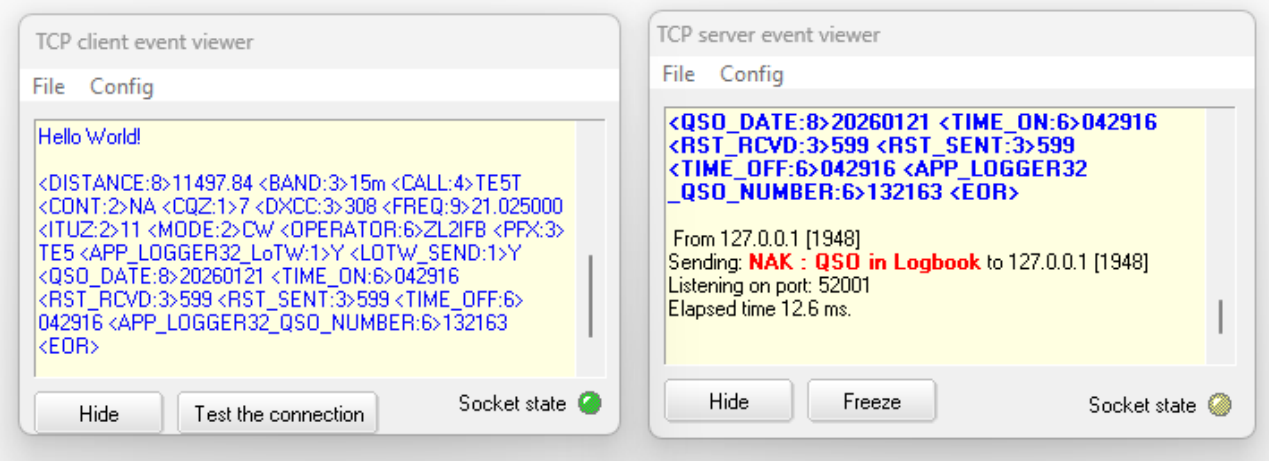


3. Right-click the TCP panel on the status bar, then click **Show TCP client event viewer.**
4. On the TCP client event viewer's **Config** menu, set up the IP address and port number for the server that is to receive the QSO data.

[401] Only newly-logged QSOs are sent – not those previous logged, not QSO records that are changed or deleted from the log, just new ones. This is *different* to parallel logging.

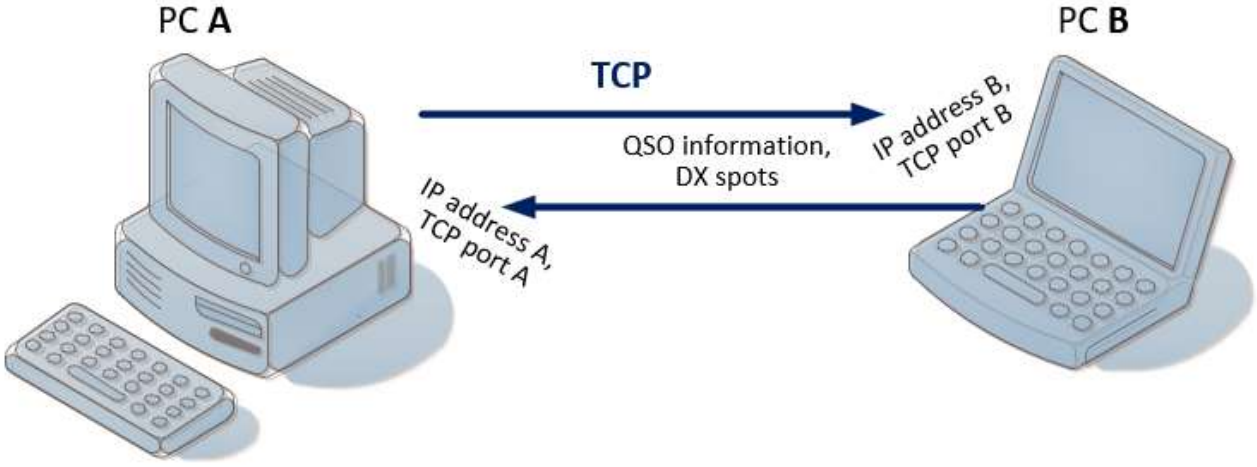
Copyright © 2026 Robert C. Furzer K4CY

5. Configure the receiving server to accept and process the QSO data as you wish, using the same port number as in step 4 and the IP address of the PC running Logger32.
6. Click the **Test the connection** button on the TCP client event viewer to send a **Hello World!** message … or log a QSO.
7. If you don't presently have a TCP server system and app to test with, you can configure Logger32's TCP server to receive the QSO messages and display them as received on the TCP server event viewer ▼ … but you MUST first turn off parallel logging for this test (if you are not sure, mouse-over the status line's TCP panel to check that it is off).



32.4 Parallel logging via TCP

Logger32's parallel logging function establishes a stable network connection between a pair of networked Windows PCs, both running Logger32, passing QSO and/or DX spot information between them either mono- or bi-directionally as ADI-formatted[402] TCP messages.



[402] ADIF compliance has implications for any non-ADIF compliant information, such as the German DOKs (secondary administrative subdivisions): the ADIF standard specifies secondaries for a few countries but *not* Germany, hence they are *not* sent to the parallel computer by default. A workaround involves stopping Logger32 on the sending PC (both of them if you want to enable bidirectional log updates and edits), using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad to edit the file *C:\Logger32\ADIFCountriesWithSecondarySubdivisions.txt* and inserting the line *230 Fed. Rep. of Germany* between *170 New Zealand* and *288 Ukraine*, then re-starting Logger32. Countries with primary admin subdivisions are similarly defined in *C:\Logger32\ADIFCountriesWithPrimarySubdivisions.txt*

Copyright © 2026 Robert C. Furzer K4CY

QSOs logged, amended or deleted on one PC trigger the sending of TCP messages through the network for those same QSOs to be logged/amended/deleted on the other PC as well.

> **Hinson tip**: if so configured and enabled, parallel logging works both ways. So if you set it up on the shack PC and another to check it out … and promptly forget about it, then later on open Logger32 on the other PC and start idly meddling with the log, perhaps 'correcting' or deleting QSOs, don't be surprised to find those same meddles have silently propagated to the main shack PC as well. If you wreck your log on either PC, the other may *also* be wrecked in parallel. Just sayin'. Love your backups.

32.4.1 Use cases for parallel logging

Why might you want to pair-up and duplicate logs and DX spots across two Logger32 systems? What would that achieve? Possible scenarios or purposes include:

- Contemporaneous backup of every QSO logged on a second PC *e.g.* in case the shack PC hard drive fails.
- Replicate the log across a desktop on the shack desk and a networked laptop used when you are out and about portable or mobile, logging new QSOs on either PC.
- Ditto for 2-shack homes *e.g.* garden shed/workshop and bedroom/spare room shacks.
- Ditto for someone usually DXing from home but occasionally or regularly from a radio club, a friend's shack, a holiday home, at work during the lunchbreak *etc*.
- An elmer shadowing and encouraging a novice *e.g.* celebrating when new ones are logged.
- Multi-op contesting or pileupping with a two-station setup[403].
- A QSL manager maintaining a contemporaneous copy of her DX station's log.
- An experienced Logger32 user remotely diagnosing someone's log entry issues, maybe?

Conversely, possible reasons for *not* using parallel logging include:

- The need for a second networked Windows PC running Logger32.
- Additional technical complexity, with the associated risks *e.g.* software design flaws and bugs.
- Performance and capacity issues – mostly negligible in practice (just another minor load on the systems and network).
- Automatic post-logging backups are disabled on paired systems (a design constraint).
- Network/cyber security risks, particularly if the PCs communicate over (and hence have network ports open onto) the Internet. You are increasing your 'attack surface'.
- Practical issues and annoyances *e.g.* if either or both PCs are using DHCP for dynamic rather than fixed IP addresses, meaning the port configurations may need to be updated each time either PC reboots or its IP address lease expires so a new IP address is issued.

[403] To be honest, rabid full-on contesters and DXpeditioners can achieve faster QSO rates using a dedicated contest logger such as N1MM+ *but* for casual contesters and keen DXers, Logger32 has advantages for DXing *while* contesting, a bit of both *e.g.* we can keep an eye on the WARC bands for interesting DX spots.

Copyright © 2026 Robert C. Furzer K4CY

32.4.2 Configure parallel logging

1. First get Logger32 set up and running separately on both PCs.

   The two installations need not be identical (*e.g.* the hardware, screen layouts *etc*. can differ) but be aware that:

   - Both PCs *must* run a version of Logger32 that supports parallel logging – meaning version 4.0.275 or later. Ideally, they should both use the same, current version to avoid possible incompatibilities, implying they should preferably both be auto-updating.
   - Both PCs *must* be networked, with a viable and reliable TCP/IP route between them *e.g.* on the same shack or home LAN. If they are on different LANs *e.g.* in local and remote shacks connected via an Internet VPN, TCP traffic must be able to pass freely between them, without being blocked by firewall rules or other issues (including policies on remote access and performance/capacity constraints) along the path.
   - The logbooks open in Logger32 on both PCs must have the same name[404] and operator. To do this, run an ADIF log export on the main PC, copy the ADIF file to the other PC via the network or USB stick, and import it into a new logbook with the same name and op.
   - When parallel logging is enabled (step 7 below), the 'automatic backup after N QSOs' function will be disabled. It seems pointless since, in effect, every QSO is automatically backed up to the parallel PC as it is logged.

2. Decide whether you want to pass QSO information in one direction only (*e.g*. *from* the shack PC *to* an 'informational' display-only PC at the pool bar), in both directions (*e.g*. between two shack PCs, one for your HF station and the other for the VHF/UHF station), or not at all (*e.g*. if you only want to use the TCP connection to send DX spots between the PCs).

   Hinson tip: with *bi*directional parallel logging, QSOs logged or changed on either PC will *also* be logged or changed on the other, in the open log, under the current operator's callsign. If the local PC is in your shack and the remote PC is your QSL manager's, any QSOs made and logged on that PC by the QSL manager under her callsign may *also* be logged on your shack PC. You probably don't want to do that! Be careful.

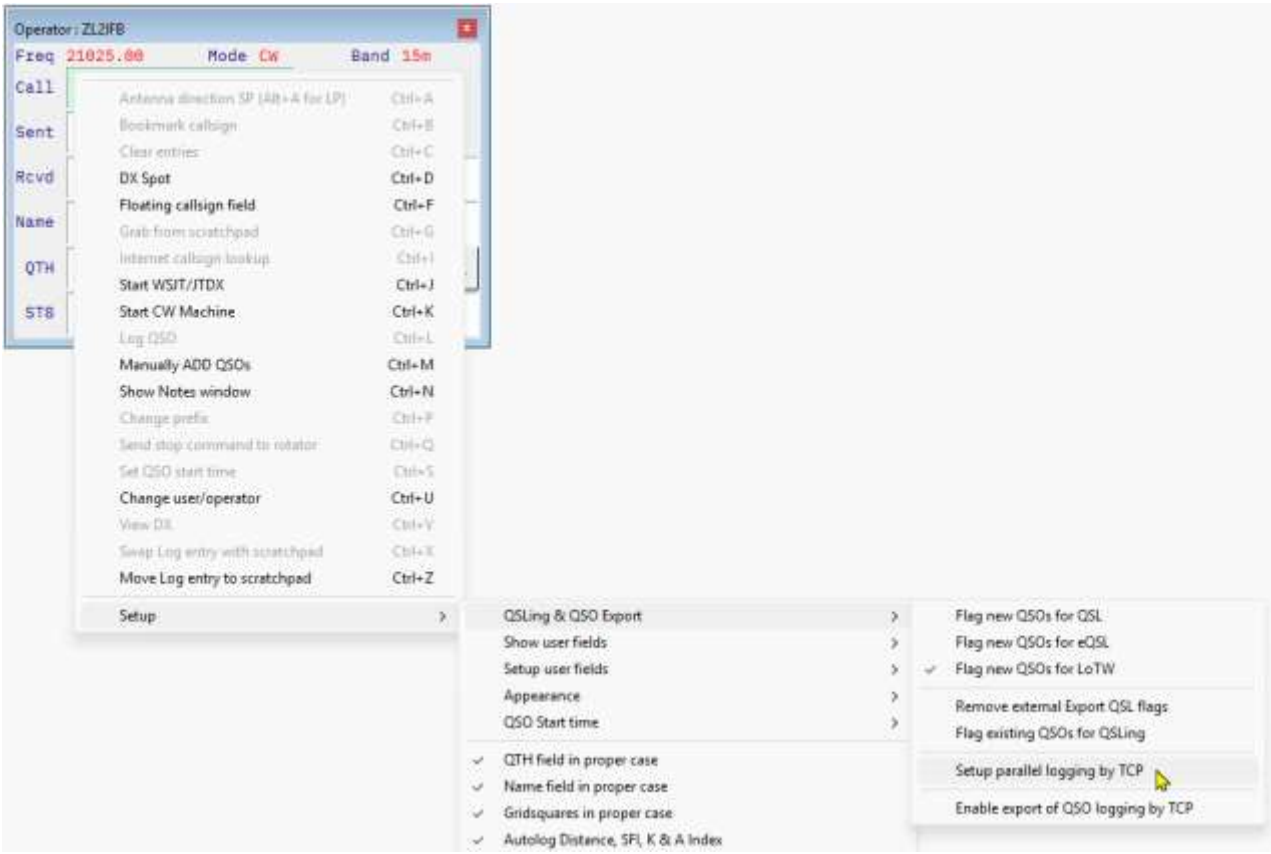
3. Find out and make a note of the IP address of each PC. It helps if they both have static IP addresses, rather than being allocated addresses dynamically by your router using DHCP, or reassigned randomly by your ISP.

   Hinson tip: if you have the choice, 4-part dotted IPv4 addresses are easier to recall and type than those 6-part IPv6 ones with five colons. I only have one colon: it's enough for me.

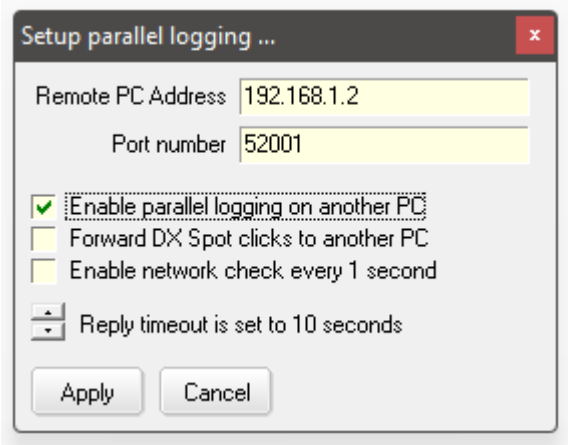
[404] When parallel logging is enabled in step 7 below, Logger32 will check that the local and remote logs have the same name, or present a warning message.

Copyright © 2026 Robert C. Furzer K4CY

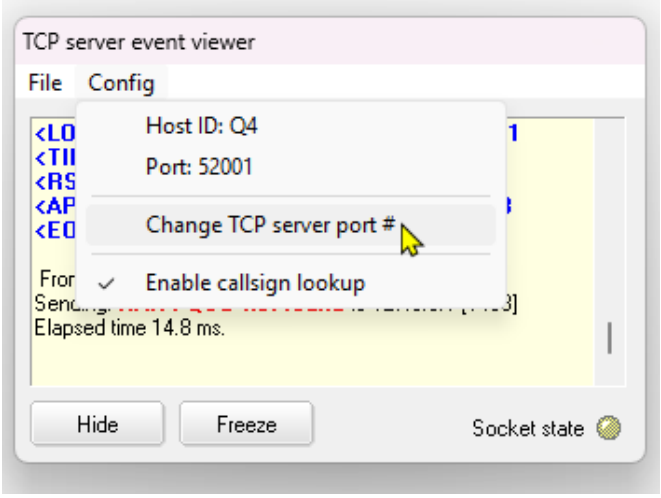
4. On each PC, right-click any data entry field in the log entry pane, then click **Setup ⇨ QSLing & QSO Export ⇨ Setup parallel logging by TCP ▼**



5. On each PC, configure the IP address and TCP port number for the *other* PC ▶



For example, enter the IP address B and port number B for laptop B shown in the diagram above into desktop A, and vice versa. When you click <**Apply**>, this PC will start trying to connect to the remote (other) PC via TCP using its IP address and port.



◀ Should you need to change ports on either PC, the TCP server event viewer <**Config**> menu has the option.

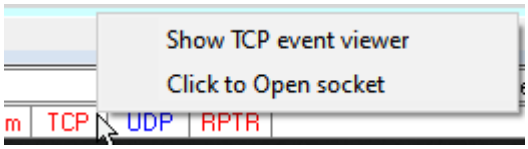
The default port number 52001 will probably work at both ends unless other software is already using it – in which case simply choose another number. Aside from being an

Copyright © 2026 Robert C. Furzer K4CY

available decimal value in the range 1 to 65535, there is nothing special about the port number: you may prefer something memorable and associated with amateur radio, such as 7388. Or not.

> **Hinson tip**: for bidirectional comms, Logger32 at each end of the parallel logging connection needs to be configured with the TCP port number for the *other* end, so it knows where to send the info. Each PC has its own unique IP address so technically they can use different port numbers giving a different TCP socket … but it's a little less confusing if they both use the same port numbers for parallel logging.

6. Check that the TCP ports are open on both PCs and accessible from the other:
   - On the TCP Server window, the <**Socket state**> "LED" should be amber (listening, waiting for a connection) or green (connected and transferring data).
   - The <**TCP**> panel on Logger32's status bar should have blue text on both PCs.
     - If TCP is red on either PC, right-click the <**TCP**> panel then click <**Click to Open socket**> on that PC ▶



   - On either or both PCs, under **Setup ⇨ QSLing & QSO Export ⇨ Setup parallel logging by TCP/IP**, click to tick <**Enable network check every 1 second**> to have that PC continually check that the TCP connection to the other PC is available.
     - If the TCP connection isn't available, a popup warning message appears ▶
     - Click the message to pause checking … until Logger32 is next re-launched when it starts checking automatically.



     - To stop the warning message popping up, either get the TCP connection working reliably … or un-tick <**Enable network check every 1 second**> to stop checking.
     - <**Reply timeout is set to** [**10**] **seconds**> says how long Logger32 on this machine will wait for a TCP reply from the remote PC after it has connected to send QSO information across the network. Click the little up or down arrows to increase or decrease the waiting time, if you like. The default 10 seconds is generally OK.

> **Hinson tip**: despite using TCP as opposed to connectionless UDP, if the network connection is unreliable (*e.g.* weak Wi-Fi), **QSOs may be lost** in transit. Any QSOs logged on either system while the parallel logging connection is down, or while parallel logging is disabled or not running on either PC (*e.g.* while the system reboots), simply **disappear into the ether and do not turn up automatically on the other PC**. They are ***not*** buffered and automatically re-sent if and when the connection reopens. Therefore, the two logs can quite easily get out of step … in which case you will need to merge the two partial logs, importing them into one complete log.

7. Using **Setup ⇨ QSLing & QSO Export ⇨ Setup parallel logging by TCP/IP** once more, click to tick <**Enable parallel logging to another PC**> and then <**Apply**> on at least one of the PCs, to have it send QSOs to the other PC.
   - If *both* PCs have this option set, parallel logging will work bidirectionally.
   - If only *one* PC has this option set, parallel logging will work unidirectionally **from** that PC **to** the other.

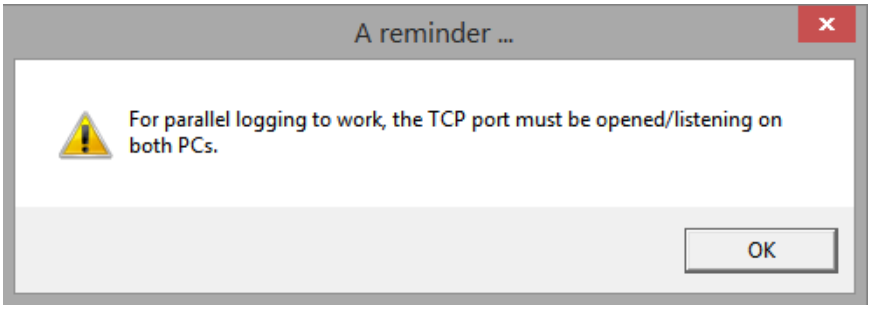
Copyright © 2026 Robert C. Furzer K4CY

- If *neither* PC has this option set, neither PC will send QSO information to the other, so parallel logging is effectively disabled.

8. Also under **Setup ⇨ QSLing & QSO Export ⇨ Setup parallel logging by TCP/IP** is the option to <**Forward DX Spot click to another PC**>. Tick this to pass the stream of filtered and de-duped DX spots from the present PC to the other PC, without the other PC having to do all the filtering and de-duping intense processing necessary if it simply received the raw stream of DX spots directly from the connected DX Cluster/s, Reverse Beacon Node/s *etc.*
9. After clicking <**Apply**>, read, consider and <**OK**> the reminder ▼

You *do* have *both* PCs configured correctly, right?

If not, go back and do so or it won't work.



10. Log a fake or real QSO on the sending PC and squeal with geeky delight as it is also logged on the other PC moments later.

    Amend or delete the same QSO and squeal again as it is magically amended or deleted at the far end.

    If you have ticked <**Forward DX Spot clicks to another PC**>, try clicking a DX spot on either PC to confirm that the radio attached by CAT to the other PC duly QSYs.

    Congratulations! Parallel logging is working! Stop squealing now, please.

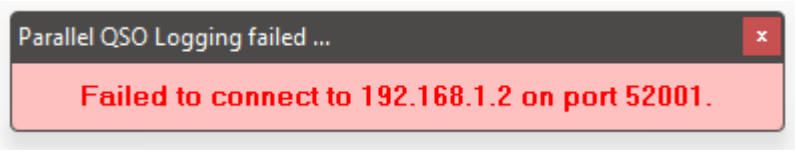
Hinson tip: opening the TCP connection, sending a QSO record to the other PC, receiving the message, and checking inserting a new/changed QSO into the log or deleting an existing QSO, takes a while to complete … which makes it quite slow (a couple of seconds per QSO). If you are making multiple log changes (*e.g.* importing an ADIF file), you will find it quicker to disable parallel logging, apply the changes to each log separately, then re-enable parallel logging – see the next Hinson tip for instructions. Parallel logging should be fast enough to keep up with new QSOs being logged in real time, even in a contest or pileup.

I run Logger32 on two PCs with parallel logging. One PC is configured for JTDX, the other for DX spot collection and processing – several RBNs feeding ten BandMaps set to show only highlighted DX spots. This configuration adds no unnecessary CPU cycles to the JTDX PC. If I see an interesting DX spot on the Telnet/RBN PC, all I have to do is click the spot to change band/mode on the JTDX PC.

Bob K4CY

Copyright © 2026 Robert C. Furzer K4CY

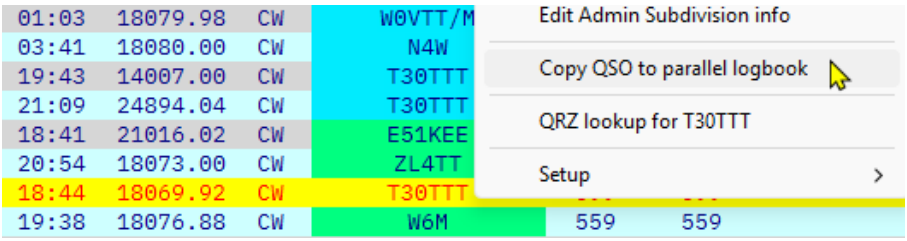
32.4.3 Parallel logging errors



With parallel logging enabled, Logger32 copes easily with minor glitches such as network congestion thanks to the TCP protocol and integrity controls. You probably won't even notice anything amiss. More significant issues, however, can result in logged QSOs not being passed across and appearing on the parallel system – issues such as the second PC not running, not having its TCP port open, and so not receiving QSOs sent from the first.

Hinson tip: if you discover the parallel system is missing *loads* of QSOs from its log, it is probably quicker to have Logger32 generate a log export in ADIF format from the originating PC, then transfer over and import the ADIF into Logger32 on the receiving PC. It's up to you whether to do partial or whole log exports. Run the export-then-import thing in both directions to be sure both logs are complete … and hang on to those ADIF files as backups in case of disaster.

If you right-click a QSO in your logbook, you will find an option to copy it to the parallel logbook ▶ … provided you are parallel logging anyway!



32.5 TCP configuration for JTDX or other digimode software

Digimode QSOs made and logged in JTDX can be passed through a TCP connection to Logger32 to be recorded in the open logbook as if they had been typed into the log entry pane.

There is an important proviso relating to the QSO times, due to time being a key field in the logbook database. The time of each QSO *must* be unique. Therefore …

> If you are operating as an UberFox, using MSHV with parallel streams, or using any digimode that can log multiple contacts per second, use the logging by UDP option, ***not*** the TCP option.

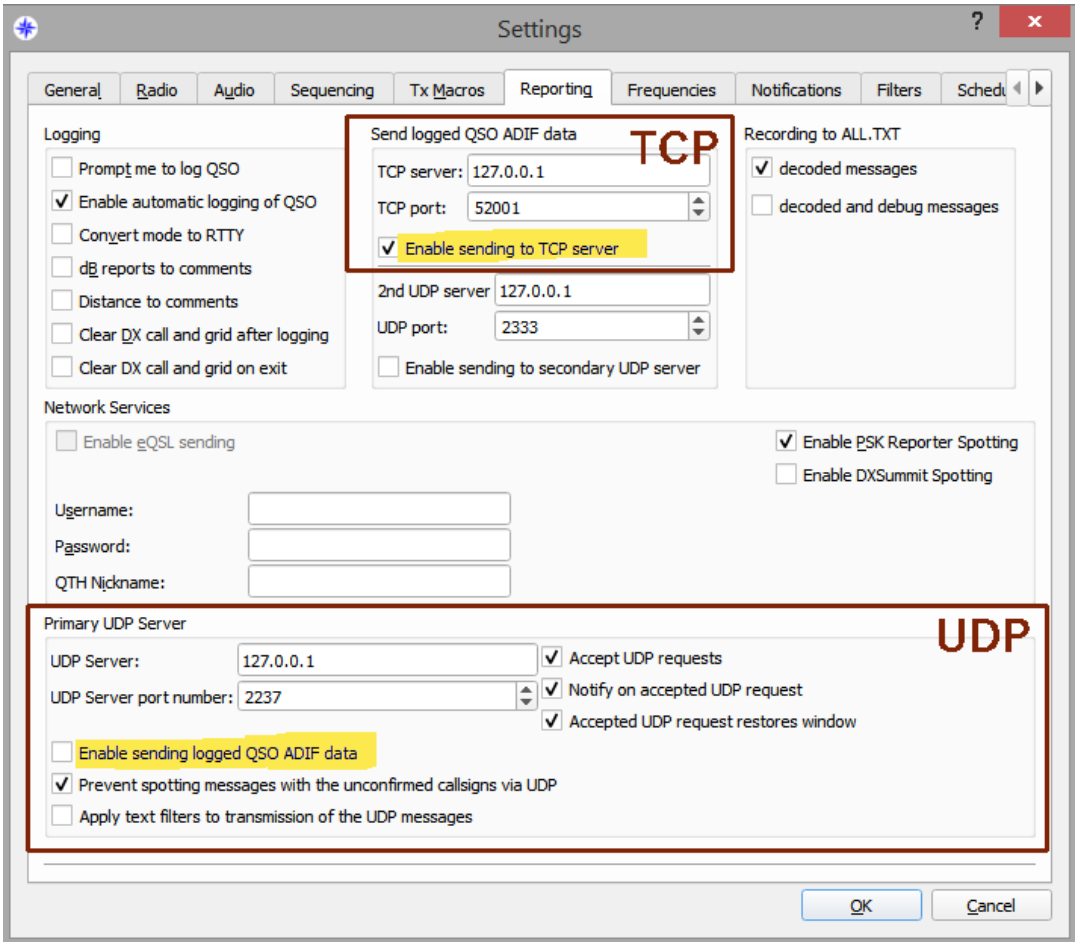
Incoming UDP logging request messages are queued by Logger32 and processed serially, one QSO at a time. Before being entered into the logbook, the time of each new QSO is checked for uniqueness. If there is an existing QSO in the logbook with the same time (down to the same second), the time of the new QSO is artificially incremented by one second and the time check is repeated up to 10 times until an empty time slot is found, allowing the QSO to be logged.

If you have hitherto been using **UDP** to link JTDX with Logger32 (*e.g.* using the UDP BandMap), sever that network connection first[405]. You can simply close the UDP port in Logger32, or in JTDX, or change the UDP port numbers so the two apps are no longer communicating via UDP.

[405] The **UDP** BandMap will close … because you are not using **UDP**. Therefore you can no longer use the manual or automated cherry-picker, or the JTDX control panel, which all depend on the **UDP** BandMap.

Copyright © 2026 Robert C. Furzer K4CY

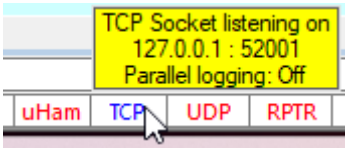
Now open the <**Reporting**> tab in JTDX settings and look at the **TCP** section ▼



If JTDX is running on the same PC as Logger32, the default TCP Server IP address (127.0.0.1) and port (52001) should be fine for both apps. However, if JTDX and Logger32 are running on *different* computers, the TCP server IP address and port settings in JTDX must match the Logger32 PC's IP address and port.

Mouseover the TCP panel on the status bar to check the info – IP 127.0.0.1 and port 52001 for *this* PC ▶

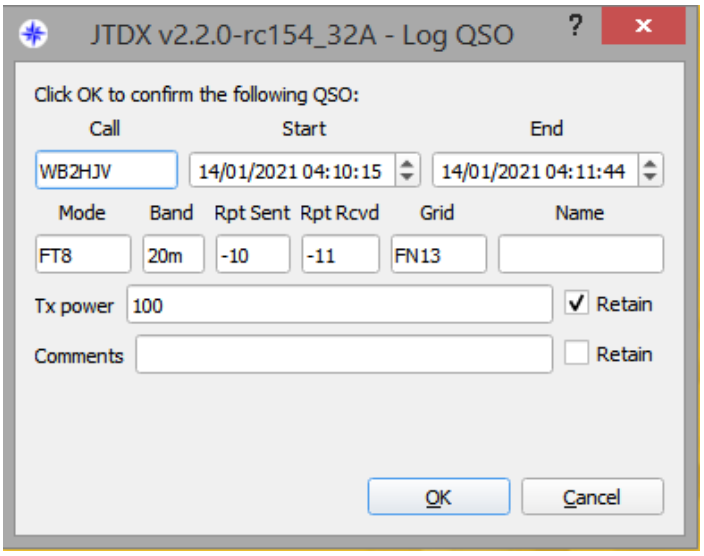
Your IP address and port may well be different.



Still in JTDX **Settings** ➪ **Reporting** tab, click to tick <**Enable sending to TCP server**> in the TCP section, telling JTDX to send QSO data out as ADI-formatted TCP messages as well as logging them locally in the JTDX log.

Hinson tip: if you are using TCP, ensure that <**Enable sending logged QSO ADIF data**> in the UDP section is *not* ticked as well … otherwise you may find that digimode QSOs are logged twice in Logger32. Pick *either* TCP *or* UDP for JTDX logging, *not not* both both.

Copyright © 2026 Robert C. Furzer K4CY

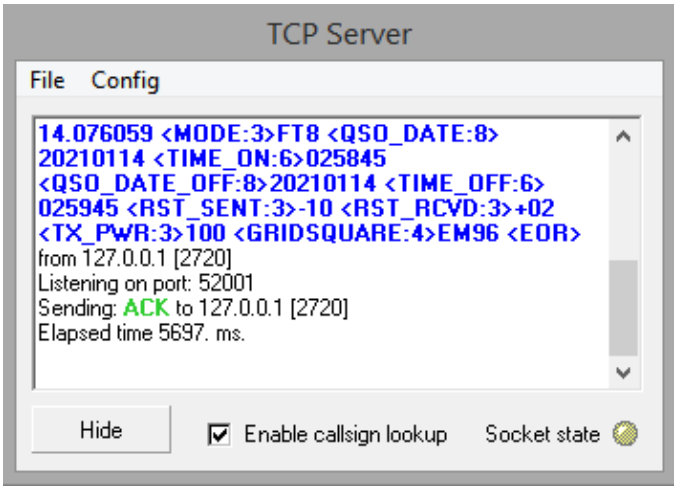


Also tick <**Enable automatic logging of QSO**> … unless for some reason you *want* ◀ the JTDX log entry pane to pop up after every QSO (maybe to add comments?), in which case tick <**Prompt me to log QSO**> in the **Logging** section of JTDX **Settings** ➪ **Reporting** tab.

Click <**OK**> to save the JTDX settings.

Now open the TCP port in Logger32. Make a digimode QSO in JTDX. When it is completed and logged in JTDX, the QSO is also sent via TCP to Logger32's port as an ADIF record, where it is received and logged automatically in the open logbook. Magic!

The TCP event viewer window pops up briefly (if so configured), showing the QSO traffic ▶



32.6 Using JTAlert with Logger32

JTAlert is a popular add-on for WSJT-X and JTDX users that looks-up decoded callsigns and generates alerts and highlighting for various wants/needs. It can, for instance, point out stations in US states or Canadian provinces that you have yet to work on the present band and digimode. It also has a real-time messaging function, allowing users to chat through the Internet while chasing new ones on-air.

There are two ways to add JTAlert to your setup …

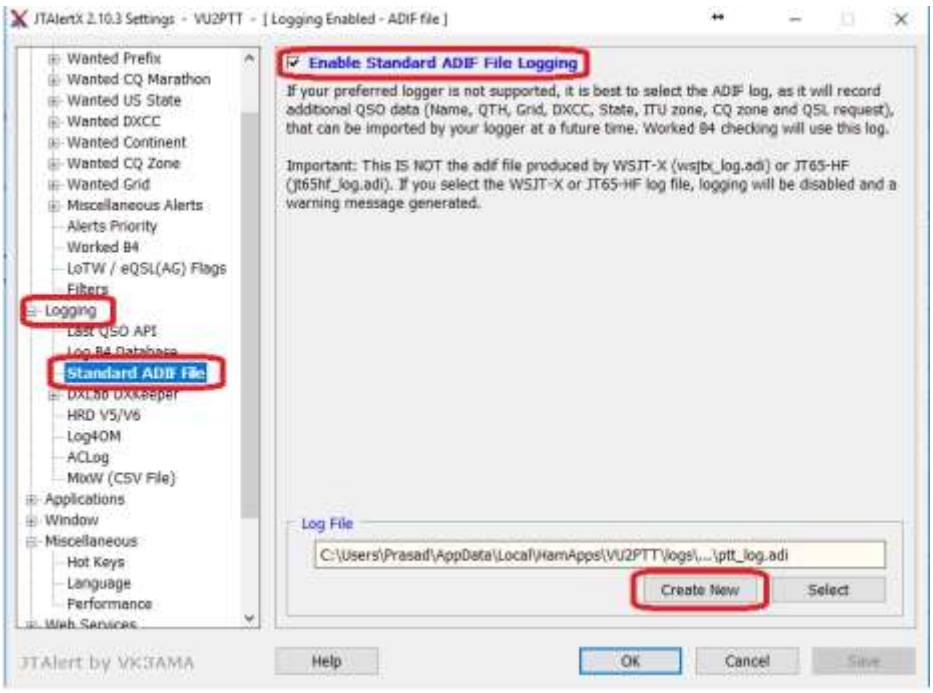
32.6.1 Method 1: TCP+UDP using socat

JTAlert can be linked to Logger32 and WSJT-X using "socat", a UDP-TCP conversion utility. Configuration instructions are available online courtesy of Prasad, VU2PTT. In short:

- Download and install socat.

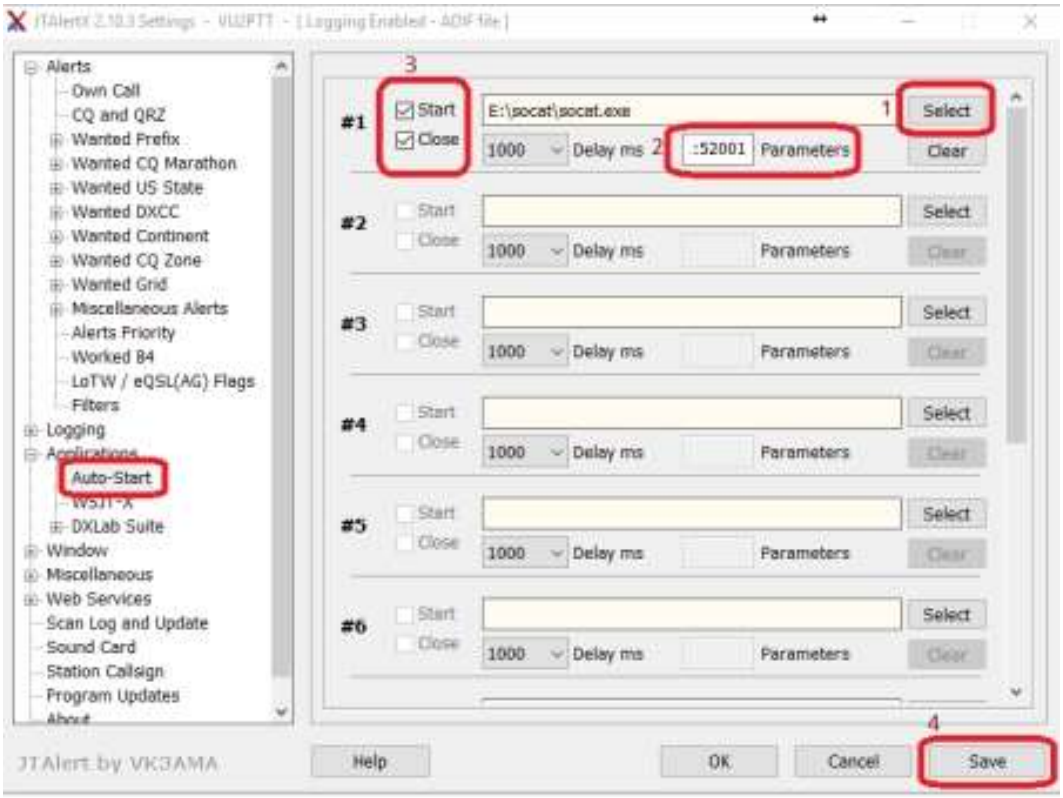
Copyright © 2026 Robert C. Furzer K4CY

- In the JTAlert *Settings* window, select *Logging* on the left, click *Standard ADIF File*, click <**Enable Standard ADIF File Logging**>, create a new log by clicking *Create New* and create a new ADIF log with a filename you wish to use ▼



Then click *OK*.

- In the JTAlert *Settings* window, click to open *Auto-Start* on the left menu ▼



On the right side of this window, configure JTAlert to use socat and automatically start and stop it whenever JTAlert is opened and closed.

Copyright © 2026 Robert C. Furzer K4CY

Click the *Select* button (#1 on the screenshot above) and use the browse window that pops up to go to the folder where you saved the socat utility. Select the file *socat.exe* and click the *Open* button.

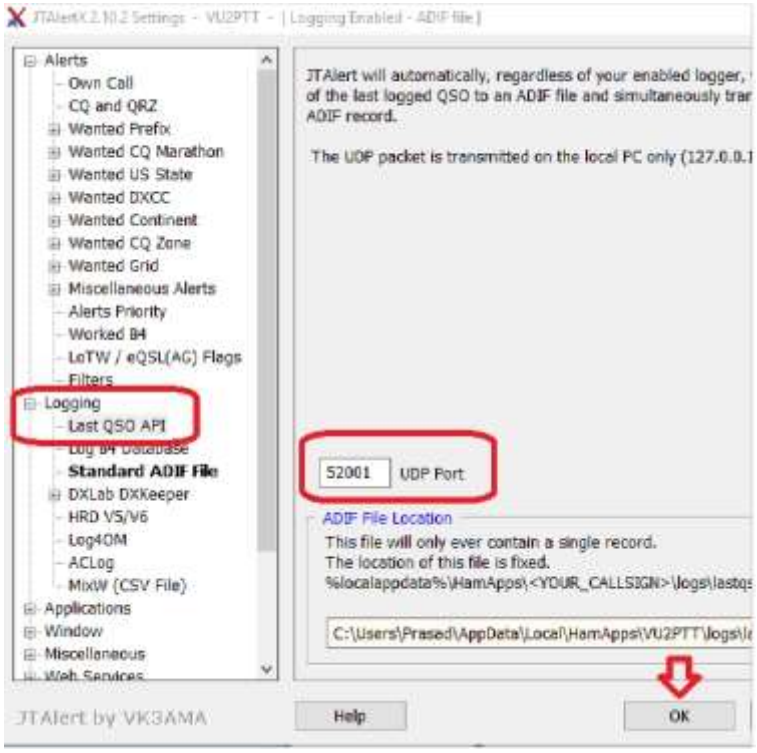
Next in the textbox marked *Parameters* (#2), copy and paste the following socat command parameters (copy these exactly: the spaces are as important as the text):

-d -v -T30 -ly UDP4-LISTEN:52001,fork,bind=127.0.0.1 TCP4:127.0.0.1:52001

Make sure the *Start and Close checkbox* (#3) on the left of these fields is selected.

Click *Save* (#4) at the bottom of the JTAlert window.

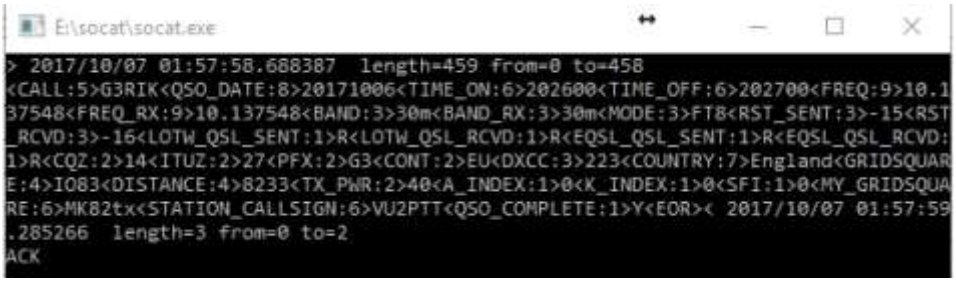
- In the *Logging* section of the JTAlert *Settings* window, select *Last QSO API* ▶
- Make sure the UDP Port is the same as Logger32's TCP port (52001 by default), then click *OK*.



Leaving WSJT-X and Logger32 running, shut down and restart JTAlert. If everything is configured correctly, the socat utility command window should pop up with a blank screen. Leave the socat window open in a corner of the screen or minimize it once everything is working fine.

Now make a QSO in WSJT-X and log the QSO. This is what happens:

The socat window shows the ADIF string sent out over the UDP port configured in JTAlert ▶



The Logger32 TCP Server window shows the incoming ADIF string received from socat.



◀ And your logbook shows the logged digimode QSO. Success!

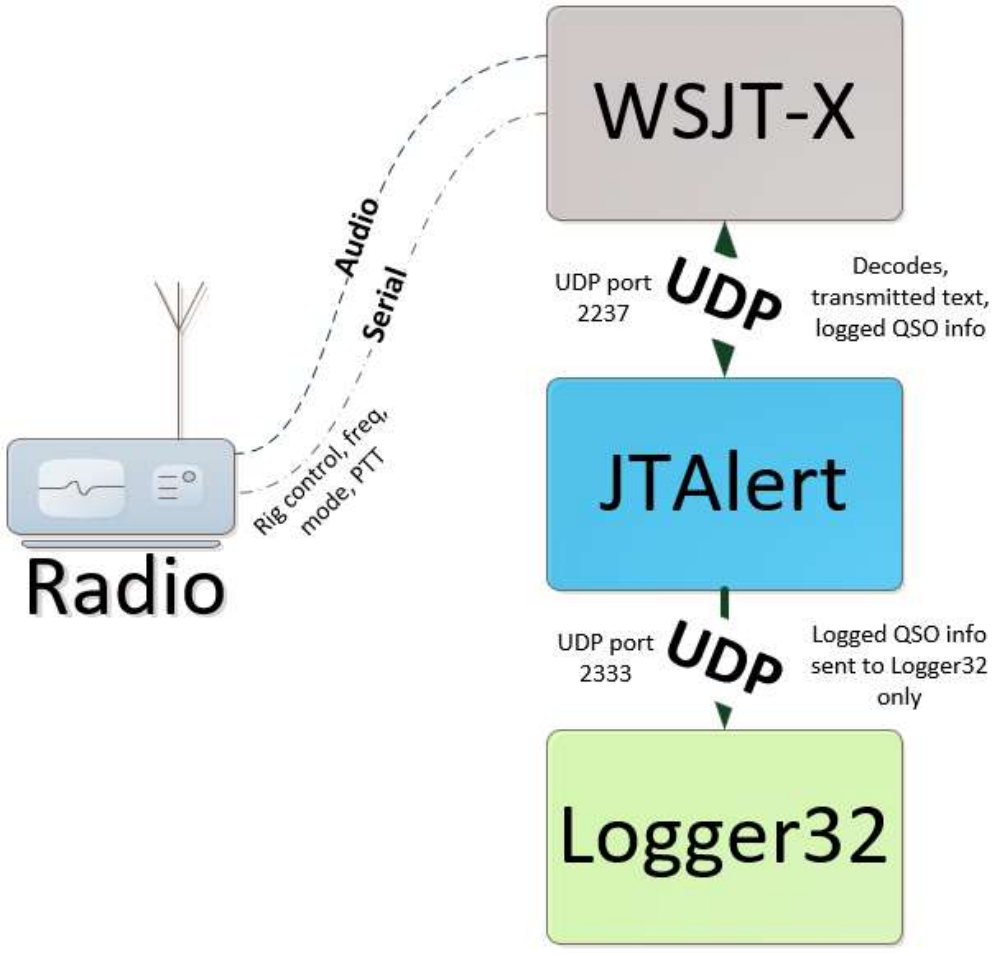
Copyright © 2026 Robert C. Furzer K4CY

- **Logging via TCP using JTAlert with WSJT-X** (a note from Ron W4LDE): if you have configured callsign lookups in JTAlert or Logger32 using a callbook server such as HamQTH, depending on the data provided by the service, Logger32 can log: ADDRESS, SFI, A-index, K-index, BAND and BAND_RX, CALL (the contacted station's callsign, naturally!), CONT (continent), COUNTY (secondary administrative subdivision), CQZ and ITUZ (CQ and ITU zones), DISTANCE (short path great circle distance in km), DXCC (entity from the DXCC list), FREQ and FREQ_RX, GRIDSQUARE, MODE, NAME, QSO_DATE, QTH, RST_RCVD, RST_SENT, TX_PWR, STATE, PFX (prefix) and more *e.g.*

<CALL:5>WT9WT<QSO_DATE:8>20171104<TIME_ON:6>190100
<TIME_OFF:6>190200<FREQ:6>10.136<FREQ_RX:6>10.136
<BAND:3>30m<BAND_RX:3>30m<MODE:3>FT8<RST_SENT:3>-03
<RST_RCVD:3>+04<GRID SQUARE:6>EM59DT<DISTANCE:3>829
<TX_PWR:2>50<A_INDEX:1>9<K_INDEX:1>1<SFI:2>73
<NAME:7>William<QTH:11>Springfield<STATE:2>IL<CQZ:1>4
<ITUZ:1>8 <CONT:2>NA<CNTY:11>IL,Sangamon
<ADDRESS:67>William W Tinsley, 114 Calvin, Springfield, IL 62704, United
States<DXCC:3>291<COUNTRY:13>United States
<MY_GRIDSQUARE:6>EM73ol<MY_CQ_ZONE:1>5<MY_ITU_ZONE:1>8
<STATION_CALLSIGN:5>W4LDE<QSO_COMPLETE:1>Y<EOR>

32.6.2 Method 2: UDP only (simpler but limited)

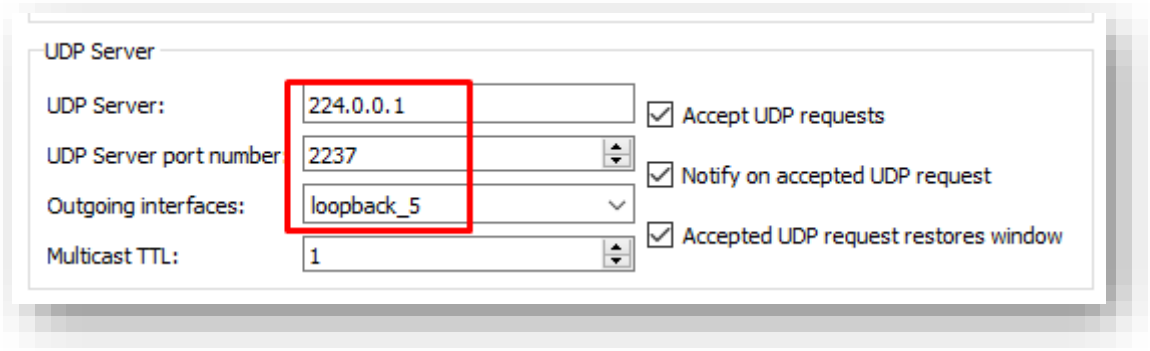
JTAlert can run *two* UDP ports to link Logger32 to WSJT-X as a middle-man, with benefits ▼



Copyright © 2026 Robert C. Furzer K4CY

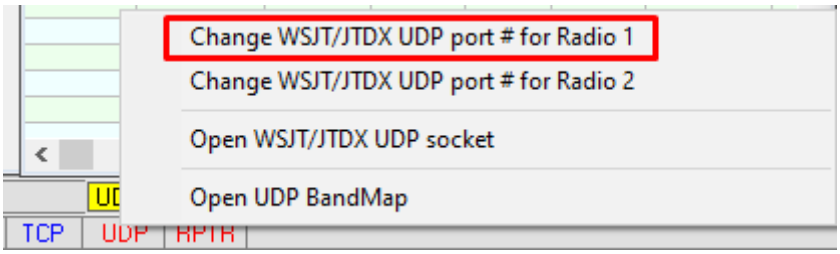
JTAlert receives data from WSJT-X via UDP port 2237 (the default in WSJT-X) and passes (some of) it on to Logger32 via UDP port 2333 (the default in JTAlert).[406]

1. In WSJT-X, open **File ⇨ Settings ⇨ Reporting**.
2. In the "UDP Server" section, choose an IP address in the multicast range, typically 224.0.0.1 onwards (as described in the JTAlert manual) and port 2237.
3. Select the available loopback interface under "Outgoing interfaces" ▼



The loopback number may be different on your system.

4. Right-click the UDP panel on the status bar to open the UDP control menu then click <**Change WSJT/JTDX port # for Radio 1**> ▼



5. Pick and enter a UDP port number that isn't presently in use *e.g.* 2333, then <**OK**>.
6. Either click <**Open WSJT/JTDX UDP socket**> on the UDP panel's right-click menu, or better still click <**Open UDP BandMap**>.

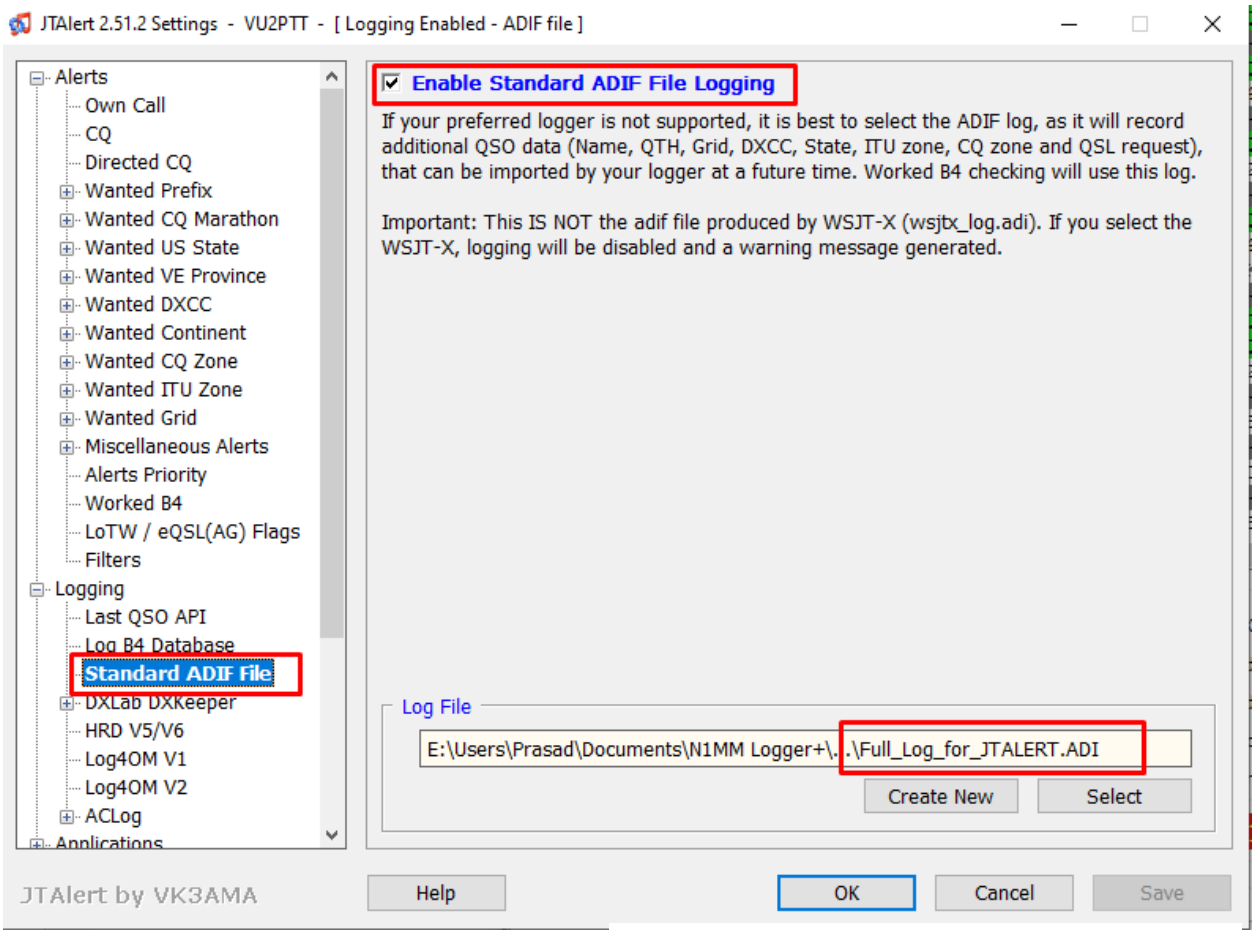
> **Hinson tip:** although the UDP BandMap shows decoded callsigns, highlighting any new ones as normal, the cherry-picker and *Calling me …* BandMap **do not work** with this configuration. If that's a disappointment, try method 1 using socat instead.

7. In order to tell JTAlert what countries, states *etc*. you have already worked, export your full log from Logger32 as an ADIF file. Remember the destination path (disk and folder) and file name for the step after next …

[406] I realise this is UDP stuff in the TCP chapter. So, go ahead, fire me.

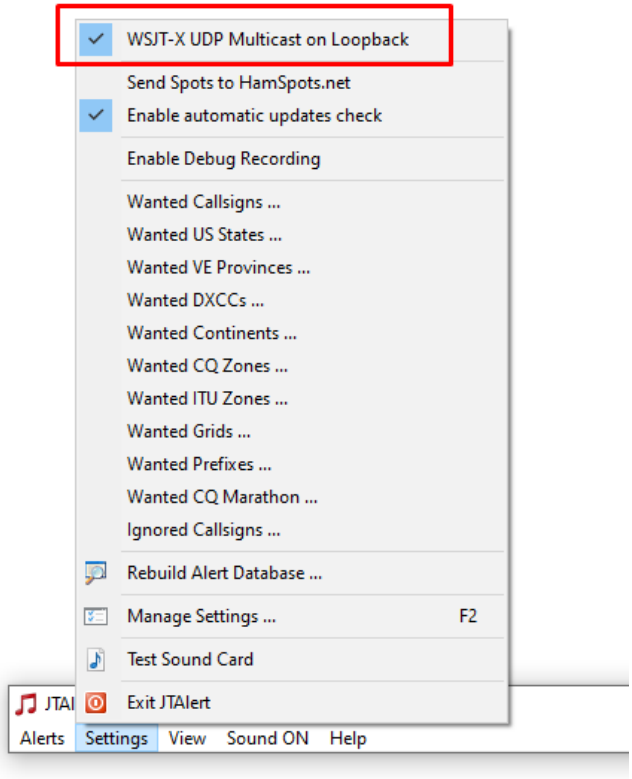
Copyright © 2026 Robert C. Furzer K4CY

8. In JTAlert, under **Settings ⇨ Logging ⇨ Standard ADIF File**, click to tick <**Enable Standard ADIF File Logging**> ▼



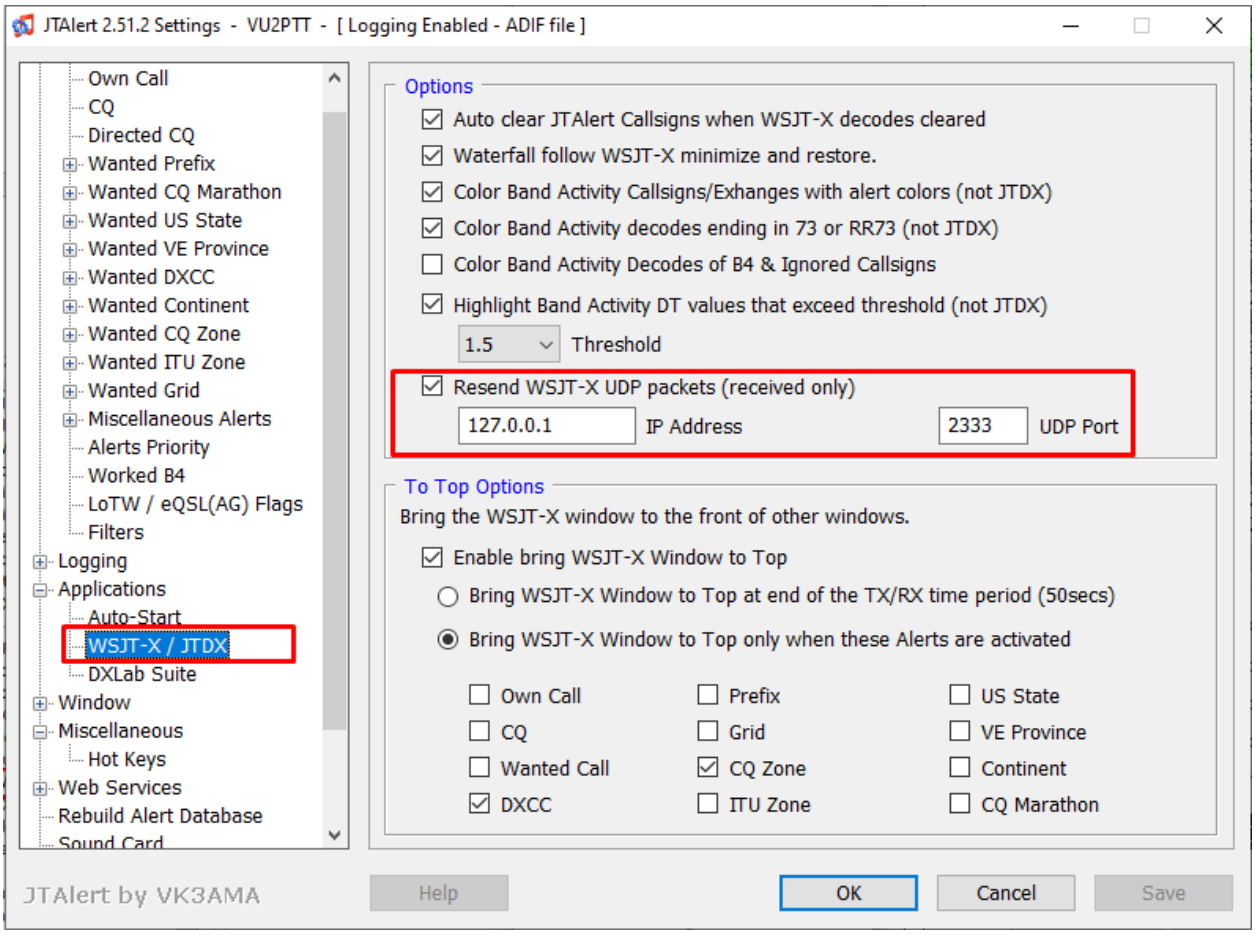
9. In the Log File field ▲, tell JTAlert the path and file name to the ADIF you just exported from Logger32.

10. Click **Settings** and click to tick <**WSJT-X UDP Multicast on Loopback**> at the top ▶



Copyright © 2026 Robert C. Furzer K4CY

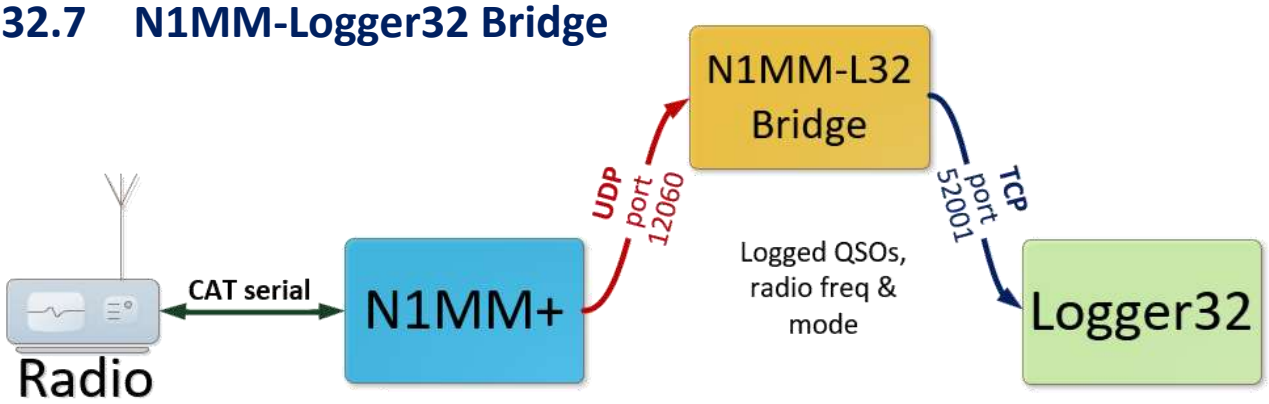
11. Configure **Settings ⇨ Applications ⇨ WSJT-X/JTDX** as shown below ▼.



IP Address 127.0.0.1 and port 2333 are the defaults.

12. Enjoy!

32.7 N1MM-Logger32 Bridge

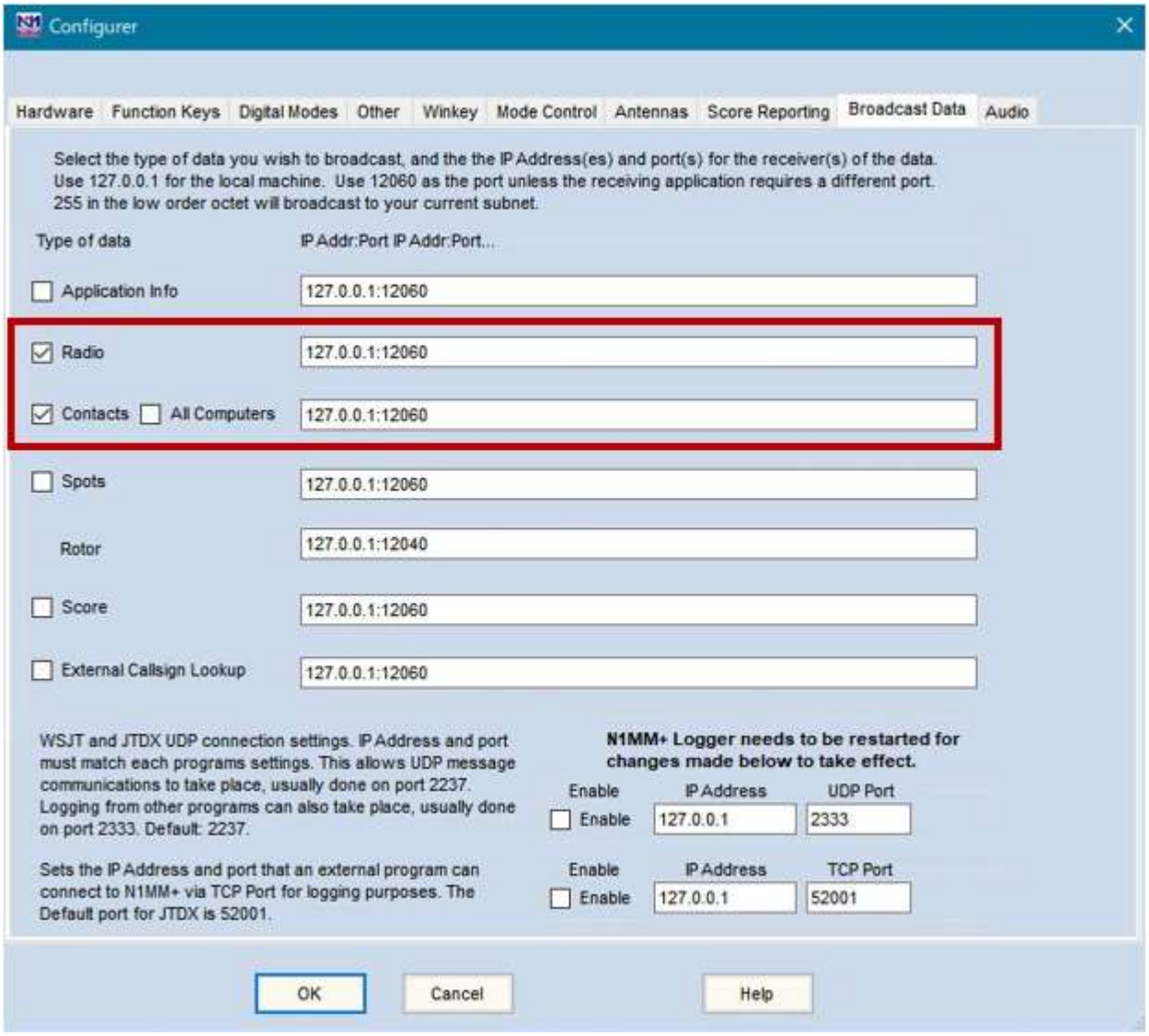


N1MM-Logger32 Bridge is N2AMG's optional add-on utility that grabs the QSO data broadcast via UDP by N1MM+ as contest QSOs are logged, translating it to TCP messages and sending it through to Logger32's TCP Server so the QSOs can be logged in your main station log too, at about the same time. This way, you won't need to remember to import your ADIF contest log into your consolidated log in Logger32 as well as submitting it to the adjudicator after the contest.

Hinson tip: provided your PC is powerful enough and your interest and attention levels high enough, Logger32 running in parallel with N1MM+ can alert you to 'new ones' that are spotted on DX cluster and shown on your BandMaps, even if they are of no value in the present contest.

Copyright © 2026 Robert C. Furzer K4CY

1. To set this up, N1MM+ needs to broadcast the QSO information via UDP. Run N1MM+, open **Config ⇨ Configure Ports, Mode, Control, Audio, Other**..., then open the **Broadcast Data** tab and click to tick the <**Contacts**> row ▼



2. If N1MM+ will have full CAT control of your radio during a contest, tick <**Radio**> as well to have N1MM+ pass the radio frequency and mode information through to Logger32[407].

   The default IP address and port should work fine but if you need to make changes for some reason, keep a note of the values you are using. The defaults are shown on the bridge diagram above. The port numbers should be the same at both ends of the TCP and UDP connections.

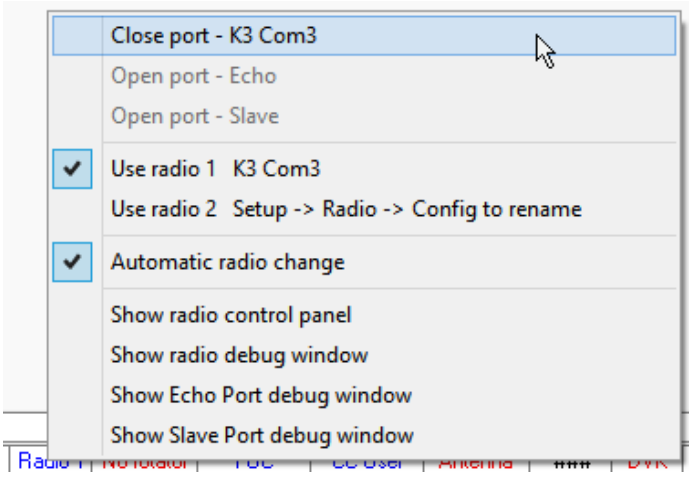
3. Now run Logger32.

[407] If N1MM+ controls the radio directly and exclusively, the radio should respond almost instantly *e.g.* when you click a DX spot on the N1MM+ band map to chase a multiplier, or click N1MM+'s band button to move someone to a new band. If radio control is shared with Logger32 or other software, delays and conflicts are possible while network messages are composed, sent, received and acted-upon.

Copyright © 2026 Robert C. Furzer K4CY

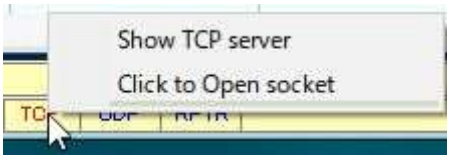
4. A CAT connection to the radio cannot be shared directly. If Logger32's CAT port is open but you want to give N1MM+ CAT control of the radio instead, close Logger32's CAT port first:

   Right-click the <**Radio 1|2**> panel in the status bar then click <**Close port – [**radio name**]**> ▶



5. Open Logger32's TCP server:

   Right-click the **TCP** panel in the status bar then click <**Click to Open socket**> ▶



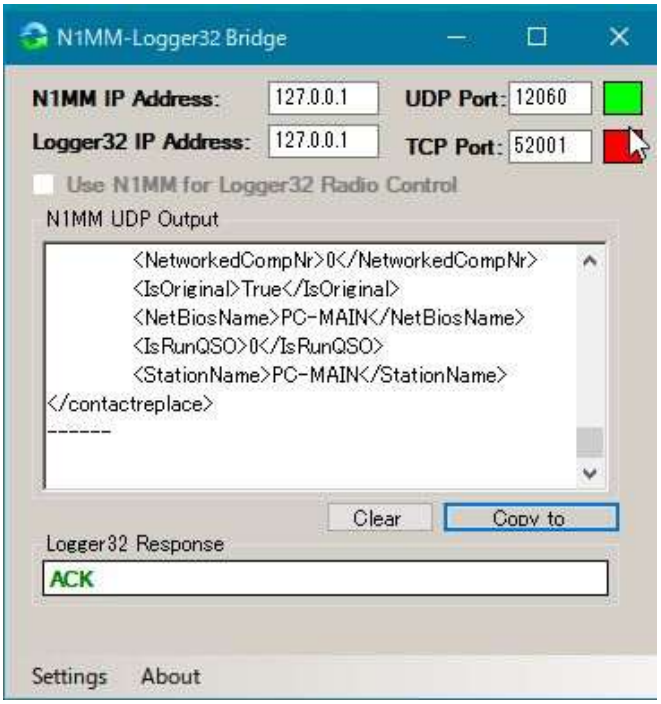


◀ When you mouseover the **TCP** panel on the status bar, a tooltip confirms that the TCP Server is running, and reminds you of the port settings.

6. Download and install the N1MM-Logger32 Bridge utility from N2AMG's website.

7. Start the N1MM-Logger32 Bridge.

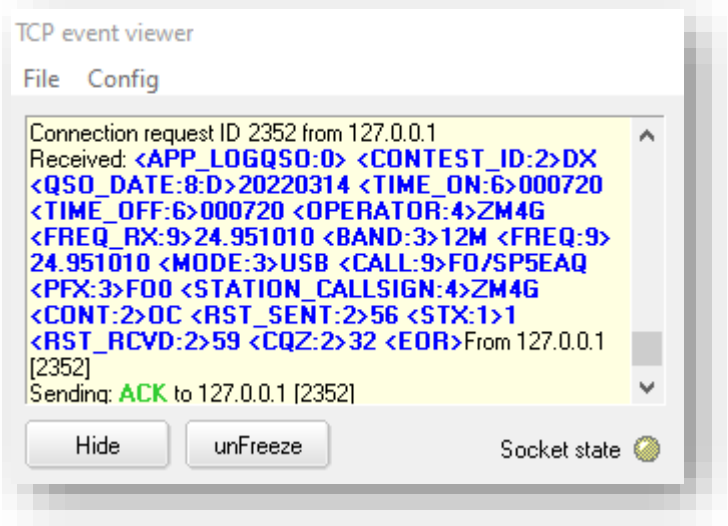
   A moment later you will see a bright green box meaning 'Ready, all systems go' ▶



8. Check and if necessary change the IP addresses and port numbers in N1MM-Logger32bridge to match the settings in N1MM+ (the UDP port) and Logger32 (the TCP port).

9. When you log a contest QSO in N1MM+, it is passed through the bridge and logged in Logger32 also. If you edit/correct or delete a logged QSO in N1MM+, it is modified or deleted in Logger32 too.

Copyright © 2026 Robert C. Furzer K4CY



◀ Here I have frozen the TCP event viewer for a closer look at the QSO information passed from N1MM+ via UDP to the bridge, then onward over TCP to Logger32.

Cool!

10. Meanwhile, the frequency and mode in Logger32's log entry pane track the radio's frequency and mode. So, if you log a non-contest QSO directly in Logger32, it should still be logged correctly, even while N1MM+ (not Logger32) has sole CAT control of the radio.

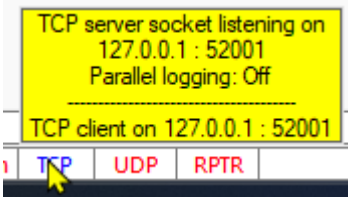
32.8 TCP Server FAQs

Q. Something's wrong with my TCP setup. How do I diagnose the problem/s?

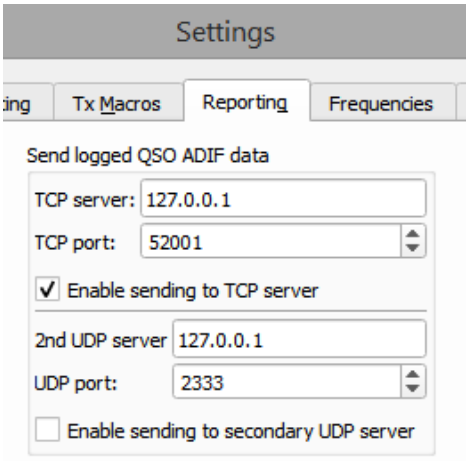
A. Show and keep an eye on the TCP event viewers while you do stuff. If that's not enough, or if messages flow past too quickly to read, or if Logger32 crashes out at the critical point, look carefully through *C:\Logger32\TCPLog.txt* for clues. The text file simply accumulates TCP messages during a Logger32 session … and is automatically cleared the *next* time Logger32 opens, so if you need more time to analyze it, make a copy *before* you next re-start Logger32.

Q. JTDX is complaining about "Socket operation timeout" or "Host not found"

A. In Logger32, [close then re-]open the TCP port. Note the IP address and port number displayed in the status bar as you mouseover the TCP panel ▼



Configure that address and port in JTDX through the **Settings** ⇨ **Reporting** tab ▶



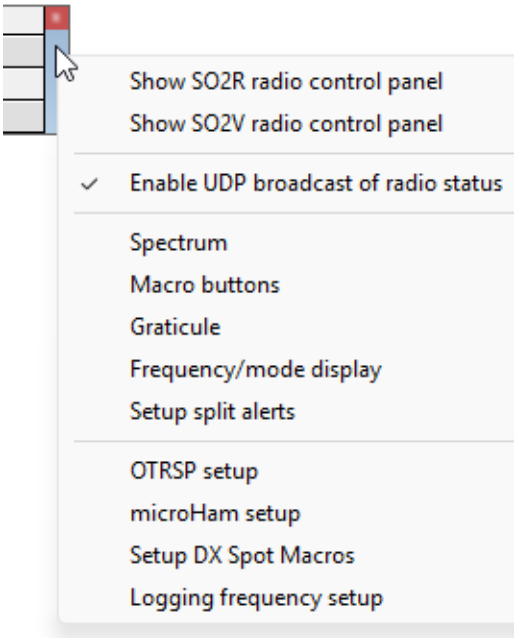
Copyright © 2026 Robert C. Furzer K4CY

Q. Can Logger32 send radio status info to other apps on other machines via UDP?

A. Yes. With a live CAT connection, Logger32 knows what is going on with your radio such as its VFO frequency and mode. Logger32 can rebroadcast the radio status information via UDP messages for use in other applications, such as a network app controlling your fancy auto-tune amplifier.

By default, Logger32 sends the radio status UDP messages to the same machine that is running Logger32 (IP loopback address 127.0.0.1) but that destination IP address can be changed in the configuration file:

1. First, tell Logger32 to start sending the radio information onto the network via UDP:
    a) Right-click <**Radio**> in the status bar at the bottom of Logger32's main screen.
    b) Click to open the **R**adio **C**ontrol **P**anel.
    c) Right-click the **R**adio **C**ontrol **P**anel in the area just below the corner X ▶
    d) Click to tick <**Enable UDP broadcast of radio status**>.



   That drops a Radio status UDP line into the current configuration file with the default IP loopback address.
2. Close Logger32 in order to be able to alter the *.ini* file that is read when Logger32 next starts up.
3. Using Notepad or your favorite text editor, look through *C:\Logger32\Logger32.ini* (or the corresponding *.ini* file/s for your configuration/s), to find the line "Radio Status UDP IP=127.0.0.1".
4. Change that loopback IP address to the IP address of the other system which is to receive the radio status information via UDP *e.g.* "Radio Status UDP IP=192.168.1.26" and save the edited *.ini*/s.
5. Start Logger32 and confirm that it and the other application are working hunky-dory.
6. If appropriate, close and restart Logger32 with any other configurations to check them too.

Q. Why am I logging digimode QSOs *twice* in Logger32?

A. Are you are using *both* UDP *and* TCP to connect JTDX to Logger32? If so, the ADIF QSO data is sent twice, received twice … and dutifully logged, twice!

Sever *either* the UDP *or* the TCP network connection to stop the dupes *e.g.* close the UDP or TCP ports from the status bar at the bottom of Logger32's main screen.

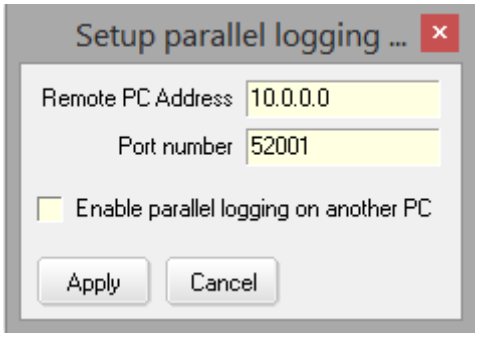
Note: if you close Logger32's **UDP** port, the **UDP** BandMap will also close. The clue is in the name. If you want to continue using the UDP BandMap, leave the UDP port open but stop your digimode software sending QSO info via TCP or close the TCP port in Logger32.

Copyright © 2026 Robert C. Furzer K4CY

Q. Why do I get error messages about the parallel logging port being unavailable?

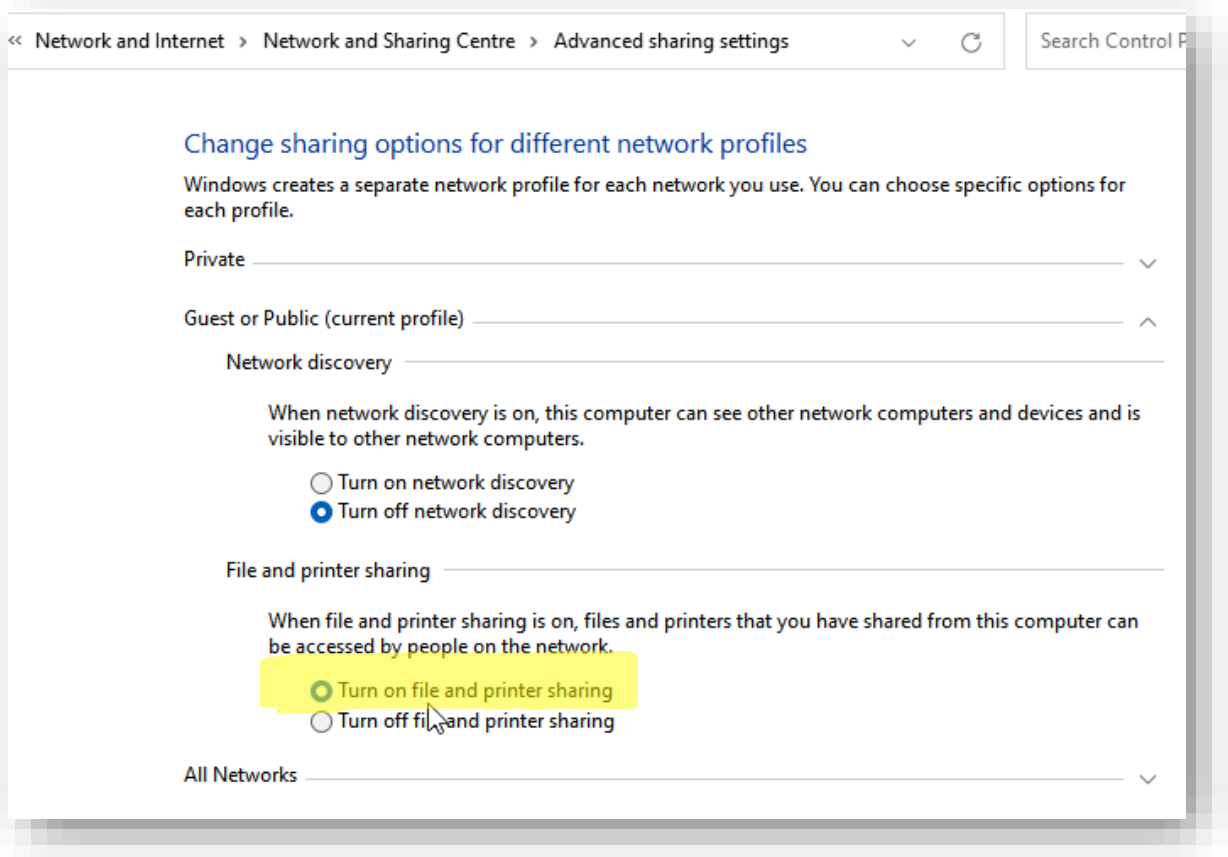
A. Because the PC you using is trying but failing to send logged QSOs to another PC via TCP for some reason.

Either fix the connection between the PCs or disable parallel logging by un-ticking the option ▶



One user reports that he had to enable Windows *File and printer sharing* to get parallel logging to work: the *File and printer sharing* settings are a little tricky to find. On my Windows 11 system, I use Control panel:

1. Tap the <**Windows**> key.
2. Type **Control panel** then click <**Control panel**>
3. Click to open <**Network and Internet**>
4. Click to open <**Network and Sharing Centre**>
5. Click to open <**Change advanced sharing settings**> on the left menu.
6. Under the current profile section, click to select <**Turn on file and printer sharing**> ▼



7. Click the <**Save changes**> button at the bottom.
8. Check whether parallel logging works now.

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: that setting opens TCP ports 139 (NetBIOS Session service and SMB) and 445 (SMB and DFS), weakening your network security. If you are concerned about ransomware or other malware, you might like to review your information risks and SMB security controls such as firewall rules, *particularly* if you are still using a rickety, old, unpatched Windows system (you crazy fool, you).

If those instructions fail on your PC, you'll have to rely on Google, Microsoft or blind luck.

Or stop logging QSOs in parallel. Have a break maybe. Stroke the CAT. Walk the DOG.

Q. Can I clear the TCP client and server event viewers? Can I resize them?

A. No. Well, closing and restarting Logger32 will empty them, and you can hide them, but it's not possible to resize them, or copy the message content to the clipboard. They are simply viewers.

Q. Is it possible to get L32 to export the log to ADIF format 'on the fly'?

A. Yes:

1. Close Logger32.
2. Make a safety copy of *C:\Logger32\Logger32.INI*[408]. Store it somewhere safe.
3. Open the *Logger32.INI* file a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad.
4. Under the [Globals] section, add a new line:

   ADIF Audit Trail=1

5. Open Logger32 and start using it as normal. As contacts are added to the log, modified or deleted, they will be dynamically appended as ADI-formatted QSO records to a file called *C:\Logger32\ADIF Audit Trail.txt*[409]
6. If needed, export an ADIF file from Logger32. Without selecting any export filters, this generates a static snapshot of the entire open log at the point you run the export.
7. Feed the ADIF files (the static snapshot, if needed, plus the dynamic audit trail with new and changed QSOs) to other applications for QSO mapping, statistical analysis or whatever.

Hinson tip: see the description of message 26 to find out how Logger32 handles QSO changes and deletions using ADI-formatted messaging.

[408] Use whichever *.INI* file corresponds to the configuration you use - possibly a new configuration with the contemporaneous ADIF export option enabled.

[409] The audit trail was designed and implemented this way in order to drip-feed QSOs to Club Log, eQSL, LoTW *etc*. via an app such as L32 LogSync.

Copyright © 2026 Robert C. Furzer K4CY

33 UDP broadcasting of DX spots

> "Broadcasting is
> the art of communication,
> connecting us to each other
> and to the world"
>
> *Tom Brokaw*

Logger32 can relay DX spots and local spots circulating on the shack network using the N1MM+ protocol. DX spots, as displayed in Logger32's DX Spots pane, are broadcast via UDP with the default port 12061. To avoid a port conflict, you can edit *C:\Logger32\Logger32.INI* and change the default port in the [Globals] section.

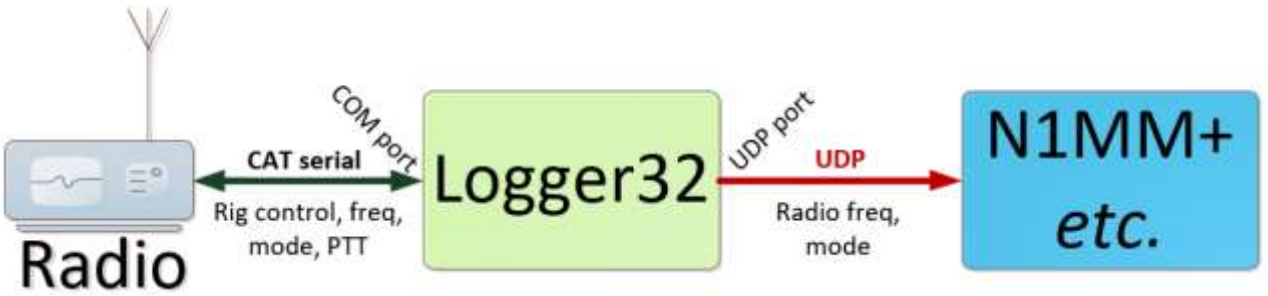
[Globals]

DX spot UDP Port=12061

DX spots are readily available to anyone from many sources - DX clusters, aggregators such as CC User, **R**everse **B**eacon **N**etwork, Skimmers *etc*. Once DX spots are fed into Logger32 and filtered as you require, they are then matched with your award records to identify any 'new ones'. Logger32 highlights the relevant DX spots in its DX Spots pane and on the BandMaps. Additionally, **the DXCC entity, distance and beam heading are included in the DX spot broadcast messages for use by other apps**.

Notes:

- Currently DX spots broadcast via UDP can be used only with Flex Radio.
- The N1MM+ protocol is explained below.
- Logger32 can *also* broadcast CAT radio data over the shack network via UDP ▼ for use by other radio apps such as N1MM+. See the Radio Control Panel chapter for details.



Copyright © 2026 Robert C. Furzer K4CY

33.1 Setup DX spot broadcasting

Right-click in the DX Spots pane then select <**Enable UDP broadcast of DX Spots**> ▶

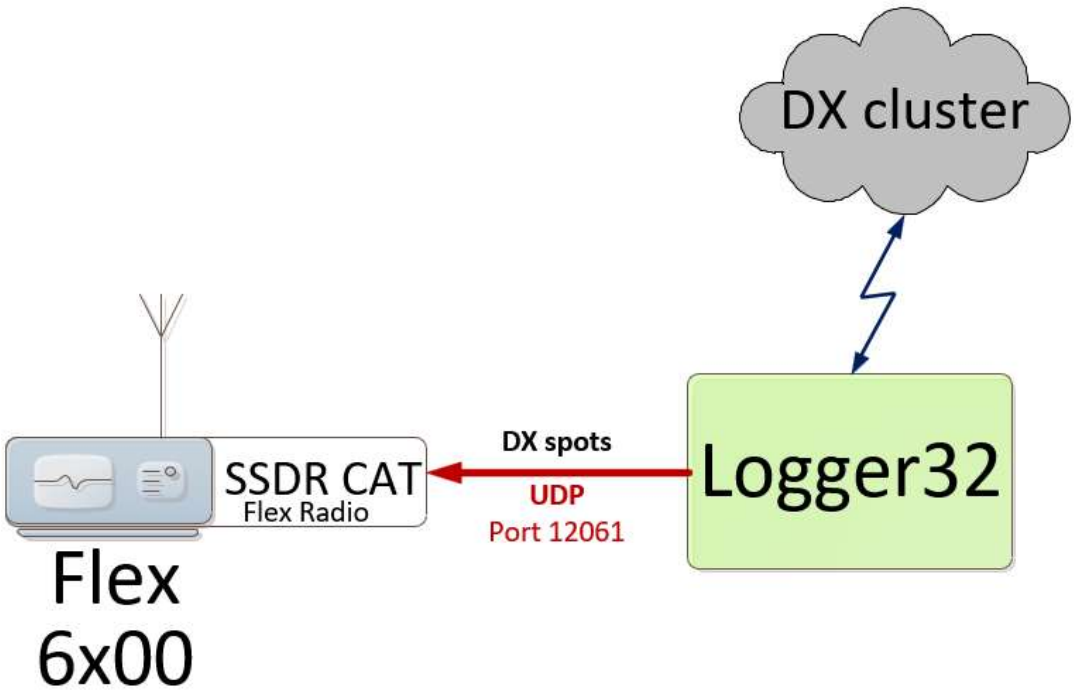


33.2 Using UDP broadcast of DX spots

Here are examples of how DX spots broadcast via UDP can be used with a Flex Radio.

33.2.1 Basic setup

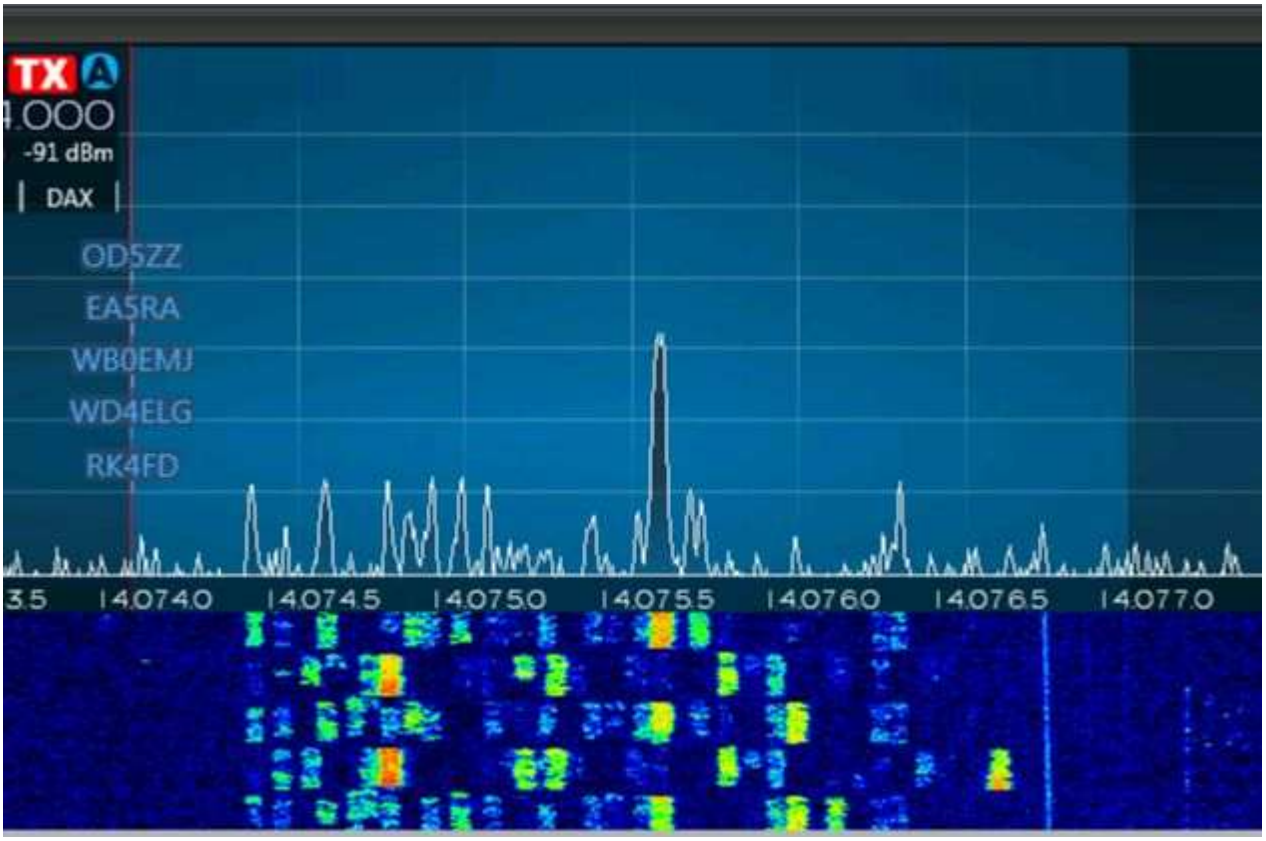
Flex Radio has a software module SSDR CAT. Connect SSDR CAT to Logger32 like this ▼



See below for details on how to set up SSDR CAT.

Copyright © 2026 Robert C. Furzer K4CY

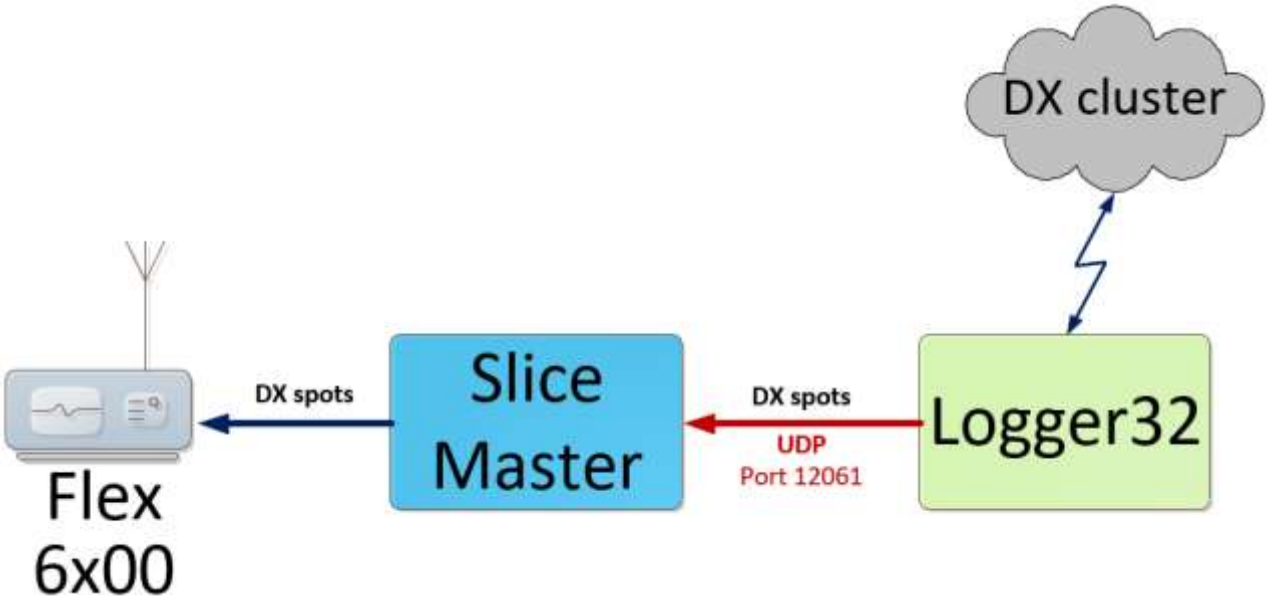
A plain list of callsigns from DX spots appears on the Flex Radio spectrum display ▼



By default, all the DX spots have blue text, with no prioritization (*e.g.* new countries and new WPX prefixes are treated the same), nor tooltips offering additional information.

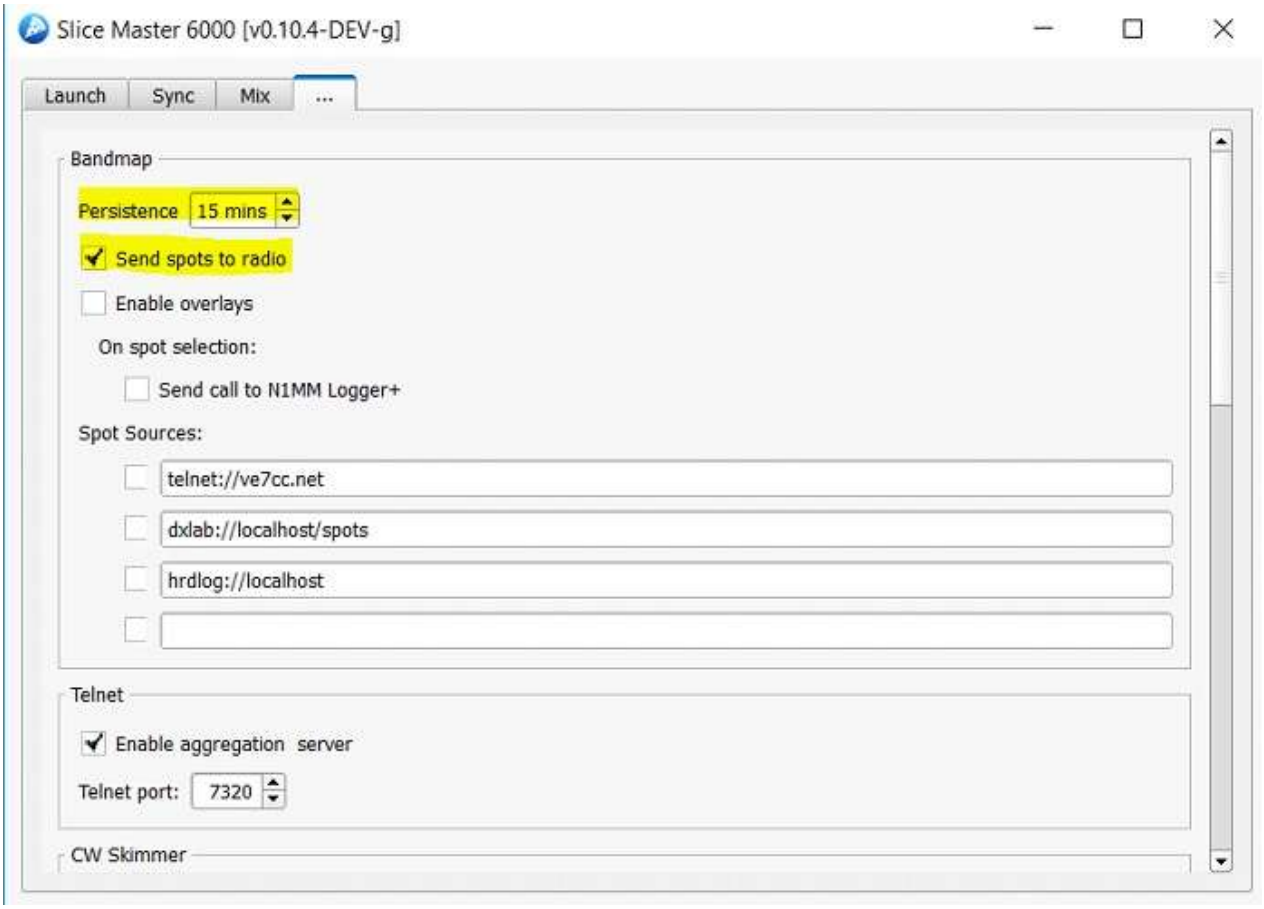
33.2.2 Improved setup

Functionality can be improved by adding Slice Master 6000 to the configuration ▼



Copyright © 2026 Robert C. Furzer K4CY

Slice Master 6000 can be downloaded and configured to pass DX spots received by UDP broadcast from Logger32 through to the Flex radio with these settings ▼



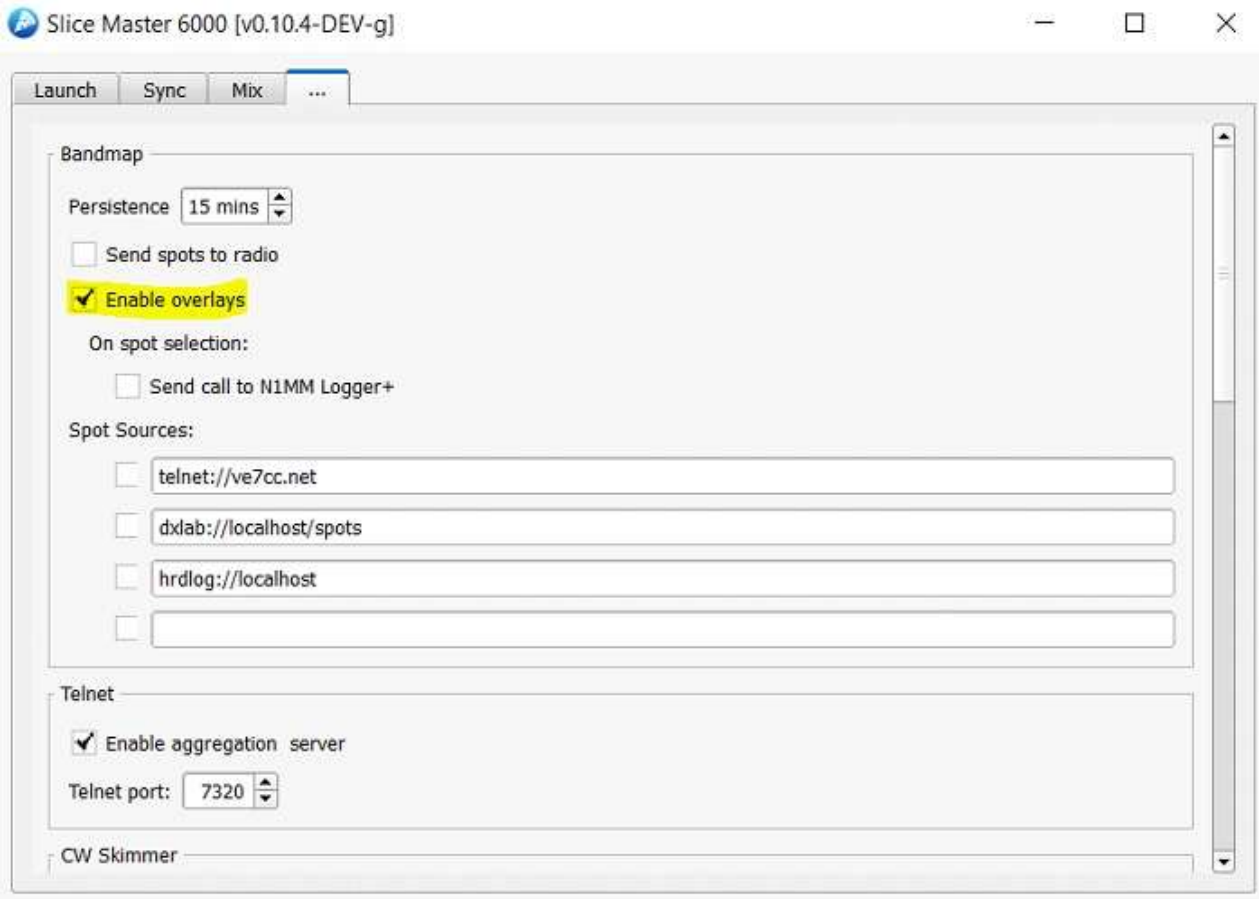
This improves the appearance of DX spots on the Flex Radio spectrum. DX spots are colored using the same color scheme as Logger32 uses to highlight DX spots (albeit the text itself is colored, rather than the background) and the highlighting is prioritized. A 'new country' will be indicated in preference to, say, a 'new mode' *e.g.* ▼



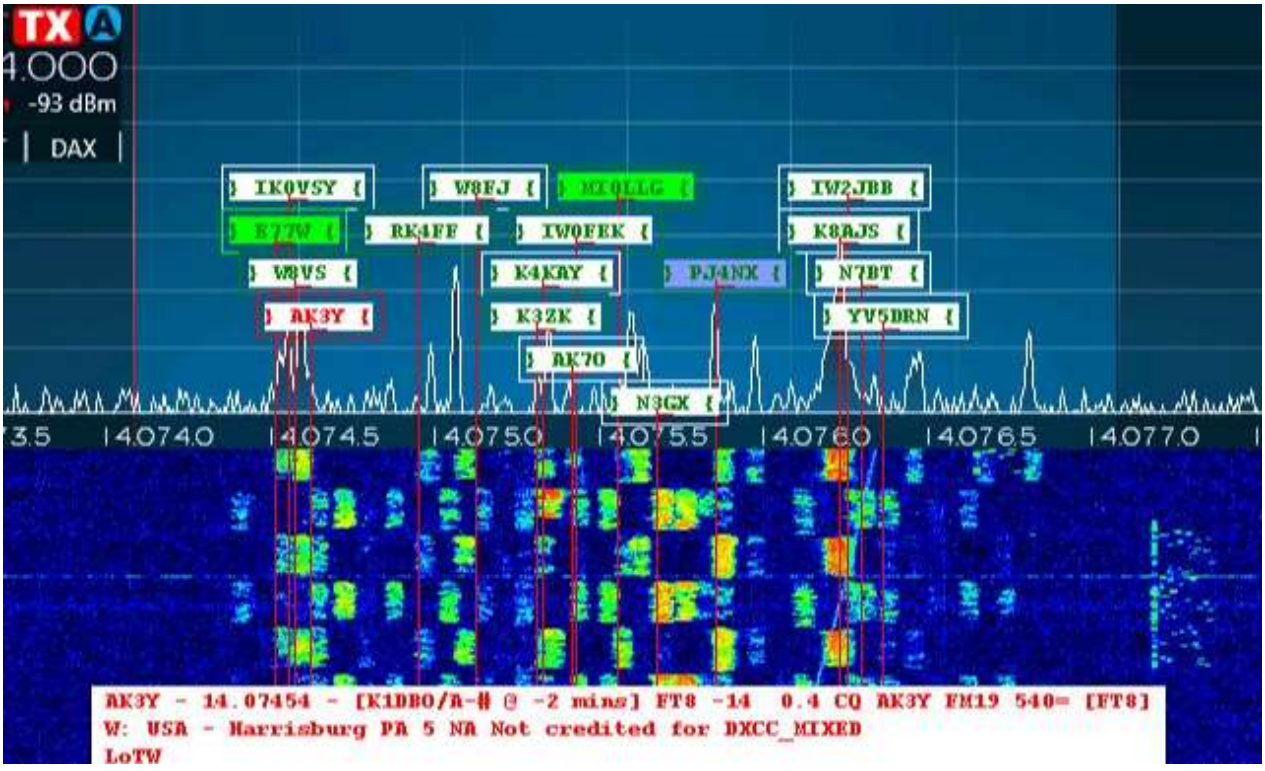
In that example, Logger32 is configured so that stations in new countries (such as UR7DWW) show up in red.

Copyright © 2026 Robert C. Furzer K4CY

Slice Master 6000 can be configured to add its own DX spots on top of the Flex Radio pop-up windows, using overlays ▼

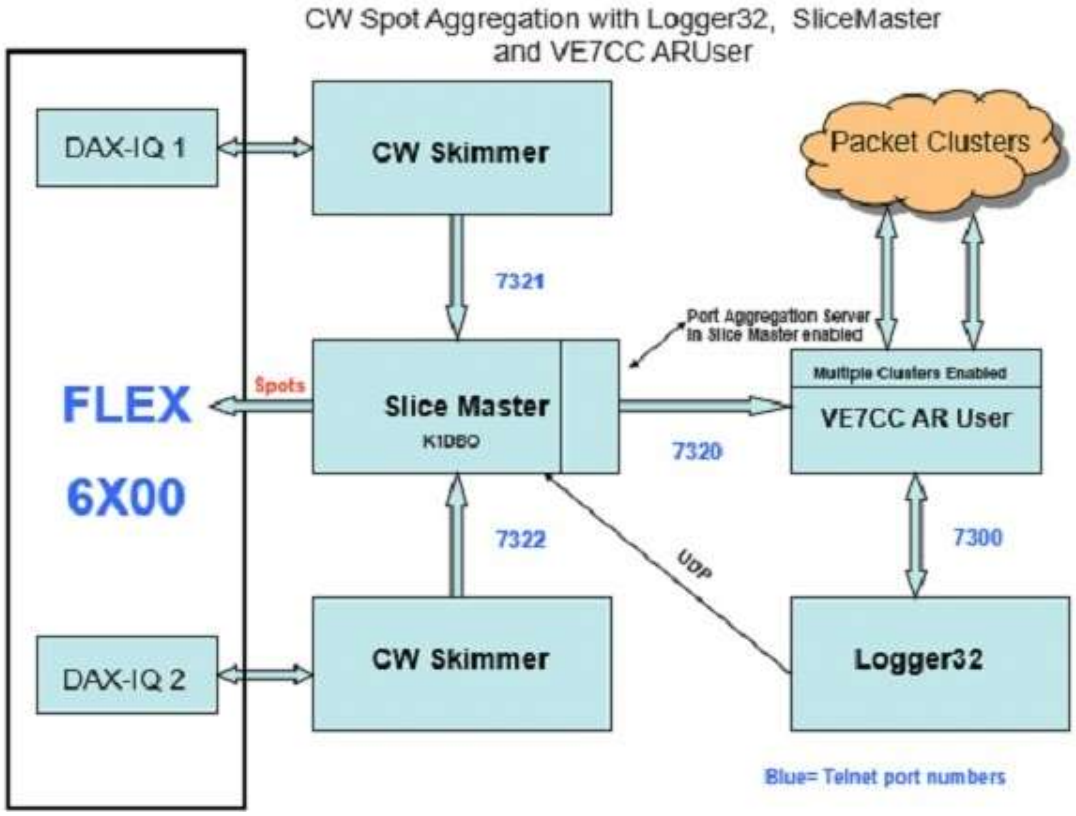


The result is DX spots with background and text colors that match Logger32's DX Spots pane. Tooltips showing additional information can be enabled. DX spots are prioritized with most wanted/need first … like this ▼



Copyright © 2026 Robert C. Furzer K4CY

For the very brave, and those with a seemingly insatiable craving for DX spots, a still more complex configuration could be implemented ▼



33.3 N1MM+ protocol

```
<?xml version="1.0" encoding="utf-8"?>
<spot>
 <StationName>K4CY</StationName>
 <dxcall>UN7NFD</dxcall>
 <frequency>14017.4</frequency>
 <spottercall>R0BB-#</spottercall>
 <comment>CW 4 DB 19 WPM CQ </comment>
 <action>add</action> ' add or delete
 <status>new qso</status>
  <logger32>
   <source>Localhose</source>
   <color>#FF0000</color> ' #RRGGBB
   <needed_reason>New Country</needed_reason>
   <dx_spot_priority>1</dx_spot_priority>
   <background>#FF9F80</background> ' #RRGGBB
   <country prefix="UN" name="Kazakhsten" zone="17" continent="AS" heading="15"
distance="10248Km"/>
   <qsl_method>LoTW, OQRS</qsl_method>
   <FocusEntry>1234567</FocusEntry>
 </logger32>
 <timestamp>2016-07-21 14:20:46</timestamp>
</spot>
```

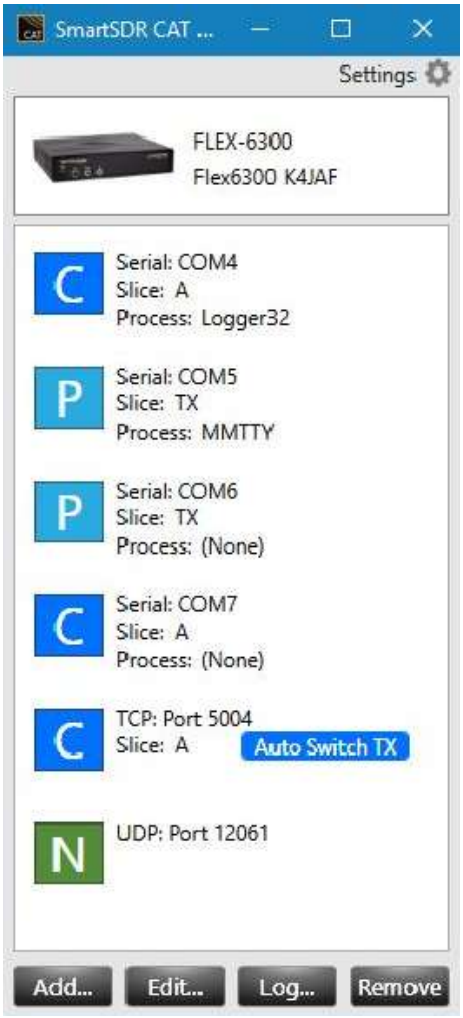
The N1MM+ protocol includes additional information from Logger32, such as DX spot priority, color, heading, distance and needed status.

◀ The XML message structure is like this.

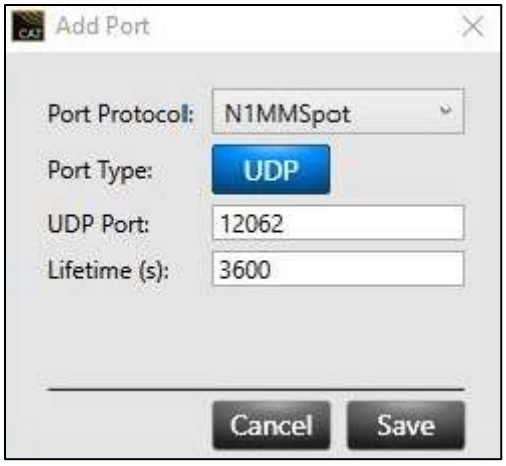
Copyright © 2026 Robert C. Furzer K4CY

33.4 How to create the SSDR CAT Flex Radio UDP Port

Open the SmartSDR CAT interface for the FlexRadio Systems Signature Series radios ▶



Click **<Add…> ⇨ Add Port**. Change the port protocol to N1MMSpot and the UDP port number to 12061 if necessary, then save the port configuration ▼



Copyright © 2026 Robert C. Furzer K4CY

34 Computer security, performance and capacity

> "Security is not about being perfect, it's about being resilient"
>
> *Theresa Payton*

34.1 Windows security and privileges

34.1.1 I'm too risk-averse to Run as administrator

Just two of Logger32's *many* functions require Windows administrator privileges[410]:

1. Resetting the system clock.
2. Auto-updates – specifically the final stage where Logger32 closes and restarts.

There are two methods of setting up Logger32 to run with administrator privileges:

1. **Program shortcut**: right-click the Logger32 shortcut on the desktop and select "Properties". Select the "Compatibility" tab. Under "Settings", select <**Run this program as administrator**> and click <**Apply**>.
2. **Program setting**: open File **E**xplorer using <**Win**+**E**>, navigate to *C:\Logger32* and right-click *Logger32.exe*. Click to select <**Run as administrator**>.

These are both one-time set-and-forget settings that apply in future whenever Logger32 is launched.

For security reasons, if Logger32 is run as administrator, then any programs that it calls – such as L32 LogSync and TQSL – must *also* run as administrator.

Hinson tip: being a risk-averse information security professional by day, I prefer *not* to run Logger32 and dependent programs as administrator. Instead, I use two pragmatic workarounds. (1) Windows 11 or Meinberg NTP keep my system clock reasonably accurate. (2) I acknowledge a Windows warning message near the end of the automated Logger32 update process, then manually close and re-start Logger32 after the update completes. That's all there is to it. Everything else works as it should … only with less risk.

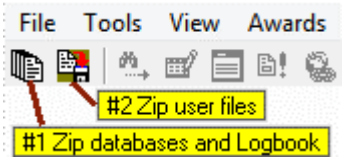
[410] Users report that Logger32 *can* access its databases on drives mapped to other computers, but Windows treats that as a privileged function requiring administrator rights. It is not a supported configuration though, whereas parallel logging is – and parallel logging does not require admin rights.

Copyright © 2026 Robert C. Furzer K4CY

34.2 Backups

Sometimes 'stuff happens' such as PC hardware failures, malware, user errors, power surges and cuts, bugs and RF interference. 'Stuff' cannot be totally prevented so when (not if!) incidents occur, backups may be the final option to recover your valuable data – your logbooks, for instance, and all those Logger32 configuration options that you've spent *ages* getting just so.

Back up or stuff up: your choice.

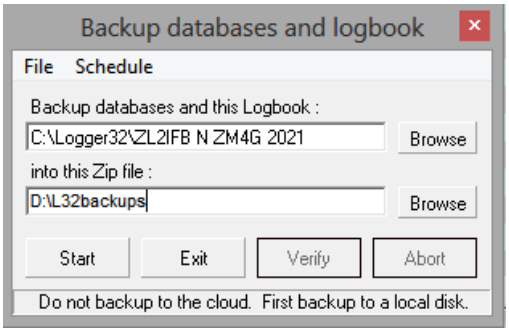


◀ Two icons on the toolbar make it simple to backup your data manually when you want, while automated backups save your data even if you forget.

34.2.1 Zip databases and Logbook

Toolbar icon #1 makes a compressed backup copy of your open log (the database and logbook files) in the ZIP format.

Only one logbook can be backed up at a time: if you have several logbooks, open and then backup each logbook separately … and try not to forget any!



◀ In the upper box, enter the disk, folder and name of the logbook you wish to backup. If you are not sure of the details, use <**Browse**> to find it.

The default logbook name is Logbook32, generally in *C:\Logger32* but you may have moved or renamed it.

In the lower box, enter the disk[411], folder and file name of the *.ZIP* you wish to create. Note the warning at the bottom of the form: save backups to a local disk first[412].

Click the appropriate button to:

- **<Start>** the backup running.
- **<Exit>** the backup function *without* backing up any [more] files.
- **<Verify>** *i.e.* check that a backup *.ZIP* file made previously is intact[413].
- **<Abort>** a backup in progress.

While the backup is running, the status and any instructions appear at the bottom of the form.

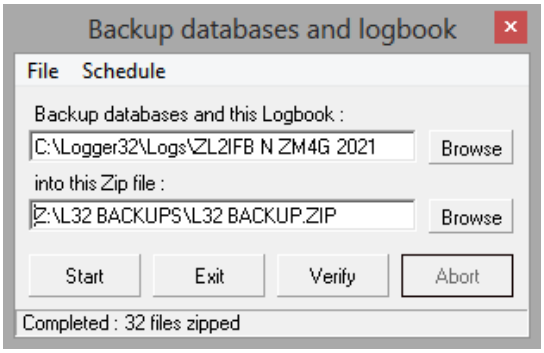
[411] If you have more than one physical disk in your PC, saving the backups on a different disk to the primary data protects you against hardware failure of the primary disk. Better still, copy the backups to a USB memory stick, a CD-RW disk or an external/portable disk drive, then physically disconnect it from your PC and put it somewhere safe. Separate logical partitions on the same physical disk aren't as safe.

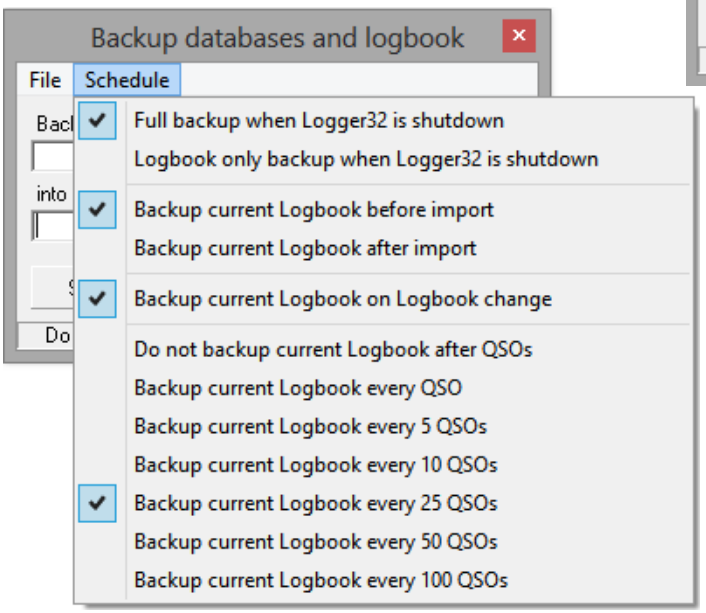
[412] If you save backups to the Internet/cloud using Google Drive, Dropbox *etc.*, the system takes time to synchronize the files after making a backup before making another. If that doesn't work, save your Logger32 backups locally (*e.g.* on your C: drive) first, then upload them to the cloud storage manually.

[413] The terse verification message "Completed" means "All OK, no problems found, have a nice day".

Copyright © 2026 Robert C. Furzer K4CY

When finished, it tells you how many files were backed up ▶



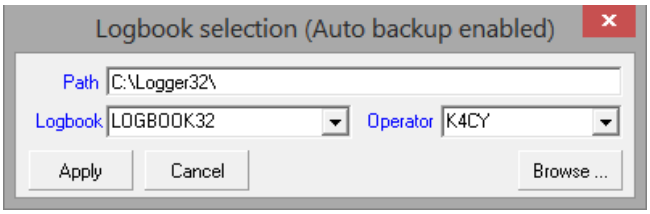


◀ On the form's menu, <**File**> has a single <**Exit**> option to close the backup function.

<**Schedule**> offers a stack of options for automating the backups:

- **Full backup when Logger32 is shutdown**: backs-up the open logbook plus the associated database files and the *.INI* configuration files. However, if you normally leave Logger32 running 24x7 on the shack computer and only rarely shut it down (*e.g.* for Windows or Logger32 updates), your most recent backup may be some time ago.
- **Logbook only backup when Logger32 is shutdown** is almost the same but it skips the *.INI* files. This saves a tiny amount of time and disk space, but potentially leaves you without a recent backup of those *.INI*s if 'stuff happens', facing the grief of having to reconfigure Logger32 from scratch. Why take the risk? This option is not recommended.
- **Backup current Logbook before import**: importing ADIF files into your log *should* be fine in theory *if* those ADIF files are truly ADIF-compliant. In practice, it is possible to mess up your log big time, if for instance the imported file contains QSOs made by a completely different operator, or if the QSOs were logged in local time instead of UTC. Making a log backup first gives you the ability to revert to the saved version if the import goes horribly wrong.
- **Backup current Logbook after import**: a post-import backup makes it relatively simple to get back to that point if you go on to mess things up after the import *e.g.* by 'correcting' the imported operators or times, only to discover that they were right in the first place.
- **Backup current Logbook on Logbook change**: if you run several logbooks (*e.g.* a personal one plus family, club or contest logbooks), this option automatically backs up the logbook that was originally open before you opened a different logbook.

The caption for the change logbook function reminds you if a backup is to be made ▶



If you do not select this option and neglect to make sufficient backups manually, you are vulnerable to 'stuff' affecting the original logbook – such as someone accidentally deleting the wrong file, or malware trashing your data.

Copyright © 2026 Robert C. Furzer K4CY

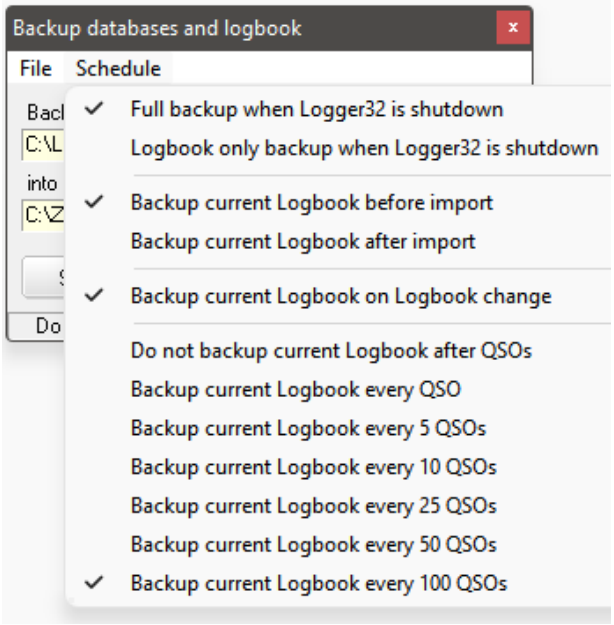
- **Do not backup current Logbook after QSOs**: leaves you to make backups at some other point, either manually or automatically (*e.g.* when you shutdown Logger32 *if* you have selected one of the first two options on this menu).
- **Backup current Logbook [after] every QSO, or [after] every 5, 10, 25, 50 or 100 QSOs**: it may seem cool to backup your valuable log automatically after each individual QSO … but hold on a moment. Think it through.

  Logger32 maintains a rolling cycle of just 10 automatic backups, so after the 11th QSO is logged, the backup made after the 1st QSO is deleted to make way for the new backup. Consequently, if 'stuff happens' to your log but you fail to notice it within the time period of those 10 QSOs, you *may* not have a previous backup to restore. Also, backups require a bit of processor and disk activity which can delay the ability to start logging your *next* QSO – especially if you have a large log and/or a slow computer. If your operating style is leisurely, your log relatively small (say, less than 20,000 QSOs) and if your computer is up to the task, then you may not even notice the delays. If you are a 5NN merchant making rapid-fire QSOs, the delays will frustrate.

  Alternatively, you may prefer to backup after 100 QSOs. Given that rolling cycle of 10 backups, this option means if necessary you can restore your log as it was up to 999 QSOs ago. Aside from avid contesters and genuine DX stations constantly working large pileups, most regular operators would take weeks, months or years to log 999 QSOs, leaving plenty of time to spot and hopefully recover from the 'stuff that happened' to the log. However, your most recent 99 QSOs *may* be lost … unless you have some other mechanism to recover them – for example, if you upload your QSOs routinely to an online log system such as LoTW, Club Log or QRZ.com, or if you (quite sensibly!) make regular PC backups.

  So, those are the extremes *if* you decide to make backups after logging a certain number of QSOs. Your backup strategy is down to your preferences and risk-aversion.

Hinson tip: being a professionally-paranoid information security consultant by day, backups are like oxygen for me … so despite having multiple backups, I'm still anxious about losing important data due to malware, disk failure, ransomware, operator error, computer theft, shack disasters and more. My backup regime includes daily, weekly and monthly backups, mostly disk-to-disk and disk-to-cloud plus offline archival to USB memory sticks, external hard drives and DVDs. I routinely upload my QSOs to online logs such as Club Log and LoTW. If anything, I have too many backups, so restoring data takes me a while to find the best backup or archive copy. Still, that's better than too few! I configure Logger32 to make backups every 100 QSOs because of all the other backups I have, and to avoid slowing me down when I'm on a run … but you are in a different situation so it is down to you.



Copyright © 2026 Robert C. Furzer K4CY

Yesterday – The Backup Song

A little ditty by Sunni Freyer and Bill Frick

Yesterday,
All those backups seemed a waste of pay.
Now my database has gone away.
Oh I believe in yesterday.

Suddenly,
There's not half the files
there used to be,
And there's a deadline
hanging over me.
The system crashed so suddenly.

I pushed something wrong
What it was I could not say.
Now my data's gone
and I long for yesterday-ay-ay-ay.

Yesterday,
The need for back-ups
seemed so far away.
Thought all my data
was here to stay,
Now I believe in yesterday.

Woe is me,
It happened so unexpectedly,
Now my clients are all mad at me,
How will I ever collect my fee?

Sheepishly,
I explained my catastrophe,
I expected some sympathy,
Now all of them are leaving me!

Let me be,
Wallowing in my misery,
Disaster, I could not foresee,
All because of a bad floppy.

Pity me,
All I wanted was to format B,
My finger slipped and pressed C,
And now I can't find a boot floppy.

Here try this,
He promised it would bring more bliss,
I didn't know it had a virus,
Its aim was true it did not miss.

Copyright © 2026 Robert C. Furzer K4CY

*"Periodically do a test restore and check that it is an exact match of the original. Several built-in utilities in Windows xx can be used to restore a file to a different location and then do a comparison with another. A backup that you can't use to restore is a waste of time and offers a false sense of security. For instance: backup to external hard drive, (auto-)update of LoTW (if used, some note-type fields may be lost), *AND* periodic trips to the bank safety deposit box to exchange a micro SD card backup for its partner. Cloud is another option if you trust the provider - see their policies: usually best effort or industry practice, both defined as that's what we do. Or whistle in the dark with your fingers crossed... HI"*

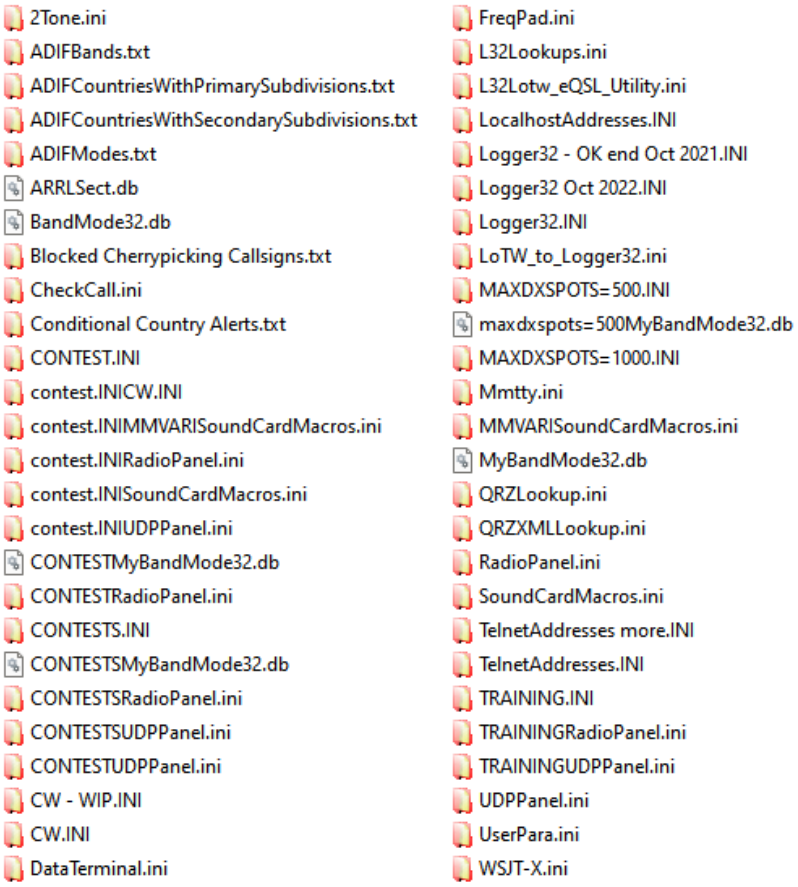
George VE3YV / K8HI

34.2.2 Zip user files

This function compresses and saves copies of your Logger32:

- *.INI* and *.ini* files containing your settings – all those little Logger32 customizations you have so patiently made and accumulated with your choices of colors, highlighting option, macros, port settings, configurations, alerts *etc*.
- *.txt* files from *C:\Logger32* containing ADIF-specified modes plus administrative subdivisions such as states and counties.
- Non-logbook *.db* database files containing details of bands and modes *etc.*

For example, these are the 52 files in a user file backup I just created ▶



Copyright © 2026 Robert C. Furzer K4CY

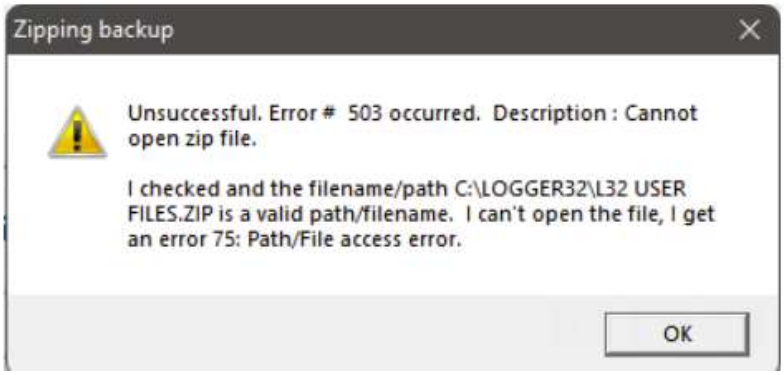
Hinson tip: why not make a backup *right now*, then double-click the *.zip* file to open it and see what yours contains? At least then you'll have at least one backup! Keep it somewhere safe!

To use the function for the first time, open it with icon #2 on the toolbar, then type the disk, folder and name of the *.ZIP* file you wish to create, or click the <**Browse**> button then find a suitable folder (*e.g. D:\L32-backups\Logger32Backup.zip*). Then click <**Start**> or another button: the form is virtually identical to the <**Zip databases and Logbook**> function described above, including the menu options to automate the backup.

The *next* time you run this function, it recalls the same details so all you need do is quickly check that the details remain appropriate then click the <**Start**> button.

While backups are being generated, Logger32 temporarily suspends the cherry-picker and TCP socket to conserve resources and complete the backup as efficiently as possible.

Hinson tip: keep an eye on those backup messages, as things may go wrong *e.g.* this ▼ error message started appearing during the automatic backups when I closed Logger32, due to some obscure problem accessing the folder or file. Manually running the log and user files backup functions pinned down the error to the user files backup, and relocating the user files backup file solved it.



A week ago, my station took several lightning strikes that damaged some equipment and my control/logging computer ***including*** *the backup thumb drive*[414]*. The computer was replaced and a new Logger32 installed. I downloaded from both Club Log and LoTW to reconstruct my log.*

Gary, W7DO

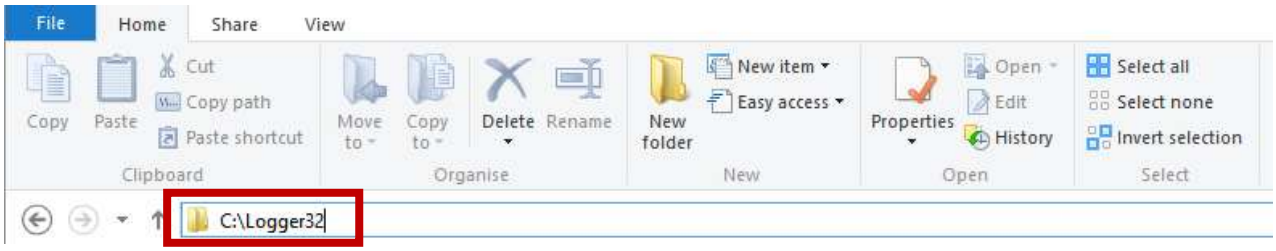
[414] This is why I recommend "offline" backups – backup devices or media that you *physically remove from the PC* after making a backup to be stored somewhere safe. Fortunately, Gary was able to recover most of his log without the damaged thumb drive, thanks to having other backup copies as well. A lucky escape!

Copyright © 2026 Robert C. Furzer K4CY

34.2.3 Restoring Logger32 files from backups

Especially if you are not confident about doing this, it is worth making a copy of the entire *C:\Logger32* folder *before* you restore files from your backup *.ZIP* so that if you mess up, at least you can get back to where you are now. Here's how to do that, step-by-step:

1. Close Logger32 in the normal way.
2. Open File **E**xplorer using <**Win**+**E**> (while pressing the key printed with the Windows icon, usually to the left of the spacebar, tap the **E** key).
3. Click the address bar ▼ near the top, then type C:\Logger32 <**Enter**> [415]



4. Press <**Ctrl**+**A**> to select the folder's entire contents.
5. Press <**Ctrl**+**C**> to copy the folder's entire contents.
6. Press <**Alt**+**UpArrow**> to go to the C: drive, or in the address bar type C:\ <**Enter**>.
7. Right-click somewhere near the middle of the window, then click **New** ⇨ **Folder**. Windows prepares to add a new folder, waiting for you to name it.
8. Type a name for the new folder *e.g.* Logger32 SAVED 2026 <**Enter**>
9. Double-click to open the new 'SAVED 2026' folder.
10. Press <**Ctrl**+**V**> to paste the copied contents of your original Logger32 folder there.

Now you are ready to **restore** your backed-up logbook and/or user files:

11. Click to open the disk drive and folder containing your Logger32 backups.
12. Double-click to open the appropriate *.ZIP* backup file - probably not the most recent one (created in step 1 above) but the previous one.
13. Extract the relevant file/s into your *C:\Logger32* folder … unless your logbook is saved elsewhere (*e.g.* in a separate data folder or another disk), in which case you'll need to restore the logbook files there instead.
14. Run Logger32 and check that the log was restored correctly. Look for:
    - A *severely* broken log that Logger32 cannot even open.
    - Substantial sections missing from the logbook *e.g.* all QSOs before or after a given date.
    - QSOs made under different callsigns (*e.g.* your normal and contest calls, or home and portable operations) missing from the log.
    - Missing or corrupted statistics (you *might* need a recalc).
    - Other forms of data corruption.

[415] *C:\Logger32* is the suggested/default installation folder for Logger32. Yours *may* be different.

Copyright © 2026 Robert C. Furzer K4CY

15. If there are problems, you can try:
    - Deleting the restored files and restoring other backups instead.
    - Importing ADIF backups of the same log (the import routine will skip *identical* QSOs but will import QSOs with differences, potentially creating duplicates).
    - Importing ADIF log files downloaded from LoTW, Club Log, QRZ.com *etc*. in the hope they will not generate *loads* of duplicates requiring a lot more work to resolve.
    - Manually correcting erroneous QSO records.
    - Opening a new logbook, importing one or more recent ADIF log backups.

34.3 Antivirus



Occasionally, antivirus packages report problems with *Logger32.exe* or some other file associated with Logger32. To date, as far as we know, these have all been false alarms so **don't panic** … but since it is impossible to *guarantee* that Logger32 or any other software will always be totally 100% virus-free, it pays to be cautious.

Most antivirus products (including the free ones) identify all the common viruses, but they differ in their detection of uncommon/novel viruses and in the speed of response when you report virus alarms to the suppliers.

You can upload suspicious files to the VirusTotal website to have them processed through a bank of antivirus products, free of charge:

- If *all* the antivirus products report no infection, the warning from your antivirus software was *almost certainly* another false alarm. Please report it to your antivirus provider anyway and take their advice – typically that means waiting for them to conduct further tests[416].
- If only a few lesser-known antivirus products report infections, the warning was *probably* a false alarm … but there *might* be an issue. Please report it to your antivirus provider and take their advice. Wait patiently for them to conduct further tests and respond.
- If several antivirus products, including any of the major brands, report infections, the warning should be taken seriously. Please report it urgently to your antivirus provider and take their advice. Wait patiently for them to conduct further tests and respond. Meanwhile, avoid attempting to run Logger32 as there *appears* to be a genuine problem. Instead, spend your time checking your backups, preparing in case you need to recover an infected system.

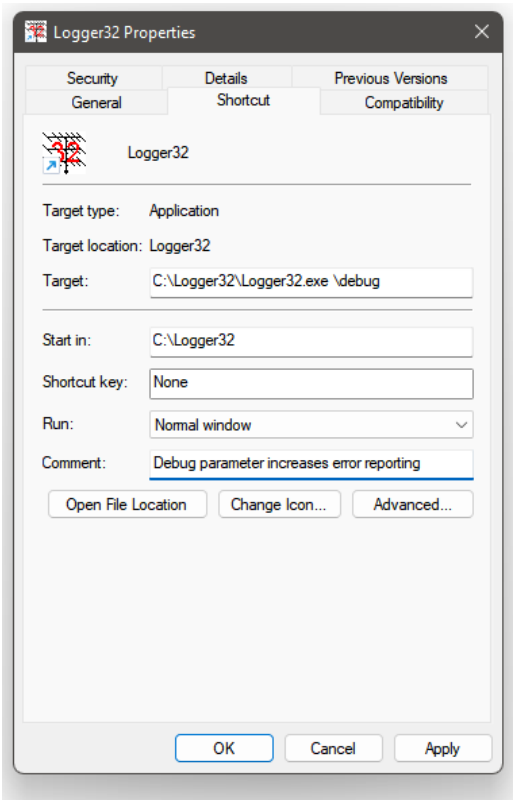
If you can't be bothered to use a service such as VirusTotal, or don't trust the results, *please* report the alarm to your antivirus provider and wait for their response. If other Logger32 users also report it, the incident may receive a higher priority and a quicker response. If nobody bothers to report it, nothing is likely to happen about it.

[416] If you are brave and keen to continue using Logger32 despite the slight virus risk, you might choose to exclude or whitelist the offending file in your antivirus package even before the antivirus company responds with an update. If you do, don't forget to remove the exclusion/whitelist after the antivirus company confirms it was a false alarm and updates the antivirus to stop reporting it.

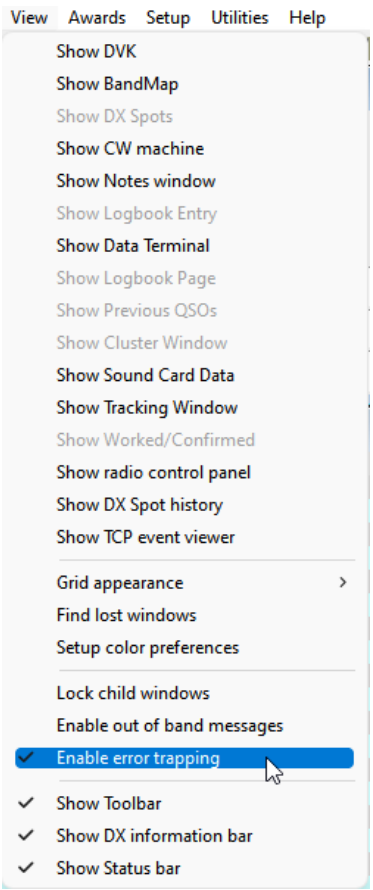
Copyright © 2026 Robert C. Furzer K4CY

34.4 Debugging

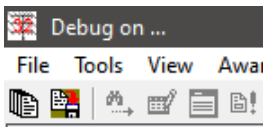
Inevitably as Logger32 has steadily grown more complex and functional, bugs that have crept quietly under the covers emerge unexpectedly, causing problems for *some* users under *some* conditions on *some* systems. Finding out what triggers a given error is generally a prerequisite to figuring out why things are going wrong, finding the bug/s in the source code or architectural flaws in the design, and exterminating them[417] – an ongoing activity for Bob and the beta test crew.



To help this, Bob has been systematically adding error-reporting functions (internal diagnostics) to Logger32 in order to detect and report, if not actually resolve, the problems that crop up in use. For most of us, most of the time, errors are either not triggered or are resolved automatically, so by default the error-reporting functions in Logger32 are largely disabled, although 'critical' errors are still reported wherever possible[418]. To help diagnose what is going on in other error situations, Logger32 users can enable additional error reporting by either or both of these two methods:



- Right-click your Logger32 desktop icon and append **\debug** to the *Target* command line ▶ then click <**Apply**>. This enables debugging mode *as the program loads*, briefly displaying a notification message in the top left corner ▶ whenever you start Logger32 by double-clicking the modified icon[419]. Load-time debugging is automatically disabled at the end of the loading phase, or sooner if five error messages have been triggered.



- ◀ With Logger32 running, click to tick **View ⇨ Enable error trapping.** When ticked, this error trapping comes into effect *after* the loading phase *i.e.* during normal program use. Use this *and* **\debug** to cover both phases.

[417] Being able to trigger a specific bug on demand makes it much easier to test and prove that it has been eliminated, once debugged. The intermittent/inconsistent ones are particularly awkward little buggers.

[418] The nastiest of critical errors may result in a sudden crash of Logger32.exe and/or Windows, with no opportunity to report anything. Logger32 simply quits, drops into an infinite loop or kills the PC. Disabling DEP seems to solve one such issue but, to be perfectly honest, we're not sure why or what the actual issue is, largely because we can't reproduce it.

[419] … so you may prefer to retain a desktop icon *without* the **\debug** modifier to avoid unnecessary distractions from minor bugs *e.g.* if you are about to run a pileup, DXpedition or contest.

Copyright © 2026 Robert C. Furzer K4CY

34.5 Patching and updating

34.5.1 Software updates

It's worth staying up to date with patches and new versions of all the software you normally use, including Windows itself plus apps such as Logger32 and utilities such as TQSL, L32 LogSync and JTDX or other digimode software. Aside from occasional improvements to functionality and performance, patches often resolve bugs, some of which can be serious (*e.g.* security weaknesses being actively exploited by hackers and malware, including ransomware, spyware and network worms) or critical (show-stoppers).

Hinson tip: *if* you have genuine reasons for *not* patching your system at least once a month, pay even more attention to your backups since there is a greater chance of your system being compromised and perhaps trashed. Losing up to a month's data is bad enough. Losing it *all* could be an unmitigated disaster of epic proportions.

34.5.2 Hardware upgrades

Upgrading or replacing your computer hardware is generally a last resort, but eventually you may be forced to do so when you've reached the limits of computer tuneups and the PC can no longer function effectively with the loading imposed upon it, or if the software it will run is so out of date that it is no longer supported and patched (implying cybersecurity issues). Face up to facts.

Interference is a significant issue for PCs used in ham shacks. Shoddy switchmode PSUs and plastic cases, low quality/old exhaust fans, and unshielded cables are likely to generate/radiate both RF and audio QRM, and may succumb to RF fields from your transmitters and antennas. There's only so much you can do with cable chokes, even those of the appropriate ferrite mix with reasonable impedance (meaning *hundreds* of ohms) at the amateur frequencies you are using.

Additional RAM, faster disks (especially solid-state/NVME disks) and more powerful CPUs may be worth it, provided your system has the capacity - meaning the physical space, sockets and BIOS/drivers needed for the upgrades. You may need a newer motherboard, PSU, graphics and audio cards facilities … but beyond a certain point, it becomes uneconomic to carry on throwing money at an old computer with piecemeal upgrades.

Hinson tip: generally speaking, conventional desktop computers [420] from the big-brand manufacturers are good value for the shack, especially if you find like-new or professionally refurbished business machines steeply discounted on the surplus market. Commercial organizations often write-off and replace their IT systems routinely every few years. The old ones *may* not have the performance needed for the very latest business apps or games, and they may not be officially certified to run the very latest version of Windows, but they will usually run Logger32, JTDX *etc.* just fine. Refurbished PCs are cheaper than new ones through the surplus/recycling market, and charity shop or eBay bargains are cheaper still. As a bonus, big-brand PCs are generally well engineered, competently designed and built as a complete system.

[420] There are far fewer upgrade possibilities with laptops, especially old laptops, and mini-PCs. Some components are custom-manufactured for each model in order to fit the very limited space in the case, with power consumption a further significant challenge. However, disks and RAM *may* be updateable – though you may need to take it to a PC repair shop with the knowledge and tools to do so. At least being portable, it's easy to take in and take home!

Copyright © 2026 Robert C. Furzer K4CY

34.6 Network security

Most of us these days have plugged our shack and other home LANs and mobiles into the Internet: we are part of the global online community and all that comes with it, with ready access to valuable information … *plus* misinformation, malware, fraudsters, hackers, snoops and spooks.

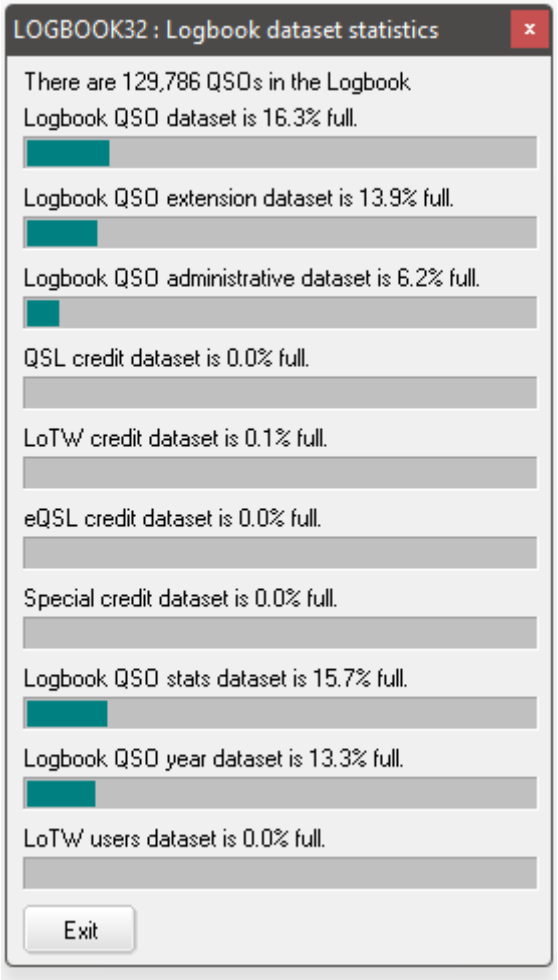
'Network security', then, involves a blend of technical and non-technical controls such as:

- 'Hardening' systems, for example configuring security correctly, using long, strong passwords, and keeping up-to-date with security patches;
- Using firewalls designed to block known network probes and attacks;
- Using antivirus software to block known malware;
- Avoiding avoidable threats *e.g.* steering well clear of dubious websites and apps;
- Reducing your personal vulnerability *e.g.* by learning about and hence not falling for "obvious" too-good-to-be-true scams;
- Reducing the impacts of incidents *e.g.* by having usable backups;
- Reducing the 'attack surface' which includes not opening network ports unnecessarily.

On that last point, think twice before you open TCP or UDP ports on your Logger32 system. Are you opening the door to trouble? Will the apps handle hackers or malware while protecting your privacy and other interests?

34.7 Capacity

Using **Tools ⇨ Database maintenance ⇨ Logbook details**, take a quick peek at Logger32's database statistics every so often (maybe once a year) to see how it's doing ▶



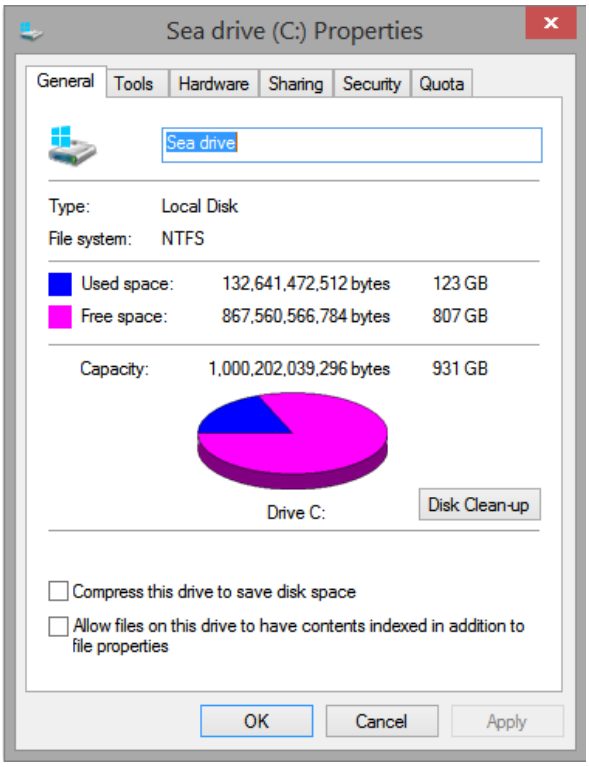
The database management system around which Logger32 is built has the capacity to log about a million QSOs per log – more than that if you only log the essential information on each QSO.

Hinson tip: if that 'million QSOs/log' limit might become a show-stopper for you, you can either split your log into chunks (*e.g.* separating routine from contest QSOs, or recent from ancient QSOs), reduce your QSO rate, or choose another ADIF-compliant logger with even greater capacity. You can easily export your logs from Logger32 in ADIF format at any time.

Copyright © 2026 Robert C. Furzer K4CY

Remember the first lesson on day one of computing 101: you can only put 10lb of crap in a 10lb bag.

Bob K4CY



◀ Keep an eye on the disk capacity (specifically, the free space – the lipstick pink area on the pie chart by default) as well, by opening File **E**xplorer using <**Win**+**E**> then right-clicking a disk and selecting <**Properties**>.

Tidy up the disk and delete unnecessary files occasionally. Once the free space falls below, say, 20% consider investing in another disk, ideally a solid-state drive to reduce database accesses to a *fraction* of the time and power needed for a conventional magnetic hard drive.

34.8 USB power capacity and quality

Jose N4IS advises using an external powered USB hub if your USB CAT connection doesn't work at all, or is unreliable. It seems the USB chips in his IC-7851 and IC-9100 draw more current than the PC's USB port can readily supply. Without the powered hub, he had to power-cycle the rigs several times to kick the CAT into life. Bob K4CY further advises using a powered USB hub for all USB devices in the shack, connected to the shack PC with a high-quality isolator. Gary ZL2iFB says he must be lucky since all his directly-connected USB devices work just fine. YMMV.

Hinson tip: keep a watchful eye on the collective power demands of your USB devices, regardless of whether they are connected directly to the PC's USB ports or via external hubs (powered or unpowered). Also be careful about which USB ports you are using, particularly for peripherals that are power-hungry and/or need high-speed data: whereas USB 2.0 ports can deliver just 2.5W, USB 3.0 or 3.1 ports are rated to deliver up to 4.5W, or up to 7.5W for USB 2 or 3 ports working in '**P**ower **D**elivery' mode. USB type C ports can potentially deliver even more power, in some cases up to 240W … *provided* you are using the correct cables, the PC or powered hub power supply is up to the task, and the devices negotiate their power demands successfully with the PC. More info. 'USB tester' or 'USB doctor' devices measure the voltage and current to calculate power … and yes you can make your own, use an ordinary multimeter or scope, or simply check the rated device specifications or labels for clues. Or wing-it like me. Be lucky.

Copyright © 2026 Robert C. Furzer K4CY

34.9 Computer tuneup

34.9.1 Reducing CPU usage

The most likely cause of performance problems in Logger32 is that your CPU cannot keep up with all the jobs being asked of it, hence it runs short of horsepower. Actually, Logger32 is not *entirely* to blame:

- Gremlins on the payroll at Microsoft ~~may~~ have snuck all sorts of stuff onto the computer, ostensibly as essential parts of Windows. They've packed so much stuff onto the poor horse that it can barely stagger along and can't see where it's going.
- The people at the company that installed your operating system and associated software and firmware (normally the folks from whom you purchased the computer, assuming you didn't assemble it yourself) may have snuck additional "helpful" support utilities and spyware onto the system.
- Bob Furzer, Makoto Mori, Joe Taylor and others may have loaded-up your computer with fancy audio-processing software for RTTY, PSK, FT# and other digimodes.
- Other apps and utilities you run (such as antivirus, email, web browsing and office) all take their bytes out of your CPU capacity.

Logger32 itself gets busy at times, for example processing DX spots, applying updates to the DXCC prefixes and exceptions from Club Log, or recalculating your statistics. These are intensive, demanding activities that inevitably take a while to run. Mostly, though, Logger just trundles along at less than 5 or 10% of the CPU capacity on average, with short peaks to about 25 or 50% on a typical PC less than ~5 years old. If you are still using an Amstrad, well, good luck with that.

The following recommendations should reduce the load on your PC, leaving more horsepower available for Logger32. A few apply to general operation but most are specific to digimodes which inevitably sap the CPU.

Person who chases two rabbits won't catch either

Aki JA1NLX

34.9.2 Quick test

A quick and easy test is to close other programs (such as your browser and email software) apart from Logger32 to see if it runs any better alone. If so, it appears those other programs are competing with Logger32 for the computer's finite resources - most likely its CPU processing, RAM memory, disk access or network bandwidth, sometimes several of those. Windows Resource Monitor usually shows which.

If some apps are particularly resource-hungry, either cut down on your usage (*e.g.* avoid running them at the same time as you are using FT8), move them to a different computer or look for lightweight, more efficient alternatives.

Hinson tip: one way to 'move them to a different computer' is to separate, say, FT# processing from DX spot processing onto separate shack computers (such as a desktop and laptop), both running Logger32 and linked by parallel logging.

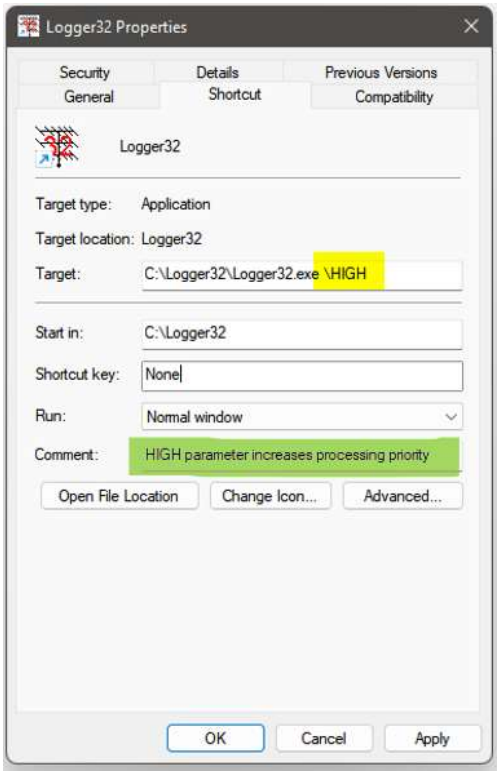
Copyright © 2026 Robert C. Furzer K4CY

34.9.3 Increase Logger32's CPU priority



Let's get one thing straight here: the command line parameter **\HIGH** has nothing to do with hallucinogens. It increases Logger32's processing priority to – yes, you guessed it – 'high', giving *Logger32.exe* a greater proportion of the available CPU time than most other Windows processes.

If you feel Logger32 is important enough to deserve a bigger slice of your CPU than all the other humdrum stuff running on your PC, by all means:



1. Close Logger32.
2. Right-click your Logger32 desktop icon ▶
3. Append **\HIGH** to the command line in the <**Target**> field.
4. Optionally, type a few words into the <**Comment**> field as a clue about the parameter you have just added.
5. Click <**OK**> to apply and save the change, closing the form.
6. Launch Logger32 by double-clicking the desktop icon in the normal fashion, and watch it *fly* as high as Benjamin Franklin's kite.

Hinson tip: if your system becomes unstable, or other apps now run as slow as a tired and emotional DXer struggling with an unruly pileup, reverse the change by undoing what you just did. Remove **\HIGH**. Coast gently back down to Earth. Hopefully you won't crash-and-burn.

34.9.4 Disable unnecessary autostart programs

Over time, typical shack PCs invariably end up running quite a variety of apps (applications and utilities). Some programs start up automatically when the computer boots *i.e.* the operating system itself, of course, plus apps that have been installed and are configured to start automatically at boot time (*e.g.* antivirus and backup utilities). All these programs load the PC to some extent, so from time to time it's worth checking that only necessary programs autostart and run in the background.

Hinson tip: Resource Monitor lists [most of] the operating system programs and apps that are running on your PC. If you see apps that you aren't actually using – perhaps programs that you once tried out or used but decided not to proceed with – they are obvious candidates to be stopped … but be careful about accidentally stopping things that are, in fact, necessary: if you are at all uncertain, Google the executable names for clues about what they are before deciding whether to kill them off, and stop them auto-starting [more below].

Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: Microsoft's Autoruns for Windows utility, originally part of the wonderful Sysinternals suite, makes it even easier to change which programs, libraries *etc.* load when your PC boots, checking all the places where they hide out … but it's not for beginners. If you are not careful, you can easily mess up your system and perhaps break it completely so it doesn't boot and run properly, if at all.

Hinson tip: if you decide to have a go at this, **don't skip step 0 below and be *sure* to have good backups** *before* you leap in at the deep end. Once you are floundering in the water, gasping for breath, it's a little late to realize you can't swim and your lifejacket is still at home in the cupboard.

The following procedure only addresses the main autostart apps. Other utilities may use different autostart mechanisms … and frankly you'd best leave them alone unless you know what you are doing.

0. Optional but recommended: create a system restore point in Windows so you can more easily reverse the following changes if things don't quite go to plan, by restoring the system.
1. Press <**Ctrl+Shift+Esc**> to launch <**Task manager**>.
2. Click to open the <**Startup**> tab or icon, displaying a table of autostart apps.
3. Click the <**Status**> column header to sort the table by that column. If the <**Enabled**> apps are not at the top of the list and there is a little arrow head pointing up, click it again to invert the arrow head and reverse the sort order (it is a toggle).
4. For each of the currently-enabled apps in the table in turn, click the <**Disable**> button to stop it autostarting when Windows boots.[421]
5. Close Task Manager by clicking the corner X.
6. Reboot the computer[422].
7. With hardly any apps now running except the Windows operating system itself plus utilities started by some other autostart mechanism, this is your opportunity to launch Logger32 and see how fast it runs when given sole access to all the PC's resources (well almost).

"Am I the only one who has zero problems with Logger32? For years now I've just switched on my PC, loaded it up, worked a few on CW and switched it off again. Reliability personified."

Vince GOORC

[421] If shown here, you may prefer to leave apps such as your antivirus and backup software enabled.

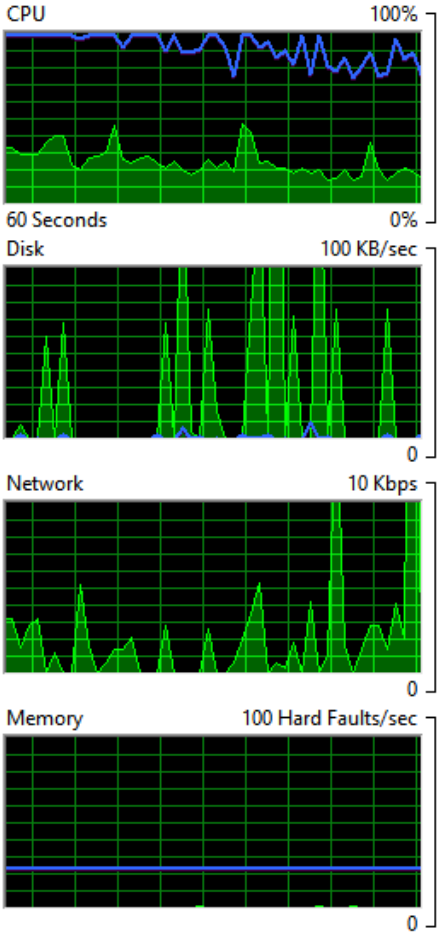
[422] It is a worthwhile precaution to disconnect from the Internet (*e.g.* remove the Ethernet cable, disable your WiFi) before rebooting if you will be running without antivirus. This will cut off the most significant threats *i.e.* malware and hackers on the Internet. Program and database updates, callsign lookups and DX cluster services won't work in this state, of course, but you can still check out the rest of Logger32.

Copyright © 2026 Robert C. Furzer K4CY

7. With Logger32 running, run Resource Monitor as well by typing it into the start menu search bar or by pressing <**Win**+**R**> then typing **resmon <Enter>**.[423]
8. In Resource Monitor, click to open the <**Overview**> tab.
9. As you use Logger32, keep an eye on the pretty graphs on the righthand side of the Resource Monitor display ▶

   Watch for any Logger32 activity that consumes a lot of CPU power or creates a lot of disk, network or memory activity, taking the light green zones up towards the top.

   Try out all the features that you normally use, for a little while. Maybe have a go at RTTY, FT# and other modes. Run some macros. Open the maps. Check your awards tables. Search for old QSOs … whatever. Exercise the system to see how it performs under load.
10. If Logger32 runs successfully and performs well, try manually starting the individual apps that previously autostarted, one at a time. Wait a couple of minutes after each one loads, giving it time to settle down. Watch Resource Monitor to see what effect it is having.
11. Now you can begin to add back the autostart programs you previously disabled through Task Manager or Autoruns. Refer to steps 2-6 … only this time you will be clicking the <**Enable**> button for one or two apps and rebooting each time, then checking performance again using Resource Monitor.



If you are lucky, you may discover one or more specific apps causing severe performance problems, sapping the life out of your system. Ask yourself whether you really *need* them to autostart every time you boot your system. If not, go back into Task Manager or Autoruns to disable them.

Hinson tip: if they are not autostarted, you can always manually launch whichever apps you need … and then close them when you've done whatever it is that you needed to do.

With v4 updates, I'm adding yet more error checking, further improving the stability of Logger32.

Bob K4CY

[423] Resource Monitor is so useful that I have it running routinely … but then I used to be an IT systems administrator back in the day when even "mini" computers were the size of fridges and "micro" meant "Can be transported on a sturdy trolley". I have pinned its speedo icon to my Windows taskbar, making it accessible with just a click. It does consume some PC resources but, for me, it *easily* earns its keep. System Explorer and Task Manager show something similar but not nearly so well. Resource Monitor is king.

Copyright © 2026 Robert C. Furzer K4CY

*"As for the bag that doesn't fit the sh*t ... there would be material to make a Shakespearian comedy ... The reality, if we want to put it simply, is that they sell us sh*t and make it look like gold, physically on a noble 8 core they are only 4 cores, which manage 2 threads each. So 4 cores, 8 threads, I don't know if I understand, let's not hide behind fig leaves, if this is the state of the art we're already scr*wed from the start. On July 16, 1969 Apollo 11 went to the moon with an on-board computer the power of a C64 and now we're getting slapped for four "SH*T" calculations? I'll just give an example and I'll stop here: Look, the architectures improve like wine over time ... and even at the same frequency, the architectures following the Pentiums are much faster. To give you an example: if there was an i7 with a single core, and with a frequency of 2.0 GHz, this would be much faster than an old Pentium 4 at 3.2 GHz ... this is why an i7 at 2.66 GHz is much faster than an QX9770, even if it has 600 MHz less, and this regardless of the presence or absence of Hyperthreading. This happened in 2007.... now we are in 2024 with numbers that I won't even mention because I know the shame ... but what the **** are we talking about? There is only one truth and it must be said though ... it is that you are trying to trick a motorbike and make it run on a racetrack where only real racing cars should compete, nothing but emulation! Technology must go hand in hand with everything else too, Windows must have the courage to pull the plug and make the real leap leaving the old junk stuck to their b*tts, rather than stressing out customers with policies, certificates and this security of my b*lls, but you Americans are afraid of your shadow I forgot ... In the end China will save us. America first!"*

Spleen duly vented by
Stefano ('Steve') IK4DRY
Lightly edited, expletives deleted

Copyright © 2026 Robert C. Furzer K4CY

34.9.5 Selectively reduce, disable or avoid resource-intensive activities

These are generally the most resource-intensive functions in Logger32:

- Importing ADIF logs is slow: a large log can take *hours.* Luckily, this is not a frequent occurrence.
- Recalculating statistics has to read and check every QSO, updating the statistics accordingly.

> **Hinson tip**: the Logger32 beta test team uses this to compare performance across different PC hardware and software settings. Fast machines recalc quicker!

Who	PC specification	Logged QSOs	Seconds to recalc	QSOs /sec	Secs/10k QSOs
SV1VS	i5--6500 32Gb Win10	50,000	221	226	44
JA1NLX	i5-11500 16Gb M2g4 SSD Win11 Turbo 4.4 GHz	60,500	290	209	48
SV1VS	i7-6700 32 Gb SSD Win10	50,000	245	204	49
ZL2iFB	i5-13500 16Gb Win 11 NVMe sys SSD data Turbo	131,317	645	204	49
ZL2iFB	i5-13500 16Gb Win 11 NVMe sys HDD data Turbo	139,589	720	197	52
ZL2iFB	i5-13500 16Gb Win 11 NVMe sys SSD data Eco	131,317	708	185	54
N3ND	i7 32Gb	62,000	350	177	56
W4UCK	i7-9700 16GB SSD Win11	13,150	85	155	65
SV1VS	i5-4590T 16Gb Win10	50,000	327	153	65
N8DC	i7 16Gb SSD	16,173	132	123	82
ZL2iFB	i7-4790 SSD 24Gb Win10 Turbo	125,821	1,080	117	86
ZL2iFB	i7-4790 SSD 24Gb Win10 Eco	125,655	1,260	100	100
JA1NLX	i7-7700 32Gb SSD	60,500	636	95	105
JA1NLX	i5-11500 8Gb SSD Win10 Eco	60,500	649	93	107
JA1NLX	i5-11500 16Gb M2g4 SSD Win11 Eco	60,500	720	84	119
JA1NLX	i5-11500 8Gb M2g4 SSD Win11 Eco	60,500	728	83	120
JA1NLX	i5-11500 8Gb PCIe4 SSD Eco	60,500	730	83	121
	Mean	**74,001**	**560**	**146**	**78**

'Recalculate statistics' is manually timed on a quiescent PC with the current release or beta of Logger32, with no DX spots flowing, no JTDX/WSJT-X decoding and little else running.
The stated times *include* the database optimisation steps at the end of the recalc.
"Eco" refers to the Windows 'balanced', 'power saver' or custom settings.
"Turbo" is the Windows power setting for maximum CPU performance.

- JTDX or WSJT-X: identifying and decoding a bunch of FT# signals from a captured audio file takes some serious number-crunching every few seconds. Read more in the next section.
- Correcting DXCC entities using info from Club Log is quite intensive.

Copyright © 2026 Robert C. Furzer K4CY

- LoTW-log synchronization using an LoTW ADIF file or Club Log method also involves quite a lot of processing.
- BandMaps and DX cluster functions: do you really need to see *all* the spots on *all* the bands at once, including those low-banders that are almost certainly unworkable during the daytime due to E-layer absorption? Receiving, analyzing, filtering, displaying and potentially highlighting each spot inevitably consumes CPU cycles. It helps to:
  - Connect to just one or maybe two DX clusters at a time: most carry similar information anyway. Two connections gives you redundancy, while three or more achieves little except load up your poor PC.
  - Configure spot filters at the DX cluster node/s rather than in Logger32, passing and processing spots only for the specific modes and bands you are interested in using. Filter at source.
  - Resist the temptation to monitor digimode spots – particularly for FT# – since they can flood your system for little return.[424]
- Map displays – particularly redrawing the base maps (luckily, this need only be done once per station QTH).
- The XY display for tuning RTTY signals in MMTTY.[425]
- RTTY notch and receive bandpass filters.
- RTTY oversampling.
- RTTY FIR filter (try the IIR or PLL filters instead).
- RTTY TX bandpass filters (limit the risk of transmission problems by keeping the audio settings in the green zone … and if this change makes no appreciable performance difference, re-enable the TX filters).
- 'Animations' generally *e.g.* menus that slide into view instead of simply appearing.
- Unnecessary/unwanted utilities running behind the icons in the notification area at the bottom right of the desktop.

34.9.6 Defragging is deprecated

As files – large files especially – are routinely created, updated and deleted from conventional magnetic spinning hard drives, the storage tends to become physically fragmented with data scattered across the disks, slowing down file access a little while the disk subsystem writes new data or finds, retrieves and stitches together all the parts of a saved file. Whereas years ago, manually 'de-fragging' (de-fragmenting) your disks was found to speed things up, ever since Windows 7, the operating system handles disk defragmentation intelligently and automatically as a background system task, once a week.

Hinson tip: these days, additional defragging is rarely helpful and may even be counterproductive. On solid state drives, the physical data storage locations don't affect access times, so defragging simply reduces the drive's remaining lifetime with no performance benefit. **Leave it to Windows!**

[424] The UDP BandMap lists the stations you are receiving and decoding on the current band: who cares what other FT# users on the same band might be decoding? If you can't decode them, you can't work them.

[425] Simply de-selecting the tuning window in <**View**> hides but does *not* stop the processing for the fancy XY graphical display. Turn off the XY display completely by clicking its toolbar icon.

Copyright © 2026 Robert C. Furzer K4CY

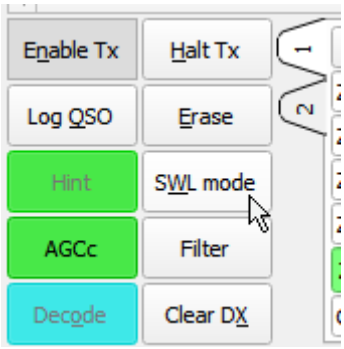
34.9.7 Throttling-back on JTDX

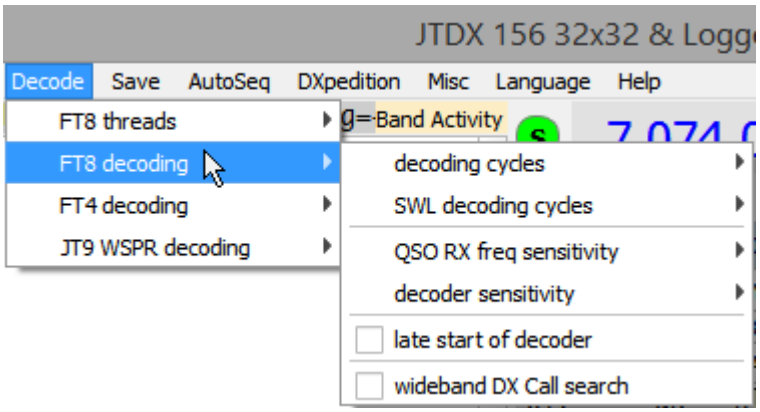
The clever digital signal processing and decoding that makes digimodes such as FT# so effective at winkling-out and making sense of very weak signals on crowded bands requires intensive computing power for a few seconds at the end of each transmission period … which limits the amount of processing that Logger32 can do on the UDP messages it receives (*e.g.* choosing which cherry to pick) in the time available to do so.

So, be careful with the decoder parameters in JTDX/WSJT-X to avoid draining the life out of your PC and increasing the lag such that your FT# transmissions are unreasonably delayed. Even a high-specification system may struggle on a busy band with so much going on.

Here are the *suggested* settings for JTDX, for someone mostly using FT8 on the HF bands on a PC system of poor to mediocre specification:

- Turn off <**SWL Mode**> on the main control panel. If the button is green on your system, click to disable it, ending up like this ▶





- Under the JTDX <**Decode**> menu ▲
  - Let JTDX control the number of <**FT8 threads**> automatically.
  - Limit the number of <**decoding cycles**> to 2 or perhaps just 1. Weak signals may be partially or completely hidden under stronger ones: the first pass through the received audio finds the strong signals which are then (in essence) removed before a second pass looking for weaker signals. There may not be time for a third pass.
  - Set the <**QSO RX freq sensitivity**> to low.
  - Set the <**decoder sensitivity**> to low. High sensitivity is needed for extreme weak signal communications such as EME, but not for ordinary HF DXing.
  - Do not enable <**late start of decoder**> to avoid cutting into JTDX's decode time, thereby limiting the time remaining for Logger32 to react to the decodes.
  - Disable <**wideband DX Call search**>.

Hinson tip: these are just suggestions. Feel free to experiment with different settings, preferably changing just one thing at a time and trying it out for a while to determine the effects before moving on. Keep a close eye on the CPU tab of Resource Monitor. For bonus points, tell us how you got on via the Logger32 reflector.

Hinson tip: if you are serious about weak-signal digimode DXing, you might try running JTDX on a dedicated high-spec computer optimized for that purpose, passing QSO and other information to Logger32 on your normal shack PC via the shack LAN. The advice on connecting Logger32 to a Raspberry Pi using a PowerShell script to read the Pi's UDP messages may inspire you.

Copyright © 2026 Robert C. Furzer K4CY

34.10 Security, performance and capacity FAQs

Q. My old shack PC crashed and burned – it's all gone. Stuffed! Wrecked! Toast! I now have a shiny replacement PC … so can I recover my log?

A. Yes, *provided* you have a backup.

Hopefully you took the *copious* advice in this User Manual to make backups, so here are some places to look for them, ranging from the most to the least valuable:

- **A parallel-logging PC**: *if* you had been parallel logging to another PC, *if* that PC survived the incident and *if* it had a complete, up-to-date, parallel copy of your entire log, then you don't really *need* to recover/restore anything! Simply export that log in ADIF format from the parallel PC and import it into Logger32, having installed it on your shiny new PC.
- **Regular offline PC data backups**: did you run PC backup software making regular (*e.g*. daily or weekly) backups of your data - including your Logger32 log files - to an external disk drive or some other data storage device/media? Assuming the storage device or medium at least survived the disaster (*e.g*. it wasn't destroyed by the same accident, malware, lightning strike, theft, flood, bush/house/shack fire, eruption, earthquake, vandal, pestilence/rat/mouse/insect infestation, power surge, hardware/firmware/software failure, missile strike, zombie attack … or whatever), you should have a recent copy of your log files to restore to a fresh installation of Logger32 on the replacement PC.
- **Online (cloud) backups**: if you had been running a cloud backup facility, hopefully the backups included your Logger32 data, and hopefully they survived the incident intact (*e.g*. they were not also destroyed/corrupted by ransomware or other malware).
- **Offline log backups/archives**: have you at some prior point (maybe last month, or last year, or last birthday …) exported your entire log as an ADIF file and saved that on a USB stick, external hard drive, CD-ROM/CW-RW/DVD disk, or some other backup format/medium? Is it stashed safely away under the mattress, at a friend's place, in a fire safe or bank vault? Locate the most recent and complete archive.
- **Online logs**: had you been uploading your logged QSOs periodically (individually or in batches) to Club Log, QRZ log, LoTW or some other log server? If so, you can probably download the log … but you will probably discover that the recovered log is minimalist, containing only the basic QSO information (*i.e*. the QSO dates and times, callsigns, bands/frequencies, modes and probably nothing more – no notes, names …).

The process for recovering your log is much the same, regardless of what kind of backup/s you have:

1. Get Logger32 installed on the replacement PC first. Don't log any new QSOs at this stage.
2. Configure Logger32 to your liking. If you have backups of the original "user files" (particularly the *.INI* and *.ini* files from *C:\Logger32*), you may prefer to restore and use them, although if the replacement PC has different disk drives, screen, fonts *etc*., you will need to make adjustments, install missing fonts *etc*.
3. If you wish, log some *fake* QSOs to check the configuration is working OK. [If for some reason you also log any genuine QSOs, they will soon be deleted … so make a separate note of the QSO details or export them as an ADIF file so that they can be recovered later, once your original log is restored from the backup.]

Copyright © 2026 Robert C. Furzer K4CY

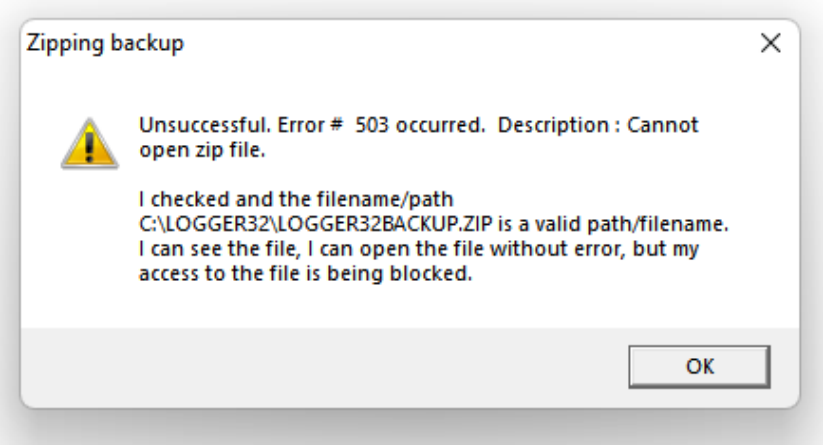
4. If you have found a backup of Logger32's original log files in Logger32's internal format:
   - In Logger32, check and make a note of the folder and file name of the open log.
   - Close Logger32.
   - In File **E**xplorer, open the folder where the log is stored and delete the log files.
   - Restore the backed-up log file to the folder on the replacement PC.
   - Start Logger32.
   - Open the restored log using **File ⇨ Change Logbook**.
   - Check that the log appears intact.
   - Recalculate statistics using **Tools ⇨ Database maintenance ⇨ Recalculate statistics**.
5. If you have found a backup log in ADIF format (perhaps several):
   - In Logger32, import the most recent and complete ADIF backup.
   - Optionally import other ADIF backups as well, plugging any gaps in your consolidated log. Logger32 will identify and skip any *exact* duplicate QSOs that are already in the log[426], but will import any new/changed QSOs, bringing your log as complete and up to date as possible.
   - Check that the log appears intact. Depending on the source of the ADIF file/s, the restored QSOs may lack non-essential information such as names, locations, notes/comments and QSL status, but hopefully all the essentials are intact and correct.
   - Optionally, recalculate your statistics (probably not necessary but worth doing at some point, especially if your DXCC or other statistics don't look quite right).
6. Don't forget to set up an appropriate form of backups this time … or be prepared to suffer the next time.

> *Ever since upgrading to version 4, I have experienced sudden crashes of Logger32. I tried numerous things while attempting to pin down the cause. all to no avail. I am using Windows 10 Professional, and it was suggested that I try running Logger32 in Windows 7 Compatibility Mode. Sure enough, ever since making that change, Logger32 has been up and running non-stop for over three weeks.*
>
> **Tim W3YQ**

[426] Any *near*-duplicate QSOs are imported into your log *e.g.* a QSO logged at 17:21:01 is not the same as one logged at 17:21:00, even if the callsign, mode, frequency *etc.* are identical. If you are importing log backups from a system that does not faithfully record the seconds, expect to end up with a load of near-duplicate QSOs in your log. Take care! It helps to backup your log *before* each import, and check your log for near-dupes *after* each import: if needs be, you can restore a good backup to un-do later imports.

Copyright © 2026 Robert C. Furzer K4CY

Q. What does this error mean when Logger32 is closing?



A. *Something* (we don't know what, exactly) appears to have prevented Logger32 from replacing the previous user file backup (in this case) during its automated shutdown backups.

Here are some things to try, in sequence, having re-started Logger32:

1. Simply have another go: try again to generate a backup. Problems can be caused by mis-handled contention for PC resources *e.g.* when two programs attempt to access the same disk file at the very same time, and one of them melodramatically throws up an error rather than waiting politely in line. Such conflict is rare and is unlikely to recur, at least not on the very next attempt.
2. Click icon #1 to run the logbook backup function, or icon #2 to run the user files backup, but rename the backup file *before* you click <**Start**>. That should create a new backup file without problems.
3. See if you can open the backup file by navigating to *C:\Logger32* in Windows **E**xplorer, then double-clicking the backup zip file (the one named in the Zipping backup error message). It should open in Windows Explorer, WinZip or some other zip program, revealing a folder containing a bunch of Logger32 *.ini* configuration and data files. If it doesn't open, or if it is empty or corrupted, it may have been damaged by a hardware problem with the disk, a software problem with Logger32 or Windows or the zip utility, malware or cosmic rays. Oh oh. **Delete the damaged backup file and try once again to create a backup in Logger32**, while you still can.
4. *Temporarily* disable your antivirus software and try again to create a backup in Logger32. If that works OK, it seems your antivirus software has become paranoid.
5. Reboot and have another go. Yes, that's right: turn it off and on again.
6. Use the disk monitor tab in Windows Resource Monitor to figure out which programs are accessing the backup file when you try to create a backup, and seek technical support on those programs and/or from the Logger32 reflector.

Q. Errrr, what's all that about 'system restore points'?

A. System restore point is a built-in Windows utility that stashes away copies of certain files that are essential to the Windows operating system, enabling us to restore/recover them if something (usually a rogue patch) screwed them up - provided the stashed copies are still available (not also trashed *e.g.* by a disk failure) and uncorrupted (*e.g.* by ransomware).

Copyright © 2026 Robert C. Furzer K4CY

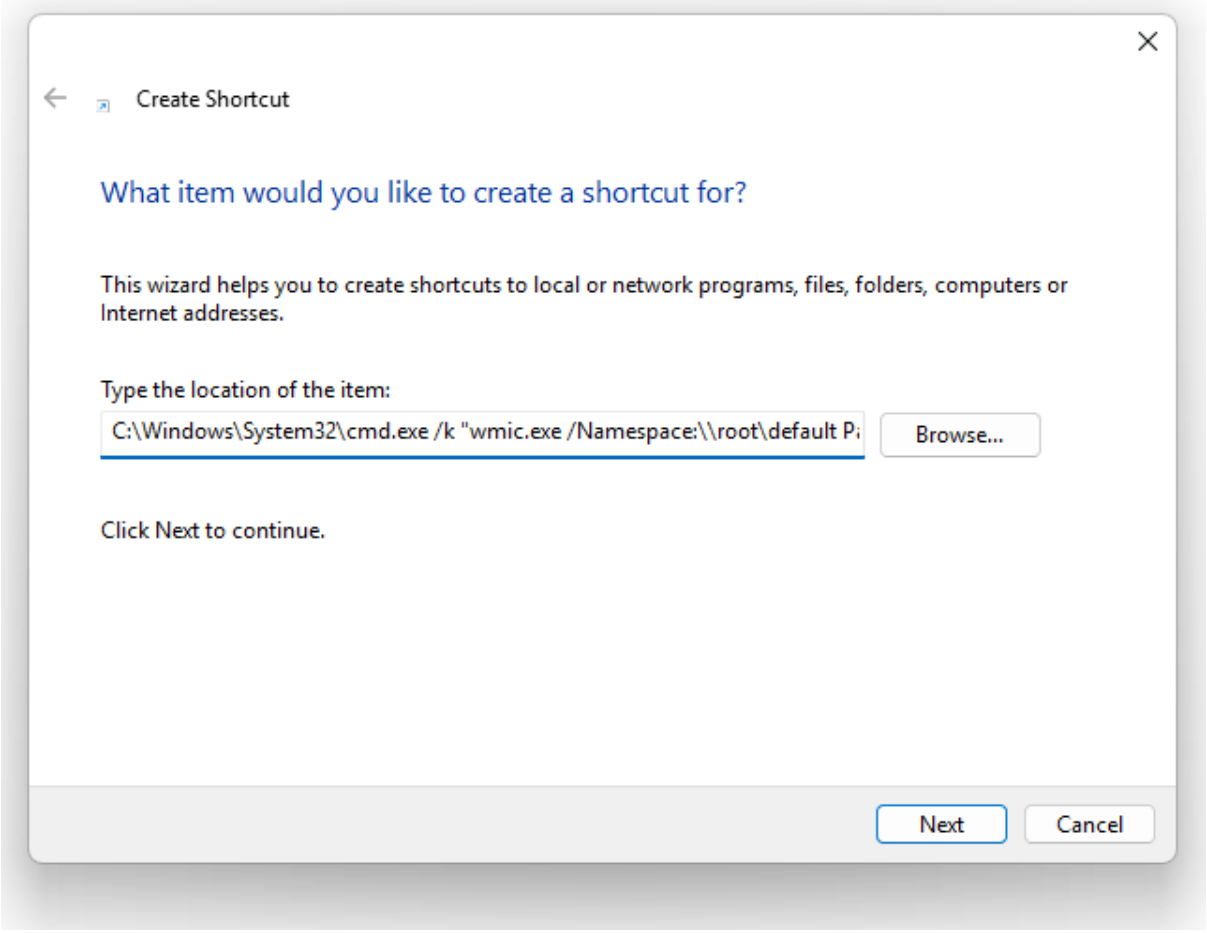
Assuming the Windows function has not been disabled and is working properly, a system restore point *should* be created automatically:

- Whenever Windows is about to patch itself;
- When we install application software, drivers *etc*. using the Microsoft Installer (which Logger32 uses);
- Weekly.

Hinson tip: to be sure, I have created a desktop shortcut to create a Windows system restore point manually and easily whenever I feel like it, which I do, roughly once a month. If that sounds like a good idea, read on for instructions …

1. Right-click your desktop, then click **New ⇨ Shortcut**.
2. Type or cut-n-paste the following command, complete with all the punctuation exactly as shown, into the new shortcut form box under "Type the location of the item:"

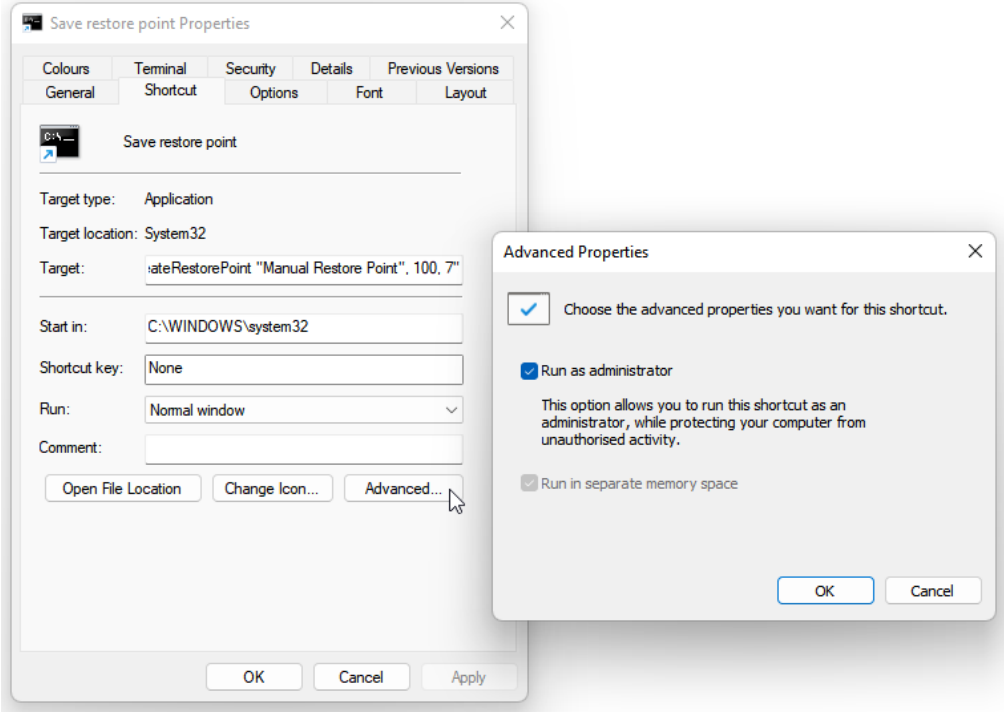
   C:\Windows\System32\cmd.exe /k "wmic.exe /Namespace:\\root\default Path SystemRestore Call CreateRestorePoint "Manual Restore Point", 100, 7"



3. Click <**Next**>, give the shortcut a name (*e.g*. "Create restore point") and click <**Finish**> to save it.
4. Right-click the shortcut icon and click <**Properties**>.
5. Click the <**Advanced**> button (on the shortcut tab).

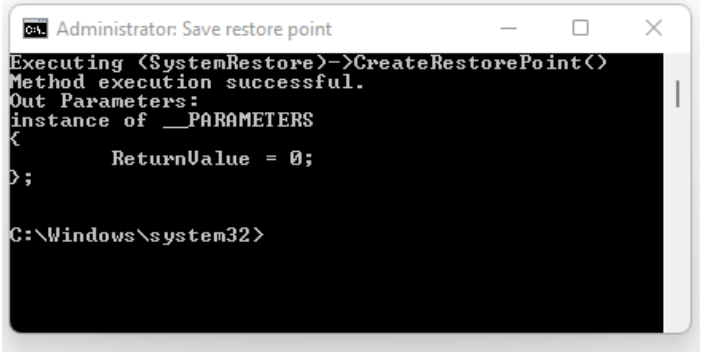
Copyright © 2026 Robert C. Furzer K4CY

6. Click to tick <**Run as administrator**> since creating restore points requires administrative privileges ▼



7. Click <**OK**> then <**OK**> again.
8. Double-click the shortcut for a test run.

The result a few seconds after double-clicking the desktop shortcut and agreeing to let Windows do some admin stuff, is this decidedly cryptic monochrome gobbledegook ▶



I am led to believe that "**ReturnValue = 0;**" is fluent Geek for *"Hey, that was a roaring success! Done! Fantastic! Restore point safely created and saved! Good going, eh? Congratulations! Take the rest of the day off: I'm going to put my feet up and rest my tired CPU for a few microseconds in celebration of a job well done! I'd gladly trot out for a beer with you only I appear to be short of a mouth. And beer. Oh and trotters."* However, the exact same message might also mean *"I honestly couldn't be bothered to create yet another restore point so soon after the last one. Don't worry, be happy. Chillax. It's all under control. Trust me, I'm Windows …"*

Copyright © 2026 Robert C. Furzer K4CY

Q. Logger32 is *the most important thing* on my PC. Nothing else matters. Can I make Logger32 the top priority in Windows?

A. Be careful what you wish for. Strictly speaking, the answer is yes, you can increase Logger32's normal priority to high or even realtime … but those options slow or stop other lower-priority activities. 'Other activities' are not just other apps (such as your web browser) but a slew of Windows activities quietly going about their business behind the scenes. Many of those background things are, in fact, rather important – such as watching for your keyboard and mouse activity, maintaining access and other security controls, and transferring data to/from the network and disk. Kinda important, I'd say …

> *"Real-time priority is really dangerous. It's higher priority than nearly everything else. It's higher priority than mouse input, keyboard input, and the disk cache.* ***If you foolishly set the priority class of a CPU-intensive program to real-time, it will suck up your entire processor, leaving no cycles for anything else. In particular, since not even input runs at real-time priority, you can't stop it via any interactive means,*** *because the thread that manages input can't even run to process your input. Mind you, even if the input thread did run at real-time priority, that wouldn't really help you any. Sure, it could receive your input and distribute it to the appropriate application queues, but those applications are themselves not running with real-time priority, so all that happens is that your input gets quickly transferred to the input queues, where it then sits waiting for the applications to process them (which will never happen since the applications are not running with high enough priority)."*
>
> ***When you set a 100% CPU program to real-time priority, you get what you asked for,***
> ***Raymond Chen***

Hinson tip: if your PC becomes *totally* unresponsive to the point that it even ignores <Ctrl+Alt+Del> or tapping the reset or power switch, there are two ways to regain control. First, press-*and-hold* the power switch. [Most] PCs respond by powering themselves off, a *very* quick but controlled power-down. The other method (switching off the PC's power cord at the wall socket or PSU switch) is basically a crash with a high risk of corrupting data. It is a risky absolute last resort. It's much safer to wait just a bit longer for the PC to respond, or try the press-*and-hold* method.

Copyright © 2026 Robert C. Furzer K4CY

Section D

Logger32 hardware interfacing

Copyright © 2026 Robert C. Furzer K4CY

35 Shared ports

> "Sharing is the bridge
> that connects us to others"
>
> *Melinda Gates*

Logger32 can use serial (COM *e.g.* RS232 or USB) ports to interface with a CAT-capable radio, rotator or TNC, to send CW, to control PTT and/or for FSK keying.

Logger32 can also use parallel (LPT, Centronics) ports for antenna switching, radio selection, CW transmission and PTT.

To accommodate limited hardware configurations with a shortage of available ports, some functions can be shared on the same ports but with these limitations:

- A TNC *must* be operated on a dedicated serial port;
- You may need additional programs such as VSPE (**V**irtual **S**erial **P**orts **E**mulator), and LP-Bridge or LPB2 (LP-Bridge 2) to share the serial ports for radios and other devices with other radio software.

To share PTT, CW and radio (CAT) control on the same serial port requires the use of TXd and RXd (transmitted and received data) plus the RTS *and* DTR modem control lines. You need to connect a homebrew or commercial PC-radio interface to the COM port (either a native RS232 port or a compatible USB/RS232 adapter) or LPT port, passing the CAT, PTT and CW keying through to the radio.

35.1 Port setup

The port setup for each function is explained in the respective chapter of this manual.

Even if serial ports can be shared among programs, the sharing is not always perfect. As each program does not know what others do, it may receive unexpected data caused by other programs. You may need to change parameters in Logger32 or in another program's setup. If you choose to use shared ports, check that shared functions are setup in a manner that will not cause conflicts.

The following serial port combinations are possible:

- CAT radio control, PTT and CW transmit.
- PTT and CW transmit.
- PTT, CW transmit and rotator control.

The only parallel port combinations possible are Antenna Switching, Selected Radio and/or CW transmit including PTT. See below for specific rules that must be followed if this configuration is used.

Copyright © 2026 Robert C. Furzer K4CY

35.1.1 Serial (COM) port configurations

These are the COM port signal/control lines that can be used by Logger32:

- **Radio control**: TxD and RxD (CAT commands sent to the radio and responses returned).
- **PTT**: RTS and/or DTR (asserted for transmit).
- **CW**: DTR for CW keying (asserted to key the carrier).
- **RTTY (FSK)**: TxD line is used to switch between space and mark tones.

35.1.2 Parallel (LPT) port configuration

These are the LPT port signal/control lines that can be used by Logger32:

- **Antenna switching**: any one of pins 2 through 9 is asserted to select antennas 1 through 8.
- **Radio select**: pin 14 is asserted to select radio 2.
- **PTT**: pin 16 is asserted to put the radio into transmit.
- **CW keying**: pin 17 is asserted to key the radio's carrier.

If the same parallel port is to be used for both antenna switching and the CW Machine, the same Hex port address must be configured in the CW Machine. You cannot share radio PTT on a serial port and CW PTT on a parallel port at the same time. **To avoid possible equipment damage, do not connect serial and parallel ports together.**

If the parallel port PTT is used for the CW Machine, the Sound card data window can control PTT by CAT radio command (*e.g.* if the radio has only one external PTT control point). Otherwise you could use a hardware interface (a transistor switch, opto-isolator or reed relay) for PTT.

35.1.3 PTT using a shared radio port

PTT can be controlled using either of the modem control pins (DTR or RTS) on a COM port while Logger32 communicates with the radio via CAT using the data pins (TXd and RXd) on the same COM port. In the Sound card data window, click **Settings ⇨ Radio PTT Options ⇨ PTT by Shared Radio Port** to set this up.

The Sound card data window can use either DTR or RTS for PTT. RTS is preferred because the CW Machine is hard-coded for PTT on RTS, with CW keying on DTR.

In the same way, CW keying can share the COM port used for CAT control of the radio, either using CAT commands to send CW (if available) or by keying the DTR line.

When configuring shared ports, be consistent to avoid conflicts between:

1. **Setup ⇨ Radio ⇨ Radio Port Configuration**
2. **Setup ⇨ Rotator**
3. **Setup ⇨ Antenna Selector**
4. **CW ⇨ Config ⇨ Keyer setup**
5. **Sound Card ⇨ Settings ⇨ Radio PTT Options**

Copyright © 2026 Robert C. Furzer K4CY

35.2 Using VSPE to share serial ports

There are two types of VSPE available, one for 32-bit OS (free!) and the other for 64-bit OS (not free!). VSPE can be downloaded from eterlogic.com/Products.VSPE.html

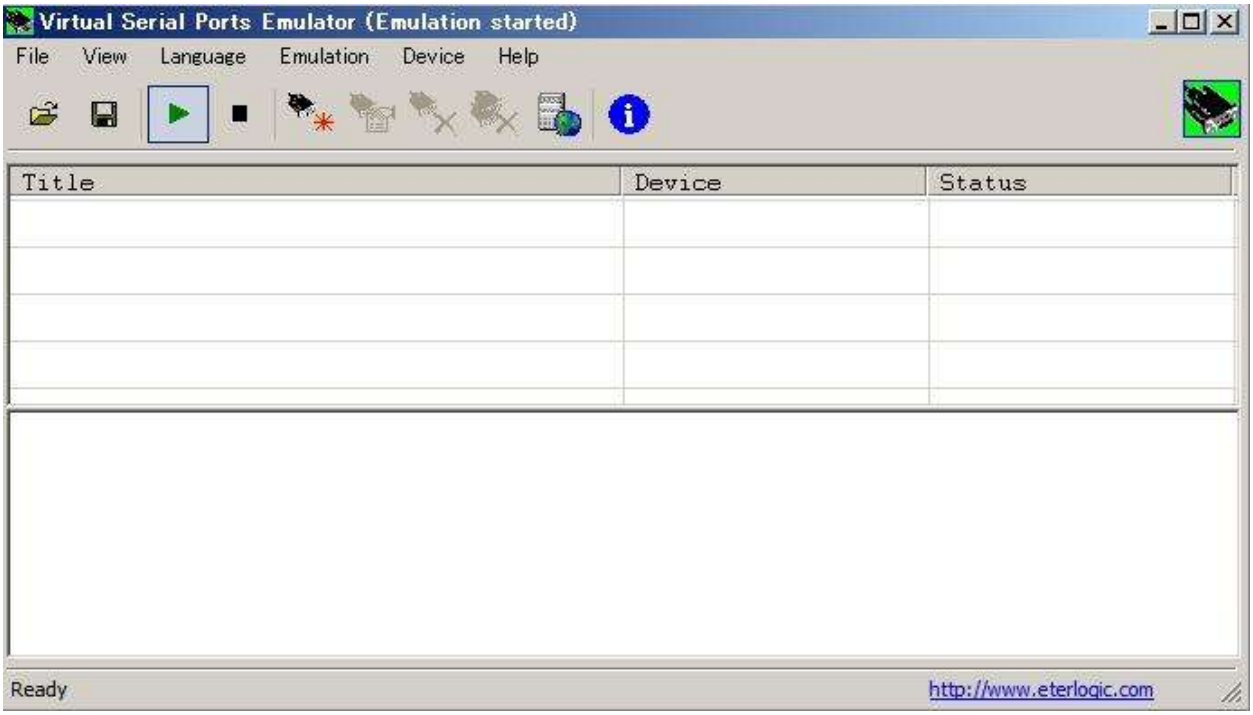
35.2.1 Setup VSPE

Although VSPE provides several ways to share serial ports, only Split Device and Connector Device are described in this section.

35.2.2 Setup VSPE Splitter Device

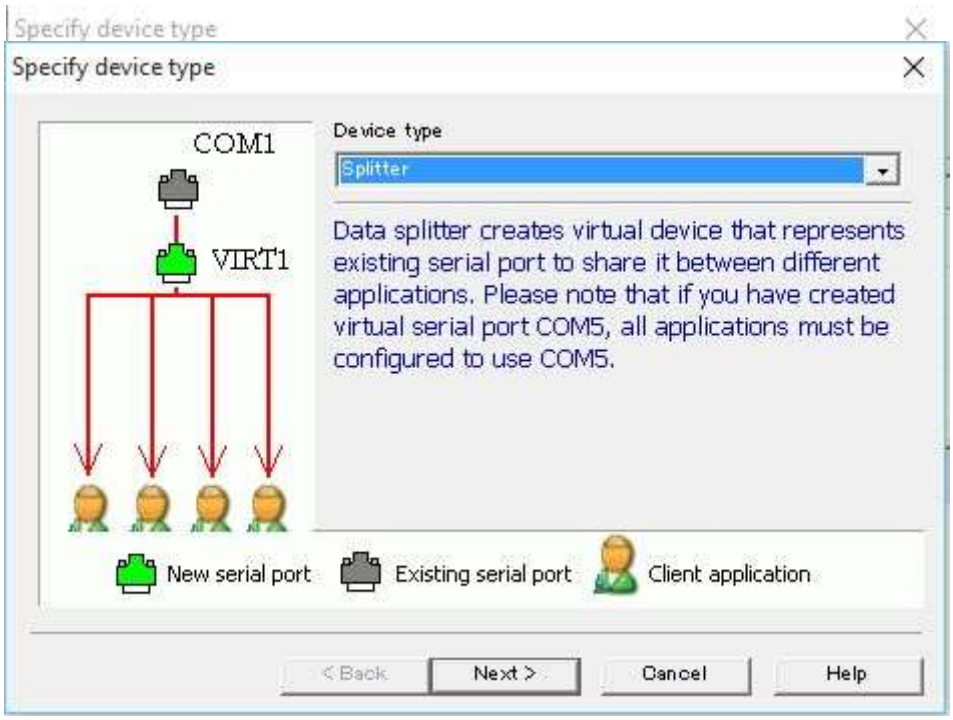
Splitter Device creates a virtual device that represents an existing serial port to share it between different apps.

Run VSPE ▼



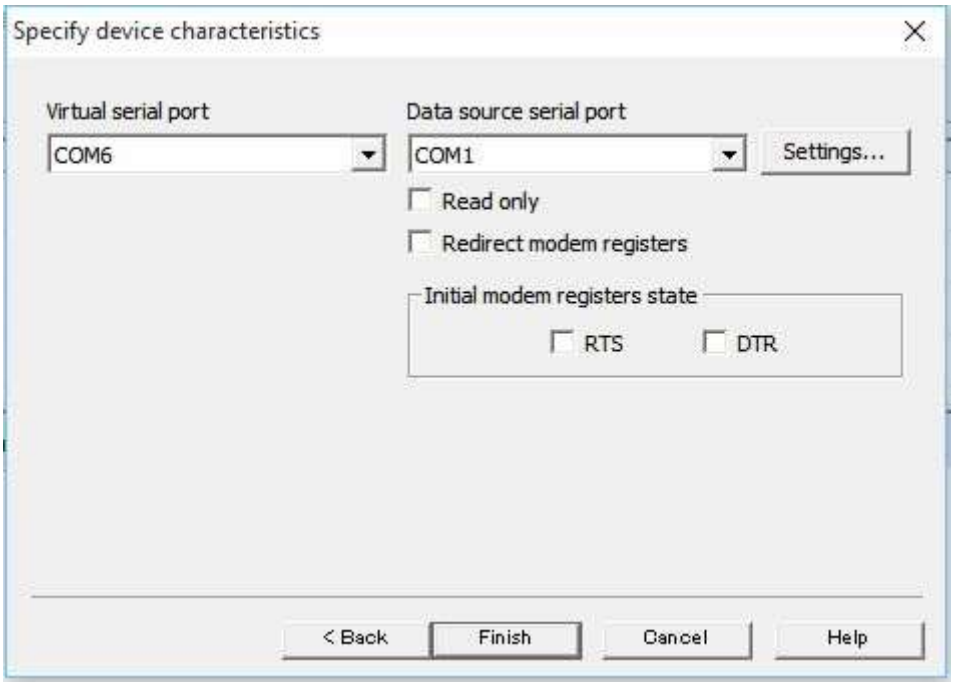
Copyright © 2026 Robert C. Furzer K4CY

Click **Device ⇨ Create ▼**



Select <**Splitter**> from the device type list ▼

Click <**Next**> ▼

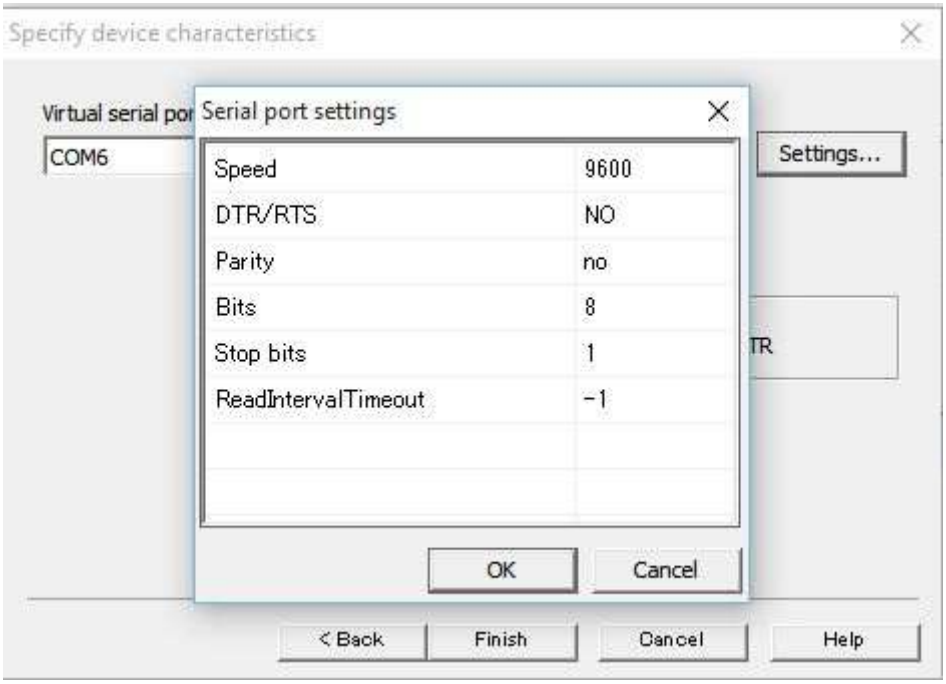


In this table:

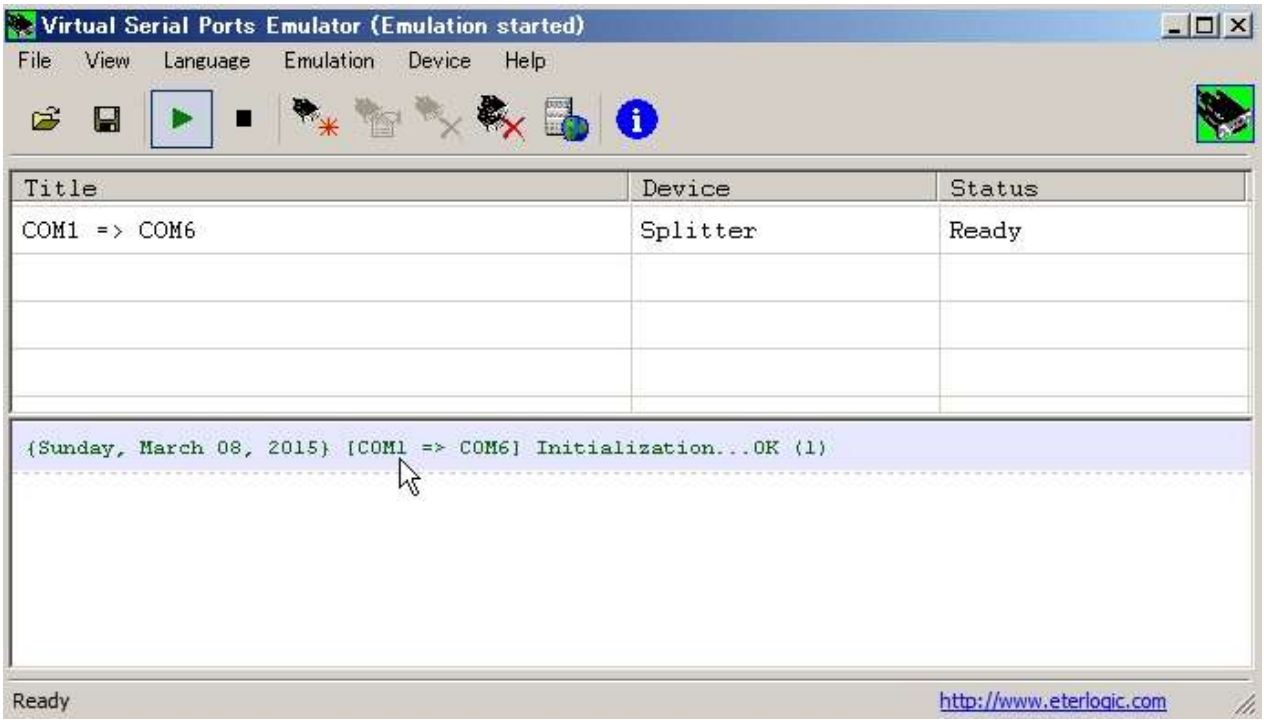
- Select COM for data source serial port. This is the serial port connected to your radio.

Copyright © 2026 Robert C. Furzer K4CY

- Select COM port for virtual serial port. This is a serial port which you will configure in Logger32 and other programs.
- Click Settings to configure baudrate *etc*. the same as the radio configuration.



Click <**OK**> then <**Finish**>.



Click **File** ⇨ **Save** as to save this setup. The next time you run VSPE, use **File** ⇨ **Open** to retrieve it.

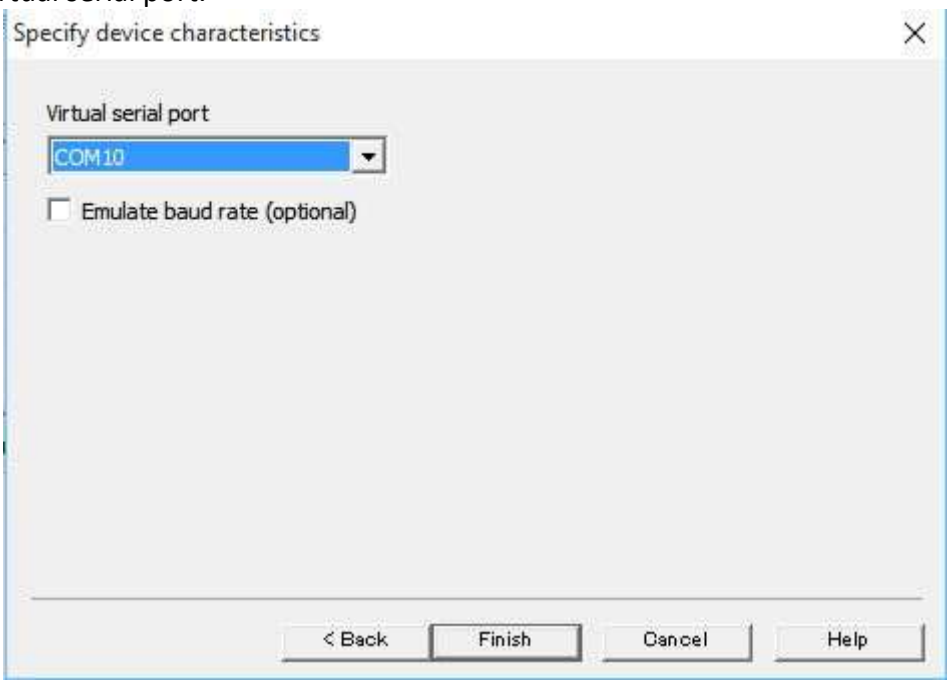
35.2.3 Setup VSPE Connector Device

Connector device is a virtual device that allows separate apps to share the same serial port.

- Run VSPE (see above)

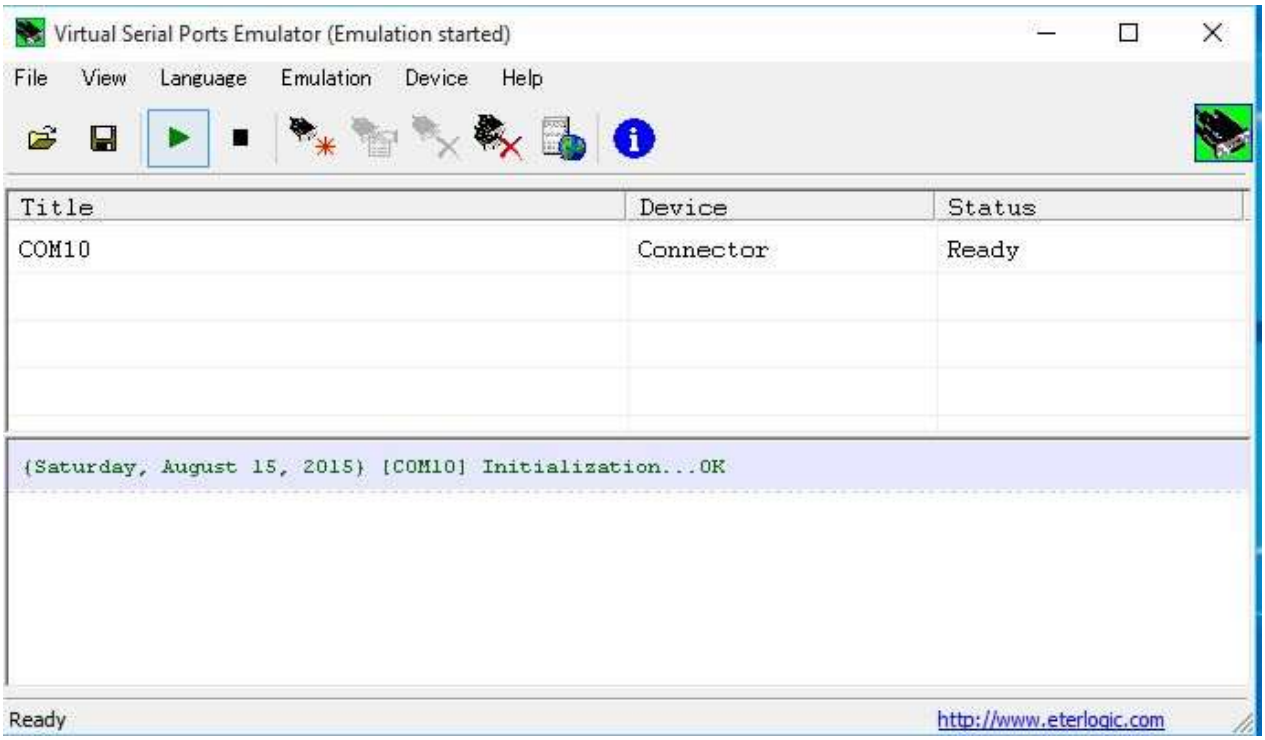
Copyright © 2026 Robert C. Furzer K4CY

- Select connector in the specify device type table (see above)
- Select virtual serial port.



Click <**Finish**>.

Click **File ⇨ Save** to save this setup. When you run VSPE next time, open the saved setup using **File ⇨ Open.**



Examples setups follow.

Copyright © 2026 Robert C. Furzer K4CY

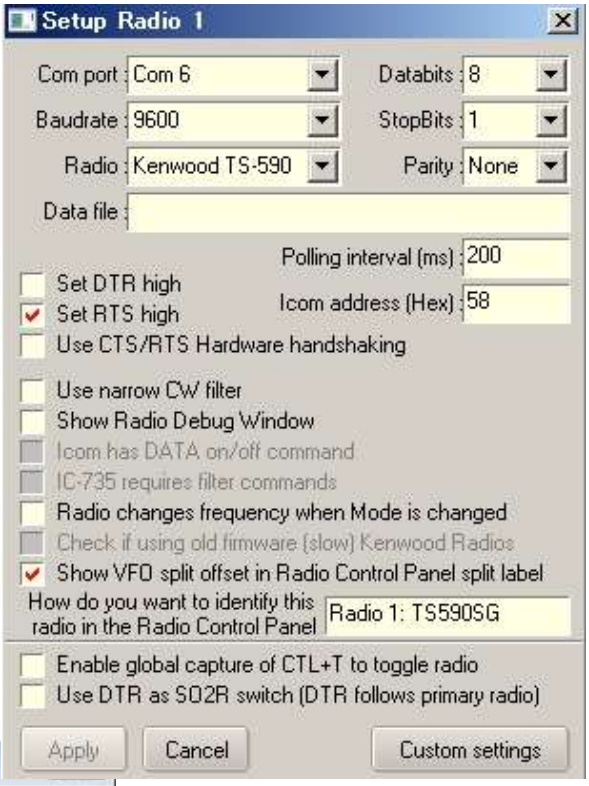
35.3 Logger32 and ARCP-590 or ARCP-590G (Splitter Device)

This section covers the graphical control program for TS-590 or TS-590G.

35.3.1 VSPE setup

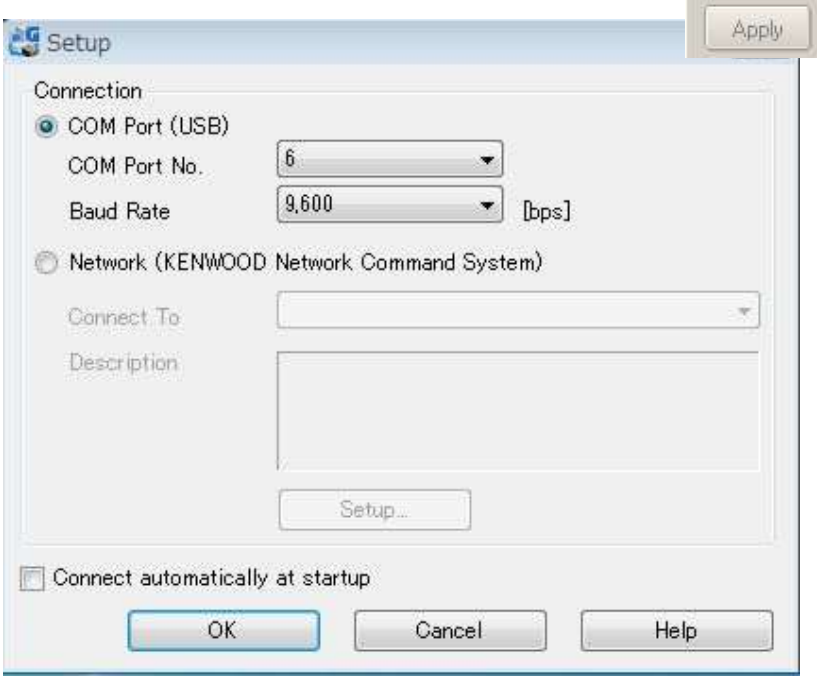
See above

35.3.2 Logger32 setup ▶



Note: the COM port and baudrate should be the same as setup in VSPE.

35.3.3 ARCP-590 or ARCP-590G setup ▼



Note: the COM port and baudrate should be the same in VSPE.

35.4 Logger32 and HDSDR (Splitter Device)

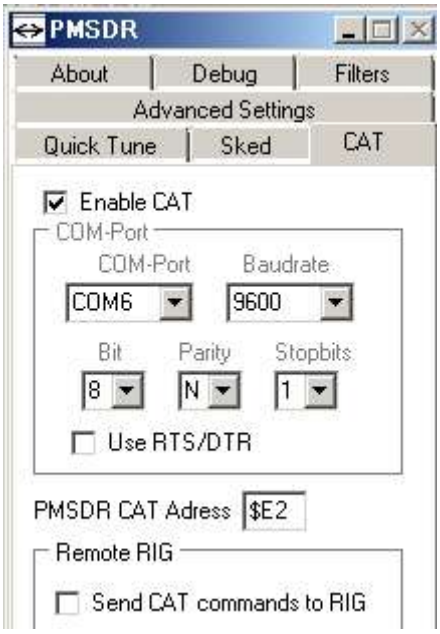
HDSDR is a panadapter program for the TS-590 or TS-590G.

35.4.1 VSPE and Logger32 setups

See above.

Copyright © 2026 Robert C. Furzer K4CY

35.4.2 HDSDR setup



This example uses a PMSDR SDR receiver.

Click <**ExtIO**> in the HDSDR window to open the PMSDR setup table.

Note: the COM port and Baudrate should be the same as in VSPE.

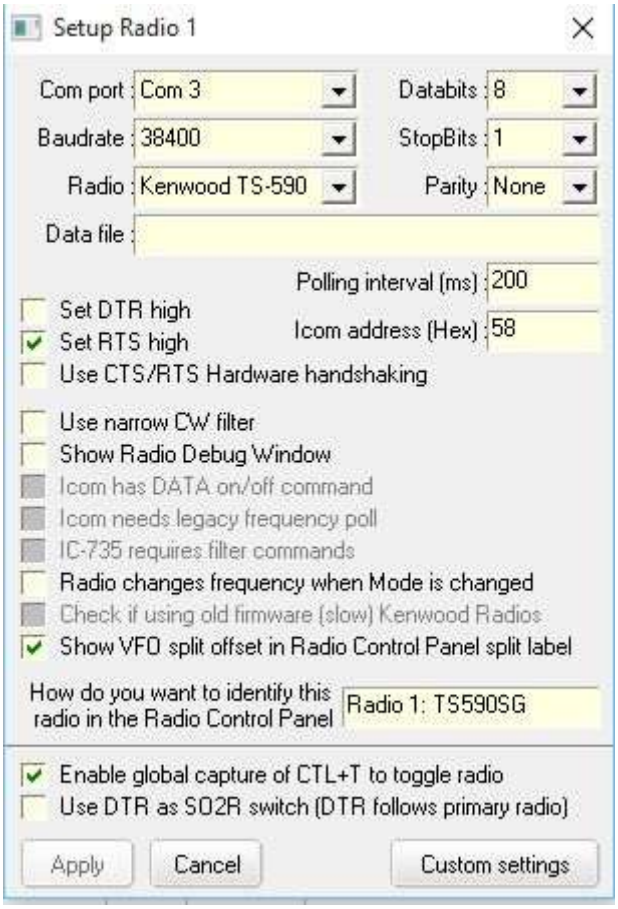
35.5 Logger32 and HDSDR (Connector Device)

In this case Logger32's echo port and HDSDR use the same serial port thanks to the VSPE Connector Device.

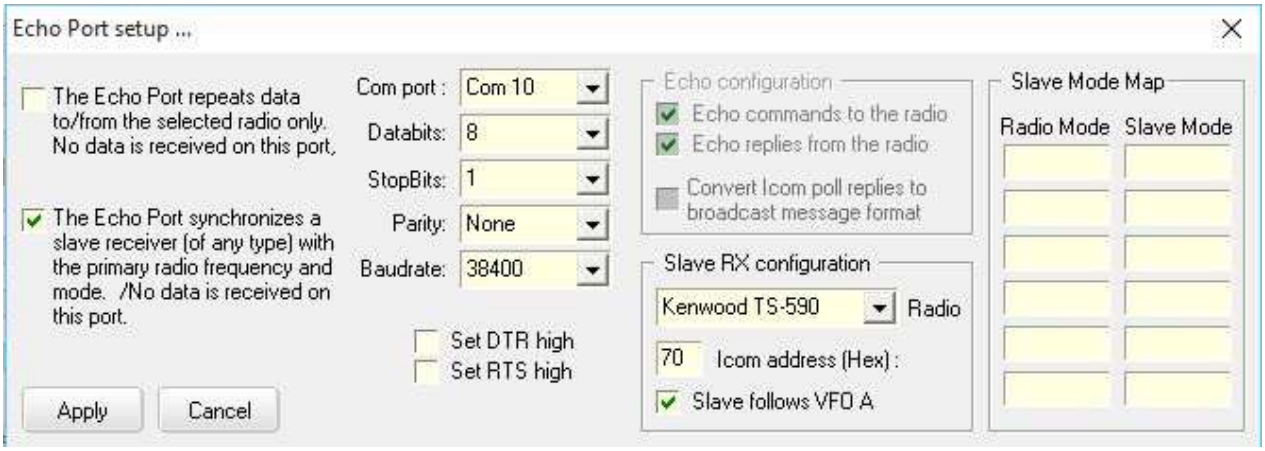
35.5.1 VSPE setup

See above

35.5.2 Logger32 setup



Copyright © 2026 Robert C. Furzer K4CY



Note: COM 10 is the Connector Device created by VSPE in this case. See echo port.

35.5.3 HDSDR program and PMSDR receiver setup



Note: COM 10 is the connector device created by VSPE. In this example HDSDR is tuned to the TS-590SG frequency while turning TS-590SG VFO knob or hitting DX spot.

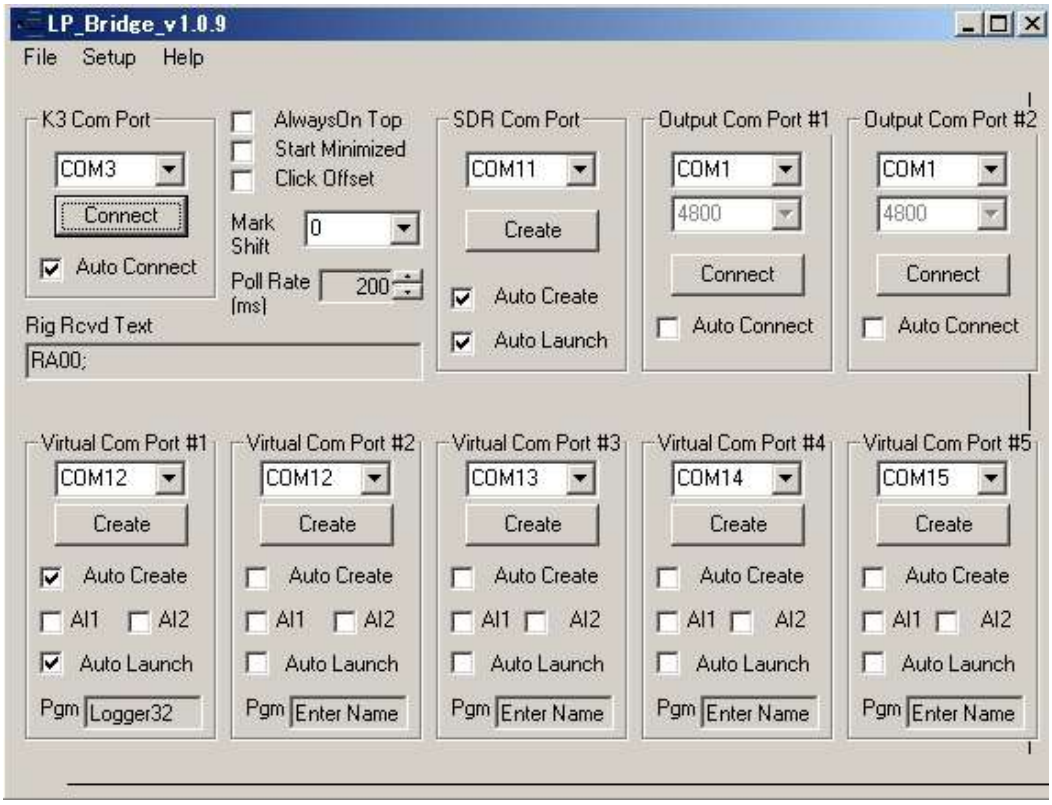
Copyright © 2026 Robert C. Furzer K4CY

35.6 LP-Bridge and LBP2

Both programs can be downloaded from www.telepostinc.com/LPB.html.

Officially, LP-Bridge only supports the Elecraft K3, however it also works with the KX3.

LBP2 supports the K2, K3, Yaesu (950,2000,5000) and Kenwood radios.



K3 Com Port: COM port which is connected to KX3. Click Connect button first. Auto Connect option is available.

SDR Com Port: COM port for NaP3 (This is a virtual serial port). Click Create button. Auto Create and Auto Launch option are available.

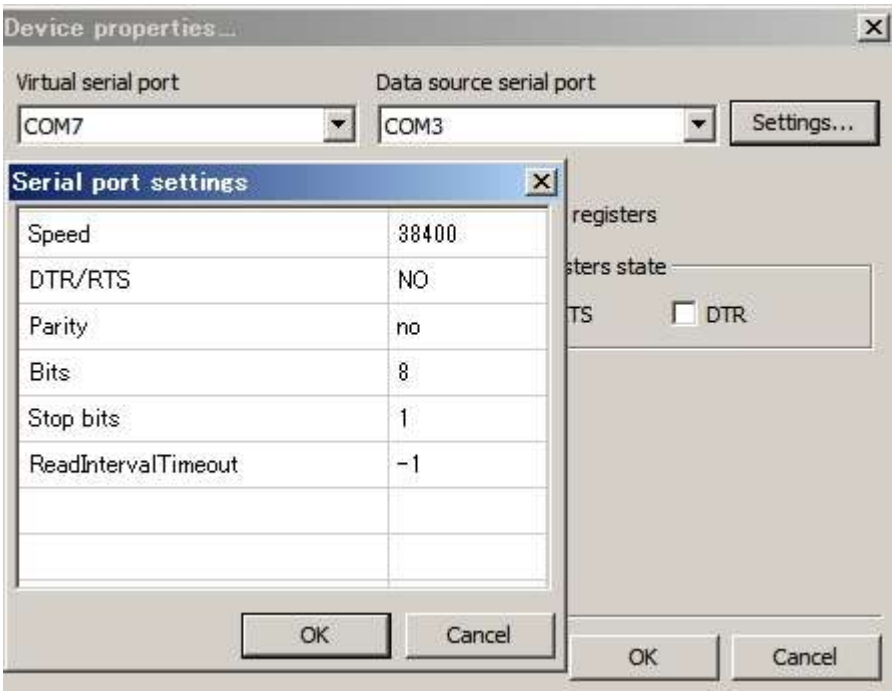
Virtual Com Port #1: COM port for Logger32. Click Create button. Auto Create and Auto Launch options are available.

35.7 Logger32 and NaP3 (Splitter Device)

NaP3 is a panadapter program for the Elecraft KX3 and K3.

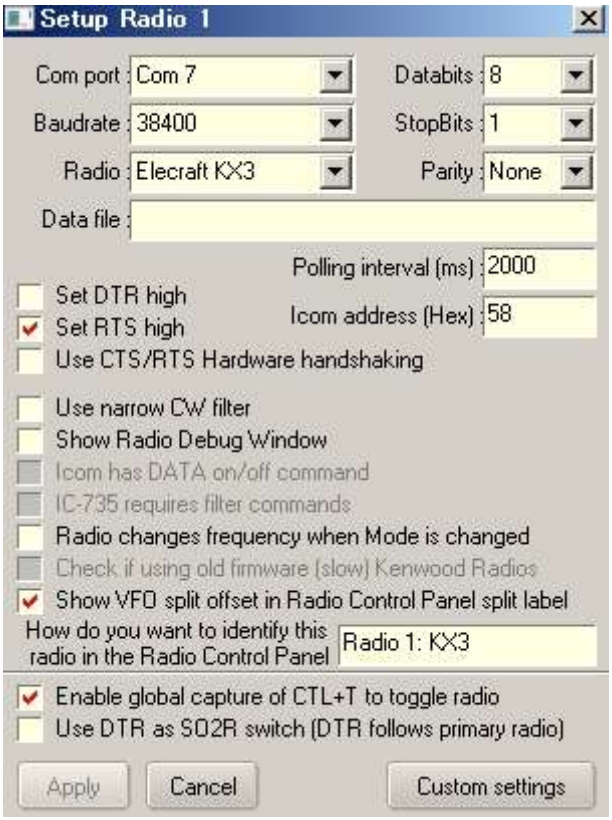
Copyright © 2026 Robert C. Furzer K4CY

35.7.1 VSPE setup



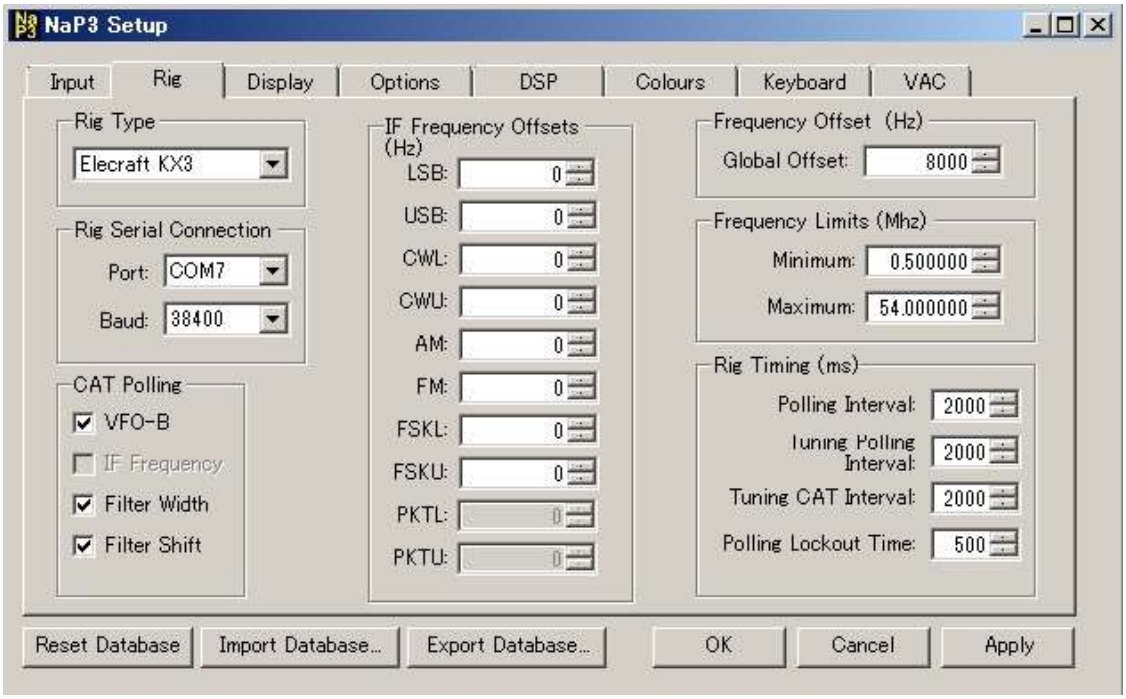
35.7.2 Logger32 setup

Note: the COM port and Baudrate should be the same as VSPE.



Copyright © 2026 Robert C. Furzer K4CY

35.7.3 NaP3 setup



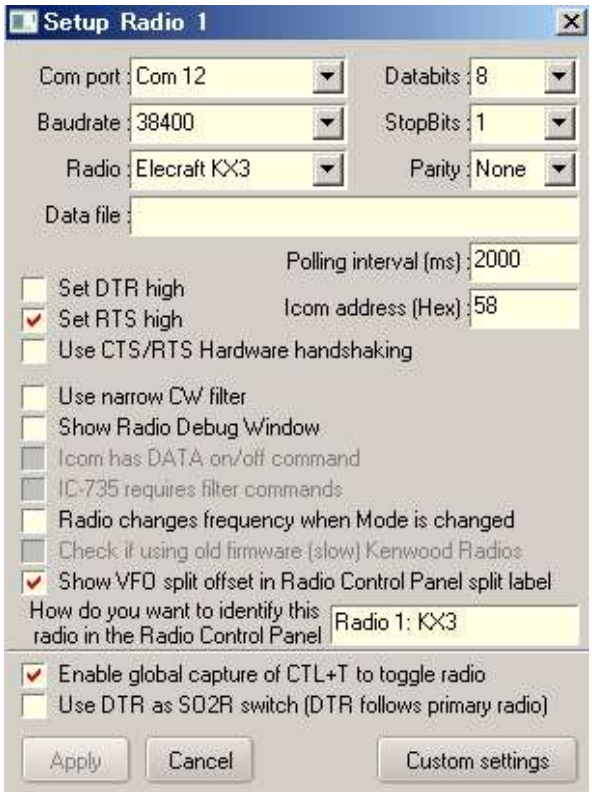
Note: COM port and Baudrate should be same as setup in VSPE.

35.8 Logger32, NaP3 and LP-Bridge

Setup and run LP-Bridge first.

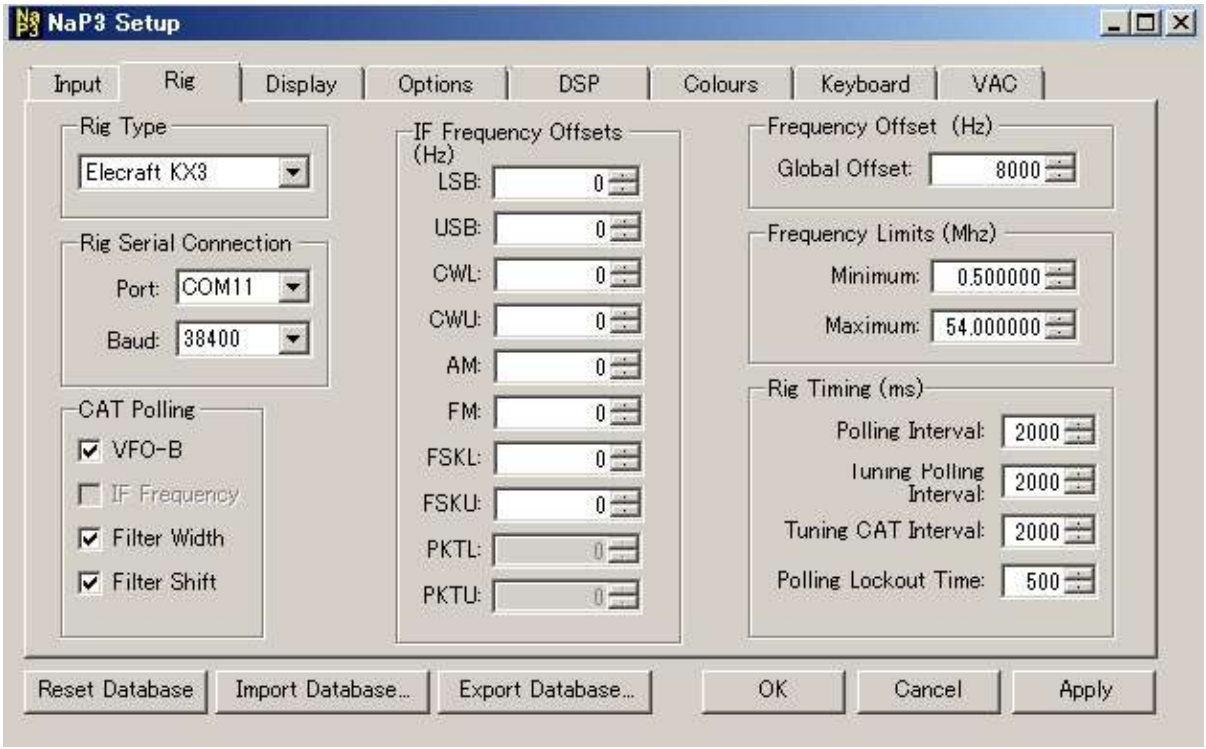
35.8.1 Logger32 setup

Note: use the same COM port and Baudrate as in LP-Bridge.



Copyright © 2026 Robert C. Furzer K4CY

35.8.2 NaP3 setup



Note: use the same COM port and baudrate as in LP-Bridge.

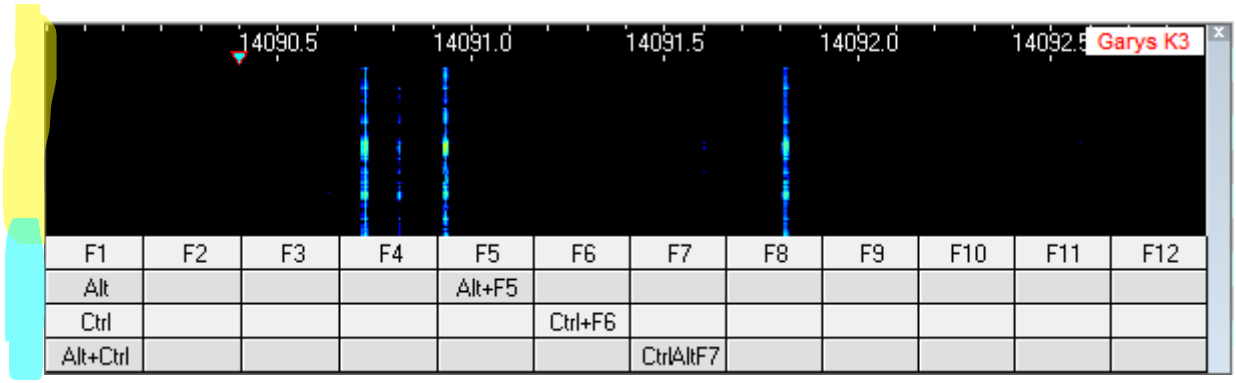
The <**Global Offset**> should be 8000 if KX3 is in IF 8 kHz mode, otherwise zero.

Copyright © 2026 Robert C. Furzer K4CY

36 Radio Control Panel (RCP)

> "Radio is the magic that fills the airwaves, transporting us to new worlds and experiences"
>
> *Bill Moyers*

The **R**adio **C**ontrol **P**anel ▼ is a compact pane to monitor the status of your CAT-connected radio and send it macro commands.



The RCP display has one or two sections:

- If shown, the optional upper section can display either a waterfall or spectrum representation of the received audio, and the VFO frequencies.
- The lower section is a table of up to 48 (1 to 4 rows of 12) user-programmable macro buttons[427].
- Click macro buttons on the RCP display to run the associated macros, or tap keyboard function keys F1 to F12 (optionally while holding Alt or Ctrl or Alt+Ctrl) to run the macros:
  - While the focus is on the log entry pane, *provided* the CW Machine Window, Sound card data window, Data Terminal and DVK windows are all closed (since, if they are open, the function keys trigger *their* macros instead).
  - While any other apps are active (in the foreground) on the PC, *if* the function keys are set in Logger32 for global capture - otherwise the function key presses are sent to those other apps, of course.

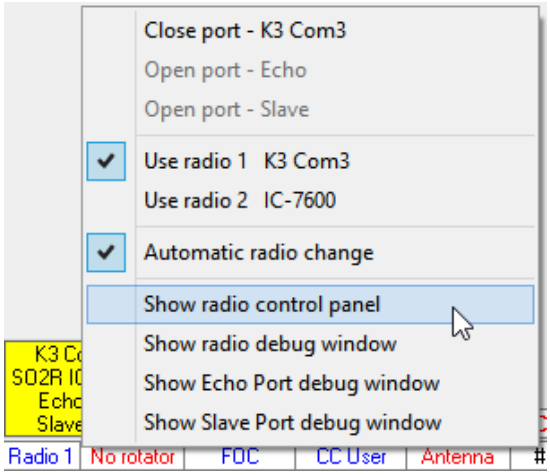
Hinson tip: even if you don't think the RCP is of interest, there are several split-related settings tucked away in here. Read on to find out about them.

[427] In the RCP image shown here, my RCP is displaying 48 macro buttons, some of which I have labelled with text such as F7 and Alt+Ctrl to remind me which hotkeys trigger the buttons, should I choose not to just click them and discover. By default, the buttons are all unlabeled, blank, boring, dull gray rectangles.

Copyright © 2026 Robert C. Furzer K4CY

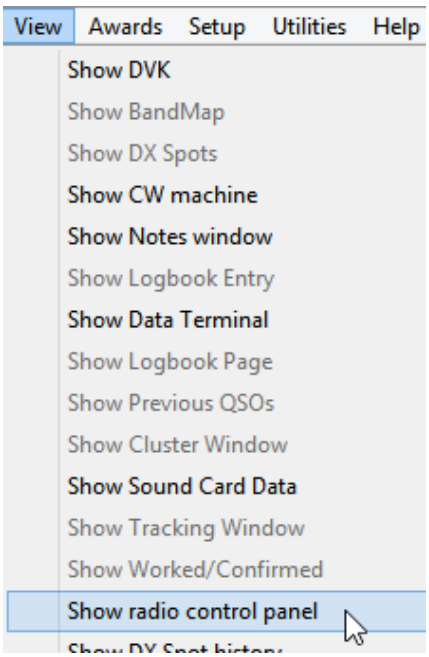
36.1 Open the RCP

Open the RCP either using
View ⇨ Show radio control panel ▶



- or -

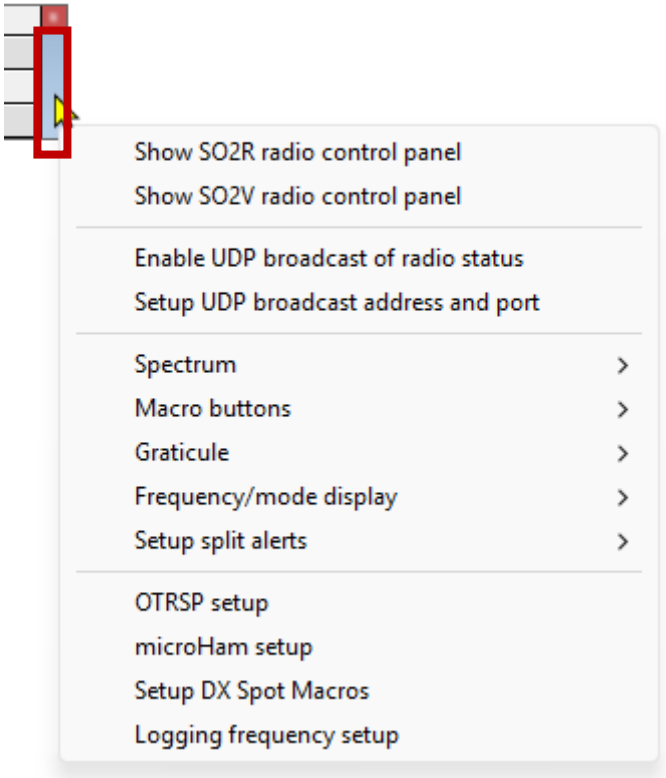
◀ right-click the <**Radio 1|2**> panel on the status bar, then click <**Show radio control panel**>



Hinson tip: despite the lack of a caption like most ordinary windows, you can reposition the RCP by clicking and dragging the blue-gray rectangle below the top-right corner  *← here!*

36.2 RCP menu

Right-click the RCP side bar (the blue-gray rectangle below the corner X) to open the RCP menu ▶



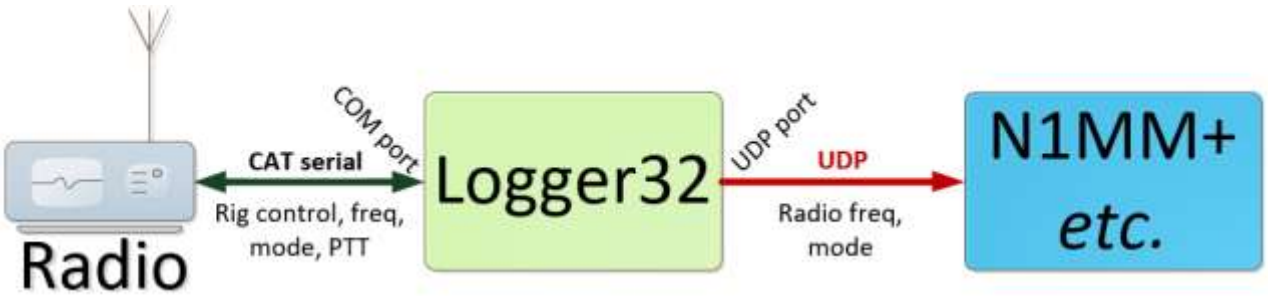
Copyright © 2026 Robert C. Furzer K4CY

36.2.1 Show SO2R|SO2V radio control panel

Additional RCPs can be opened to monitor and control your radio's sub-receiver (SO2V) or second radio (SO2R). The additional RCPs have their own configuration menus and macros. If an SO2R/SO2V window was open when the RCP was last closed, it opens automatically when you next open the RCP.

36.2.2 Enable UDP broadcast of radio status

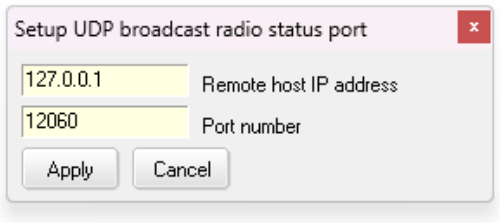
The RCP can **broadcast** CAT poll replies from the radio on a UDP port, allowing other software (such as N1MM+) to track/monitor the radio frequency, mode *etc*.



Logger32 retains full control of the radio through its CAT connection ▲: the RCP's UDP **broadcast** function simply *sends* radio status information to other software. Any radio commands *received* from the other software on the echo/rebroadcast UDP port (such as 'QSY to another contest band') are callously ignored by Logger32.

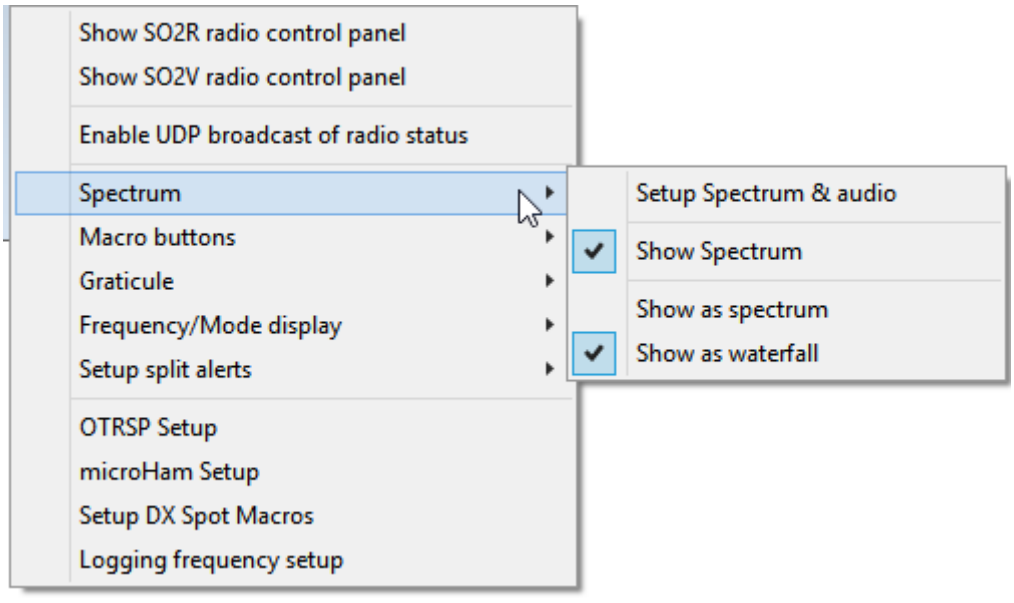
36.2.3 Setup UDP broadcast address and port

This is where we configure the UDP **broadcast** of CAT information received from the radio ▶



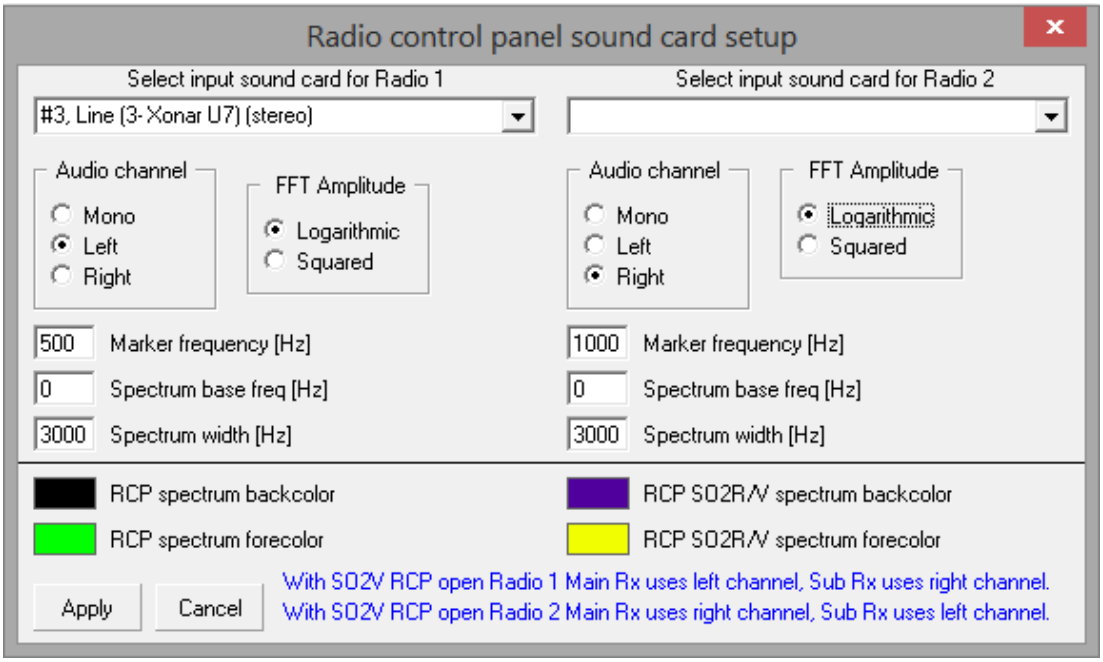
36.2.4 Spectrum

Click <**Spectrum**> on the RCP menu to open a submenu ▼



Copyright © 2026 Robert C. Furzer K4CY

- **Setup Spectrum & audio** lets you select the sound cards and audio channels to use for Radio 1 and Radio 2, set the frequency marker position, adjust the audio frequency range of the spectrum/waterfall display (if shown), set the **F**ast **F**ourier **T**ransform amplitude scaling, and alter the spectrum background and text colors ▼



The FFT Amplitude options affect the way that stronger/weaker signals are represented on the spectrum (the Y-scale) or waterfall (the Z-scale *i.e.* pixel intensity) displays: 'logarithmic' is similar to an S-meter scaled in dB.

The displayed audio spectrum can extend from 0 to 3000 Hz but your radio (and sound card!) may have limited bass response while the treble is limited by the receiver's bandwidth.

- **Show Spectrum**: tick this to display the receiver audio output either as a line graph plotting amplitude against audio frequency ("Spectrum") or a rolling raster display using rows of colored pixels of varying intensity reflecting the instantaneous amplitude at each frequency across the range (producing a cascading effect, hence "Waterfall"). *Un*tick it to reduce CPU load, giving a more compact display showing just the RCP macro buttons ▼

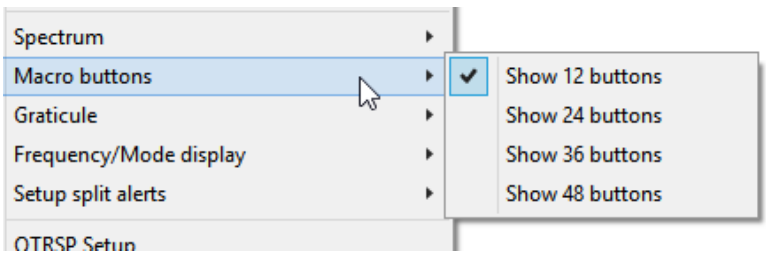
F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
Alt				Alt+F5							
Ctrl					Ctrl+F6						
Alt+Ctrl						AltCtrlF7					

- **Show as spectrum|waterfall**: tick to choose the spectrum or waterfall displays.

36.2.5 Macro buttons

Each RCP (the primary one plus those for SO2V or SO2R if used) has its own set of up to 48 macro buttons.

Choose between 1 and 4 rows of 12 macro buttons for each RCP ▶



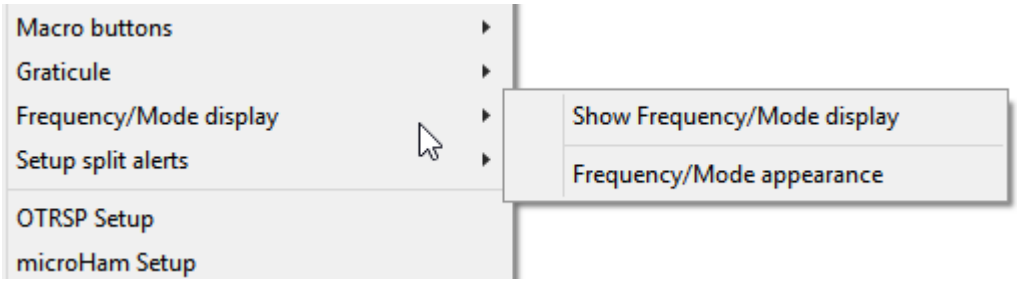
C:\Logger32\RadioPanel.ini stores all the RCP configuration data including the macros.

Copyright © 2026 Robert C. Furzer K4CY

36.2.6 Graticule

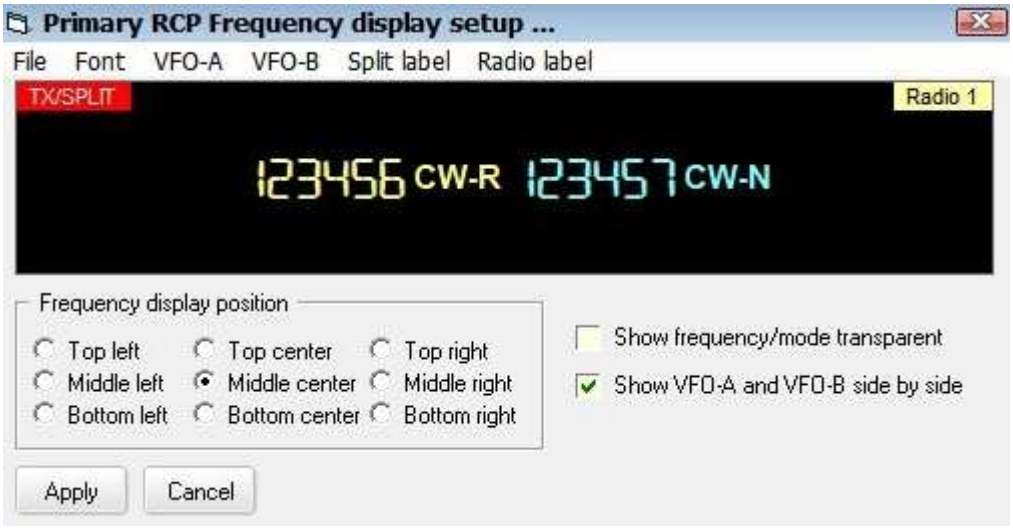
Choose either the radio's VFO (RF) frequency or the audio frequency on the scale above the spectrum or waterfall, if displayed by the RCP.

36.2.7 Frequency/Mode display



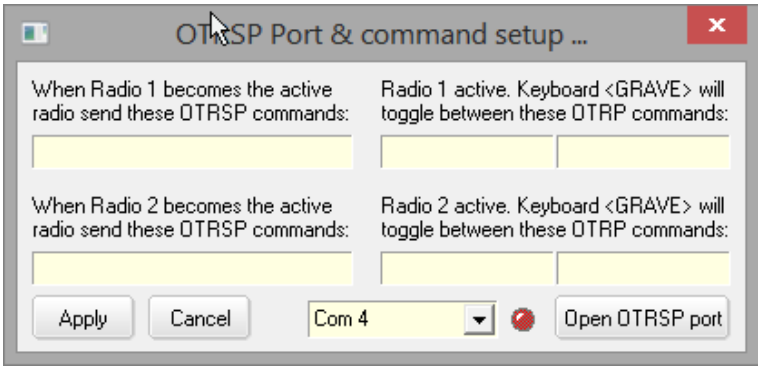
◀ The radio's VFO frequencies and modes can be shown in a text overlay on the RCP.

Using <**Frequency/Mode appearance**> on the submenu, you can replicate your radio's front panel LCD display on the RCP through judicious use of the options ▼



36.2.8 OTRSP Setup

<**OTRSP Setup**> on the RCP menu opens a form to configure radio-specific macros for the **O**pen **T**wo **R**adio **S**witching **P**rotocol for SO2R ▶



Custom OTRSP commands can be sent automatically when the active radio is switched using <**Ctrl**+**T**> or a mouse click. The format follows that of OTRSP macros *i.e.* THIS|AND|THAT.

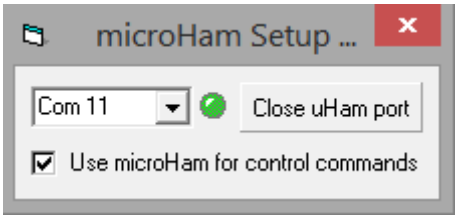
As with RCP's DX spot split and unsplit macros, this is a set-and-forget option. Once configured the way you like, the RCP need not remain open for the macros to function.

See the OTRSP chapter for more.

Copyright © 2026 Robert C. Furzer K4CY

36.2.9 *micro*Ham Setup

This form lets you define which serial port your *micro*HAM box is connected to, and optionally hands radio control to the *micro*HAM rather than Logger32 using CAT ▶

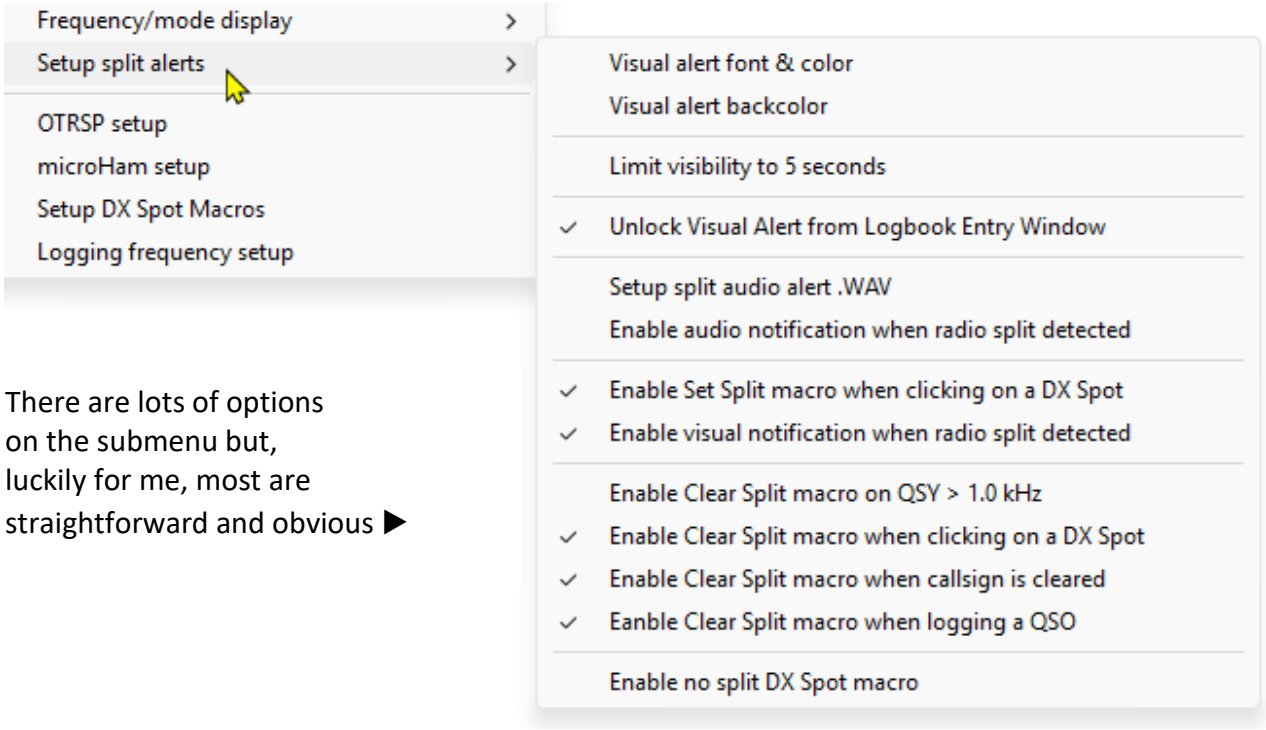


<**Use microHAM for control commands**> leaves the microHAM port open if it is being used by other functions, such as the antenna switch or DVK.

36.2.10 Setup split alerts and automatic split cancellation

You can configure audio-visual alerts to warn/remind you when your radio is in split mode *e.g.* after having:

- Clicked a DX spot with "UP 1" in the spot comments field.
- Clicked an RCP button to run a fancy radio macro that sets up the main and sub-receiver frequencies, modes, bandwidths and audio settings ready to operate split in a DX pileup.
- Reached out an arm across the shack desk to poke the split button on the radio's front panel[428].



There are lots of options on the submenu but, luckily for me, most are straightforward and obvious ▶

- **Visual alert font & color**: "OPERATING SPLIT" can use your choice of font and text color.
- **Visual alert backcolor**: to catch your eye, you might like to give the alert a day-glo bright background with a contrasting foreground (text) color.
- **Limit visibility to 5 seconds**: if that day-glo bright splash is burning your retinas, it can be made to disappear automatically after 5 seconds, hopefully before you become 'vision impaired'.

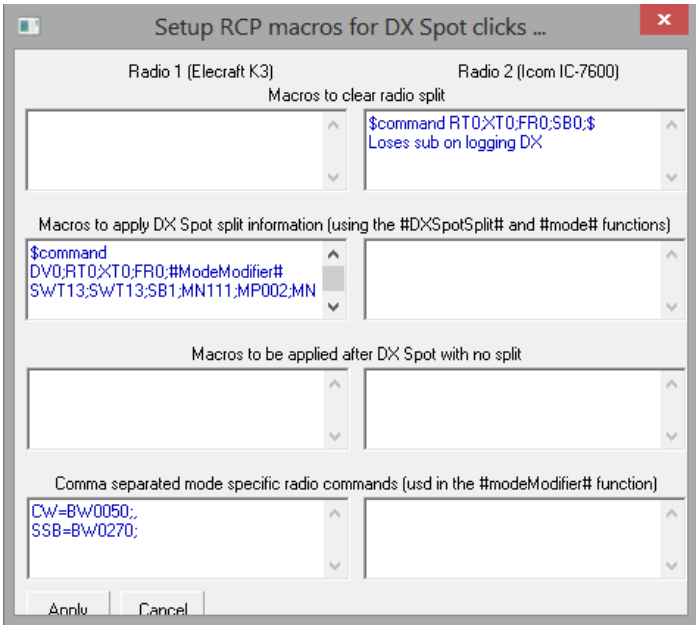
[428] Yes, the radio buttons still work! So does that big knob we used to use for tuning around the bands …

Copyright © 2026 Robert C. Furzer K4CY

- **Unlock Visual Alert from Logbook Entry Window**: usually, "OPERATING SPLIT" appears on the log entry pane, given that we are usually looking there while operating and logging QSOs. If you select this unlock option, you can drag the eye-watering "OPERATING SPLIT" message elsewhere: it is no longer confined to the log entry pane.
- **Setup split audio alert .WAV**: Logger32 can play an audio file when you enter split mode – possibly a subtle beep/chime or, for more impact, record yourself screaming "SPLEEEEEEET!".
- **Enable audio notification when radio split detected**: on the off chance that you haven't totally lost the use of your arm hence you do occasionally reach out to the radio, this option can sound the alert when you tap the SPLIT button on the front panel, assuming that maybe you didn't do it deliberately (in which case you presumably *know* that you did so!).

Then follows your choice of triggers to un-split the radio automatically under various circumstances:

- **Enable Clear Split Macro on QSY > 1.0 kHz**: if you are one of those strictly compliant hams in the habit of taking the DX station's "UP1" instruction literally, this option can automatically un-split your radio when you tune away by more than 1 kHz *e.g.* when you have worked or given up all hope of working the DX. If you are not so strictly compliant, you can fine-tune your transmitter within ±1 kHz as you try to find a clear calling frequency in the pileup … *but* if you move just past the 1 kHz threshold, your radio will instantly unsplit, meaning that you will probably now be transmitting on top of the DX station, creating a great gnashing of teeth among other hopeful callers and the inevitable rude and self-defeating chorus of "SPLEEEEET! SPLEEEET! SPLEEEEEEEEET YOU EEEDIOT! JEEEEEZ!" followed by uncomplimentary comments on DX cluster and sinister threats to you, your family and your dog. So be careful.
- **Enable Clear Split Macro when clicking on a DX spot**: Logger32 can automatically un-split your radio when, having been calling/working a DX station using split, you then click on a [different] DX spot. This is *much* less likely to cause grief and aggravation than the previous option. The very fact that you have clicked a different DX spot strongly suggests you are no longer trying to work the original DX station operating split.



- **Enable Clear Split Macro when callsign is cleared**: another auto-un-split function, this one is triggered by you clearing the DX callsign from the log entry pane, generally because you have worked and logged them (which clears the callsign field automatically) or given up and moved away in search of someone easier to work (hitting <**Ctrl**+**C**> to clear the callsign field manually). It is safer than the <**Clear on QSY**> option, but not quite as safe as <**Clear on clicking a spot**> since you may, perhaps, be using the log entry pane to look up the callsign of someone of interest heard in the pileup, which involves clearing the original DX callsign to do the lookup …
- **Enable Clear Split Macro when logging a QSO:** cancels split automatically when you hit <**Return**> or <**Ctrl**+**L**> to save the QSO from the log entry pane to your logbook. It should work nicely *provided* you never log a QSO with a DX station working split until the QSO is completed.

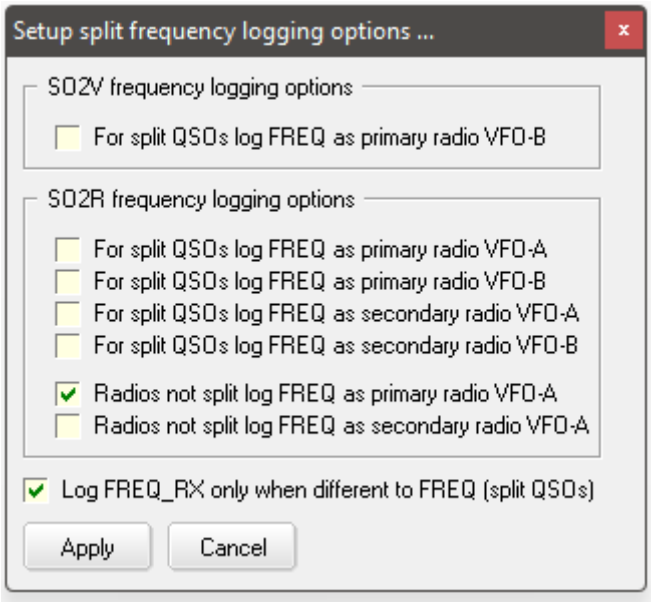
Copyright © 2026 Robert C. Furzer K4CY

> **Hinson tip**: take a brief moment to savor the QSO before logging it, just in case you need to send a parting comment to the DX station. If you transmit *after* the split has been cleared, you will surely provoke the wrath of the spleet poleece.
>
> **Supplementary tip:** depending on exactly what your Clear Split Macro does, you may be lucky enough to find your original transmit frequency is still in the other VFO. If needs be, you can simply re-enable split on the radio to transmit on your original split frequency.

- **Enable no split DX spot macro:** select this to execute your preprogrammed macro only if you clicked an ordinary simplex DX spot (with no QSX UP or similar comments). You might, for instance, like to configure your filters *etc*. differently for simplex and split spots.

36.2.11 Logging frequency setup

The ADIF field **FREQ** is intended to hold our transmit frequency, so we need to tell Logger32 which VFO on which radio we are transmitting on so that it can be logged correctly. There are several frequency-logging options when the radio is operating split ▼



- **For split QSOs log FREQ as primary radio VFO-B**: select this if you normally work split by listening on VFO A while transmitting on VFO B, the conventional type of split operating[429].
- **SO2R frequency logging options**: when operating split with two radios, choose which VFO on which radio to log as the transmit VFO. Alternatively, if we do not operate split (*e.g.* if we use one radio solely as a receiver to listen to the DX station, always transmitting on the other) we can log either radio's main VFO A as the transmit VFO.
- **Log FREQ_RX only when different to FREQ (split QSOs)**: selecting this simplifies the log a little. The frequency of the transmit VFO is always logged in the FREQ ADIF field. When operating split, the receive VFO frequency is *also* logged to the FREQ_RX ADIF field, otherwise it is unused and shows a blank cell in the logbook if this column is displayed ▼

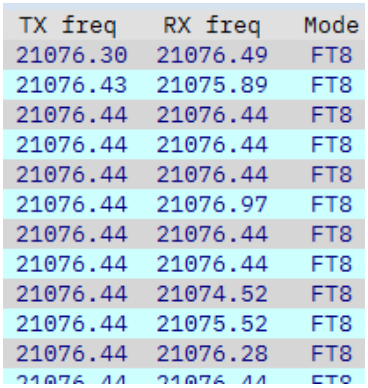
Date	Start	TX freq	RX freq	Mode
20 Jun 19	18:45	3573.59	3573.77	FT8
20 Jun 19	21:59	18130.50		CW

[I have changed the titles of my logbook columns – see here for instructions]

[429] Some of us with dual-receiver radios such as the K3 prefer to listen on the main receiver *i.e*. VFO A in one ear as we tune through the pileup using the big knob to find a suitable transmit frequency, while simultaneously listening to the DX station in both ears on the sub-receiver VFO B, leaving its little knob alone. This 'reverse split' configuration means VFO A *always* controls our transmit frequency, whether we are working split or not. The radio itself is *never* in split mode, with the advantage that we are less likely to turn on split accidentally and so transmit on top of the DX.

Copyright © 2026 Robert C. Furzer K4CY

Since operating split (audio offset) is the norm for FT# and similar digimodes, WSJT-X and JTDX routinely pass *both* the FREQ *and* RX_FREQ values to Logger32 via UDP (if so configured), so both values are logged accordingly – even when they are the same ▶



TX freq	RX freq	Mode
21076.30	21076.49	FT8
21076.43	21075.89	FT8
21076.44	21076.44	FT8
21076.44	21076.44	FT8
21076.44	21076.44	FT8
21076.44	21076.97	FT8
21076.44	21076.44	FT8
21076.44	21076.44	FT8
21076.44	21074.52	FT8
21076.44	21075.52	FT8
21076.44	21076.28	FT8

36.3 RCP (radio) macros

RCP macro buttons are intended to send CAT commands and parameters to the radio in use, hence the $**Command**$ and $**HexCommand**$ macros are the main ones here. Macros designed for the CW Machine window and Sound card data window do not work in the RCP except for those detailed below.

Setting up the RCP macro buttons is described in this chapter. Up to four rows of 12 macro buttons can be defined, including another set for the second radio if you are using two radios.

36.3.1 RCP functions

These are remarkably similar to Logger32's macros, except the key words are enclosed in hashes rather than dollars, allowing radio commands to be embedded *within* macro scripts.

Function	Description
#Call#	Replaced by the callsign of the station you are logging (the one currently in the Call field of the log entry pane).
#DXSpotSplit#	Applies the split noted in a DX spot's comments *e.g.* "Up 1".
#Greeting#	Send a greeting appropriate to the local time of person you are contacting. If Logger32 cannot determine their local time, the default greeting is sent. See Macros for information on setting up the default greetings. This function is only available for radios that permit CW keying by CAT commands. The syntax is like: $**Command** BLAH **#Greeting#** BLAH$$**LogImmediate**$
#Keyboard#	This function, imbedded in a macro, lets you increment the selected VFO by the number of kHz you type in *e.g.* $**HexCommand FE FE 04 E0 05 #Keyboard# FD**$ increments the IC-735's active VFO by as many kHz as you enter (up to 99). Type the number of kilo Hertz as two digits *e.g.* 03 for 3 kHz. Plus (QSY HF) is the default. Decrement the VFO (QSY LF) by preceding the number with a minus, hyphen or dash (*e.g.* -03 for 3 kHz down). Whereas there is normally a 5-second window to enter the value, as soon as 2 digits (*e.g.* 05 or 10) have been entered, the macro executes immediately.

Copyright © 2026 Robert C. Furzer K4CY

Function	Description
#Mode#	When using any of the **#Split#**-type functions, it is important to set the same mode for both VFOs. **#Mode#** is a single digit function. Where the radio set mode requires two digits, configure the macro to include the leading "0" or other digit as required *e.g.* the Yaesu FT-920 sets VFO A with: 00, 01, 02, 03 *etc*. and VFO B with: 80, 81, 82, 83 *etc*. A macro to set the VFO B mode must have a leading "8" *e.g*. $**HexCommand 00 00 00 8#Mode# 0C**$ The CAT command A>B generally changes the VFO B frequency and mode to that used by VFO A. If there is no such command on your radio, the **#Mode#** function does the same thing. In the case of the TS-850 for example one could use this function: $**Command FR1;**$ Swap to VFO B $**Command MD#Mode#;**$ Make the mode the same as VFO A A macro using this function must swap VFOs both before *and* after.
#ModeModifier#	Sends a bandwidth command to the radio when a split command changes the operational mode. The syntax is similar to the **#Split#** function: $**Command xx xx #ModeModifier# xx**$
#MyCall#	Inserts the current operator *i.e.* the callsign currently being used on-air to identify this station. This function is only available for radios that accept keying information when in CW mode.
#Name#	Inserts the contents of the Name field from the log entry pane. This function is only available for radios that accept keying information when in CW mode.
#SentRST#	Inserts the contents of the RST_SENT field from the log entry pane. If the field is empty, it defaults to 599.
#Split*N*#	A clever feature specifically for the RCP is a **#Split#** function for use within macros. When embedded in a macro string, it inserts a frequency based on the current VFO frequency, formatting that frequency for the command and radio in use. ***N*** is a frequency offset in kHz. For example, to shift the VFO frequency HF by 10 kHz, use **#Split+10#**. To shift the VFO down 5 kHz, use **#Split-5#**. The frequency string that the function generates is formatted for a $**Command xxxx**$ macro specific to the radio in use. The frequency string the function generates is always in kHz regardless of the radio type.

Copyright © 2026 Robert C. Furzer K4CY

Function	Description
#Splitxxxh#[430]	The "**h**" specifies that the **#split#** function is embedded within a $**HexCommand xxxx**$ macro. The function generates a hexadecimal frequency string, formatted for the particular radio in use.
#Splitxxxq#	Use this for a **#split#** function embedded in a $**qsy xxxx**$ macro.
#Spleeetxxx#	As part of the ongoing multi-lingual nature of Logger32, **#Spleeetxxx#** is interchangeable with the above listed **#Splitxxx#** commands.
#Wait n#	$**Command #Wait n# xyz;**$ or $**HexCommand #Wait n# xx yy zz**$ will pause for n seconds before executing the command xyz or hexadecimal command xx yy zz. Examples: $Command **do this first;**$ $Command **#wait xx# do this next after waiting xx seconds;**$ A space is optional, so **#Wait5#** and **#Wait 5#** both work.

36.3.2 Examples of simple macros

Also see the CAT chapter.

- ICOM example macros:
  - This macro changes an ICOM VFO to 14123.456:

    $**HexCommand FE FE 74 E0 05 56 34 12 14 FD**$.

    The red numbers are the frequency formatted in hexadecimal pairs reading right-to-left as required for the ICOM.
  - A macro to move the VFO 10 kHz HF:

    $**HexCommand FE FE 74 E0 05 #Split+10h#FD**$

    The **split** function shown in red reads the current VFO frequency, adds 10 kHz to it, and converts the result to hex pairs in the usual right-to-left ICOM sequence. In this example, the output of the **#Split#** function (34 12 01 14) is then inserted into the macro in its place, giving the resulting macro: $**HexCommand FE FE 74 E0 05 34 12 01 14 FD**$ … which moves the VFO to 14011.234 (10 kHz HF of where it was, in this case 14001.234).

36.3.3 Radios that convert text in CAT commands to CW

In addition to commands and functions effecting the frequency and radio selection control, a few functions are available for those radios that will accept text via the CAT interface and send it as CW *e.g.*:

$**[Hex]Command TXON; SCW;#Call# DE #MyCall# HI #Name# UR 599 OK bk;**$

$**[Hex]Command TXOFF;**$

[430] The **#Splitxxx#** function has been simplified. The **#Splitxxxq#** and **#Splitxxxh#** functions no longer *require* the "q" or "h", but if present, they still work as designed.

Copyright © 2026 Robert C. Furzer K4CY

where:

TXON is the CAT command to put the radio into transmit.

SCW is the command to send CW.

TXOFF returns the radio to receive.

#Call# and **#Name#** are replaced by data from the log entry pane (if it is open) and **#MyCall#** is replaced by the current operator.

An example for the K3/KX3:

$**Command KY #Call# Hello #Name#;**$

- or -

$**Command KY 73 #MyCall# *;**$

36.3.4 Simple macros for the IC-7600 and IC-756 Pro III together with a suggested use in the RCP

Function	ICOM IC-7600 (hex address 7A)	ICOM IC-756 Pro III (hex address 6E)
Mute	$**HexCommand** **FE FE 7A E0 14 01 00 FD**$	$**HexCommand** **FE FE 6E E0 14 01 00 FD**$
Set AF volume 30	$**HexCommand** **FE FE 7A E0 14 01 30 FD**$	$**HexCommand** **FE FE 6E E0 14 01 30 FD**$
Set AF volume 50	$**HexCommand** **FE FE 7A E0 14 01 50 FD**$	$**HexCommand** **FE FE 6E E0 14 01 50 FD**$
USB	$**HexCommand FE FE 7A E0 06 01 FD**$	$**HexCommand FE FE 6E E0 06 01 FD**$
LSB	$**HexCommand FE FE 7A E0 06 00 FD**$	$**HexCommand FE FE 6E E0 06 00 FD**$
RTTY	$**HexCommand FE FE 7A E0 06 04 FD**$	$**HexCommand FE FE 6E E0 06 04 FD**$
CW	$**HexCommand FE FE 7A E0 06 03 FD**$	$**HexCommand FE FE 6E E0 06 03 FD**$
Split on	$**HexCommand FE FE 7A E0 0F 01 FD**$	$**HexCommand FE FE 6E E0 0F 01 FD**$
Split off	$**HexCommand FE FE 7A E0 0F 00 FD**$	$**HexCommand FE FE 6E E0 0F 00 FD**$
A=B	$**HexCommand FE FE 7A E0 07 B1 FD**$	$**HexCommand FE FE 6E E0 07 B1 FD**$
Swap A/B	$**HexCommand FE FE 7A E0 07 B0 FD**$	$**HexCommand FE FE 6E E0 07 B0 FD**$
QSY to 14195 kHz on SSB	$**QSY 14195**$ $**HexCommand** **FE FE 7A E0 06 01 FD**$	$**QSY 14195**$ $**HexCommand** **FE FE 6E E0 06 01 FD**$
Split up 5	$**QSY #Split+05q#**$	$**QSY #Split+05q#**$
Split down 10	$**QSY #Split-10q#**$	$**QSY #split-10q#**$
NB on	$**HexCommand** **FE FE 7A E0 16 22 01 FD**$	$**HexCommand** **FE FE 6E E0 16 22 01 FD**$

Copyright © 2026 Robert C. Furzer K4CY

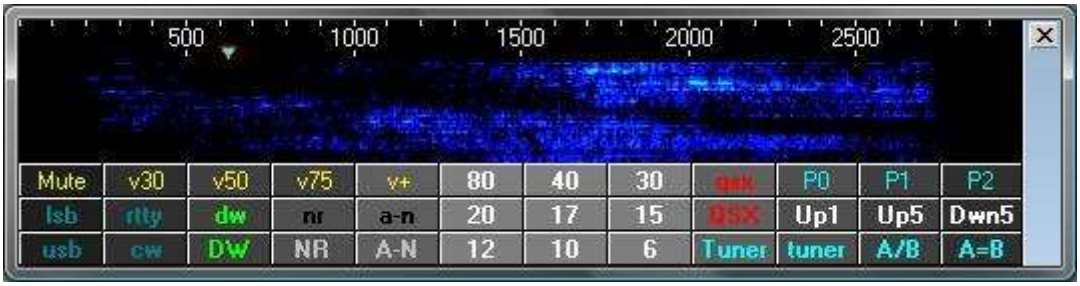
Function	ICOM IC-7600 (hex address 7A)	ICOM IC-756 Pro III (hex address 6E)
NB off	$HexCommand FE FE 7A E0 16 22 00 FD$	$HexCommand FE FE 6E E0 16 22 00 FD$
Dual-watch on	$HexCommand FE FE 7A E0 07 C1 FD$	$HexCommand FE FE 6E E0 07 C1 FD$
Dual-watch off	$HexCommand FE FE 7A E0 07 C0 FD$	$HexCommand FE FE 6E E0 07 C0 FD$

If the function exists, the only difference in the hex command between all ICOM radios (apart from the IC-735) is the third hex byte *i.e*. the default hex address for each model of ICOM radio.

ICOM IC-7600 ▼



ICOM IC-756 ProIII ▼



36.3.5 Examples of combined macros

For a **Kenwood TS-850**:

Set up split operation based on the frequency in the A VFO:

$Command FB#Split 05#;$

Note the semicolon between the final # and $ characters. This takes the frequency of VFO A, adds 5 kHz, and QSYs VFO B accordingly.

If the macro button is repeatedly clicked while VFO A is active, VFO A stays still and VFO B remains on the same 'up 5' frequency. However, if VFO B is active, it steps up in 5 kHz increments every time the macro runs. This effect can be reversed if the command is changed from FB to FA.

By the way, the 05 is not a value but a parameter code indicating (by coincidence) a +5 kHz offset.

Copyright © 2026 Robert C. Furzer K4CY

For a **Yaesu FT-920**:

$**HexCommand 00 00 00 01 01**$	Sets FT-920 to split mode
$**HexCommand #Spleeet+0h# 8A**$	Transfers VFO A frequency to VFO B (A>B)
$**QSY#Split+10#**$	Increments VFO A by 10 kHz
$**HexCommand #Spleeet+10h# 0A**$	Increments VFO A by 10 kHz (A+10)
$**HexCommand #Spleeet+10h# 8A**$	Sends VFO A+10kHz to VFO B (A+10 > B)

When Logger32 polls the FT-920 for the frequency and mode of VFO A, it only reports the correct values if the radio is in VFO mode. If the radio is in MEM, M-TUNE or QMB, an error message pops up on the RCP spectrum pane and at the bottom of the radio debug window advising you to turn off the MEM, M-TUNE or QMB as appropriate.

For a **Yaesu FT-1000mp**:

$**HexCommand 00 00 00 00 85**$	Copy VFO A frequency and mode to VFO B
$**HexCommand #Split+5h#8A**$	Push VFO A plus a split offset onto VFO
$**HexCommand 00 00 00 01 83**$	Enable dual-receive
$**HexCommand 00 00 00 01 01**$	Enable split mode

All combined:

$**HexCommand 00 00 00 00 85 #Split+5h#8A 00 00 00 01 83 00 00 00 01 01**$

For the **Yaesu FT-2000**, this acts a little differently:

$**Command FB#Split 05#;**$	takes the frequency from VFO A, adds the offset and applies it to VFO B

If the command is changed to read:

$**Command FA#Split 05#;**$

it behaves like the TS-850 and steps VFO A up by the stated offset each time the macro runs. So, a complete macro to take a frequency in VFO A and make sure that the frequency and mode for the QSX is correct will be:

- $**Command AB;**$ Copy VFO A to VFO B to make the frequencies and more importantly the modes the same.
- $**Command FB#Split 05#;**$ Take VFO A frequency, add the offset and push to VFO B
- $**Command FT3;**$ Transmit on VFO B
- $**Command FR2;**$ Enable both the main and sub-receivers

For the FT-2000, exactly the same result can be achieved using quick split:

- $**Command AB;**$ Copy VFO A to B to make frequencies and modes the same.
- $**Command EX033+05;**$ Set the offset value in the menu settings for the QS function
- $**Command QS;**$ Select quick split option
- $**Command FR2;**$ Enable both the main and sub-receivers

Copyright © 2026 Robert C. Furzer K4CY

For the **Elecraft K3**:

Method 1 using the **#Split#** function:

$**Command K31;FT0;SWT13;FB#Split+1#;SWH13;**$

… where:

- K31: sets K3 mode
- FT0: sets TX VFO to VFO A. Always reset to VFO A before sending the split command.
- SWT13: transfers VFO A to VFO B (VFO A = VFO B)
- FB#Split+1#: sets VFO B up 1 kHz
- SWH13: sets radio into split

Method 2 using K3 commands:

$**Command K31;FT0;SWT13;FT1;UPB4;**$

… where:

- K31: sets K3 mode
- FT0: sets TX VFO to VFO A
- SWT13: transfers VFO A to VFO B (VFO A = VFO B)
- FT1: sets TX VFO to VFO B
- UPB4: moves VFO B up 1 kHz (UPB5 = +2 kHz, UPB6 = +3 kHz, UPB7 = +5 kHz)

For the **Elecraft K4**:

While the K4 correctly interprets the K3 CAT commands, in order to make optimal usage of the K4's built-in features such its internal keyer, **D**igital **V**oice **R**ecorder, display controls, band memories, dual VFOs and more, it is necessary to use the extended K4 CAT controls. Read-up on Logger32 macro programming in the Macros and Programmable macro buttons and hotkeys chapters, and find more about the K4 in the Elecraft chapter.

For the **ICOM IC-735 (hex address 04)**:

$**HexCommand FE FE 04 E0 05 #Split+10h#FD**$	Increment the active VFO by 10 kHz

---

For the **ICOM IC-7600 (hex address 7A):**

$**HexCommand FE FE 7A E0 0F 01 FD**$	Turn split on
$**HexCommand FE FE 7A E0 07 b1 FD**$	VFO A=B
$**HexCommand FE FE 7A E0 05 #Split-10# FD**$	Define fixed split 10 kHz down
$**HexCommand FE FE 7A E0 07 b0 FD**$	Swap VFO A/B

Example setup macros for split from keyboard:

$**HexCommand FE FE 7A E0 0F 00 FD**$	Turn split off
$**HexCommand FE FE 7A E0 07 b1 FD**$	VFO A=B
$**HexCommand FE FE 7A E0 0F 01 FD**$	Turn split on

Copyright © 2026 Robert C. Furzer K4CY

$**HexCommand FE FE 7A E0 05 #Keyboard# FD**$ Up/down a number of kHz

$**HexCommand FE FE 7A E0 07 b0 FD**$ Swap VFO A/B

For the **ICOM IC-746 PRO (hex address 66)**:

$**HexCommand FE FE 66 E0 0F 00 FD**$ Split off

$**HexCommand FE FE 66 E0 07 A0 FD**$ VFO A=B (I always receive on VFO A)

$**HexCommand FE FE 66 E0 07 01 FD**$ Select VFO B

$**HexCommand FE FE 66 E0 0F 01 FD**$ Split on

$**HexCommand FE FE 66 E0 05 #DXSpotSplit# FD**$ Set Split from spot

$**HexCommand FE FE 66 E0 07 B0 FD**$ VFO A/B

$**icomVFOB**$ Read VFO B

For the **Kenwood TS-590**:

The following macro puts the radio into split mode with QSX applied to VFO B:

$**Command VV;**$ Copy VFO A frequency & mode to B

$**Command FT1;**$ Turn split on

$**Command #ModeModifier#FB#DXSpotSplit#;**$ Set VFO B frequency

Undo split:

$**Command FT0;**$ Turn split off

For the **Kenwood TS-990**:

The following macro puts the radio into split mode with QSX applied to the sub band:

- $**Command VV;**$ Copy main band to sub band (frequency & mode)
- $**Command TB1;**$ Turn split on

$**Command #ModeModifier#FB#DXSpotSplit#;**$ Set sub band frequency

Undo split:

$**Command TB0;**$ Turn split off

The following macros can be configured on RCP function keys/buttons for quick access. Up 1 split:

$**Command SP1**;$

$**Command SP001**;$

Up 2 split:

$**Command SP1**;$

$**Command SP002;**$

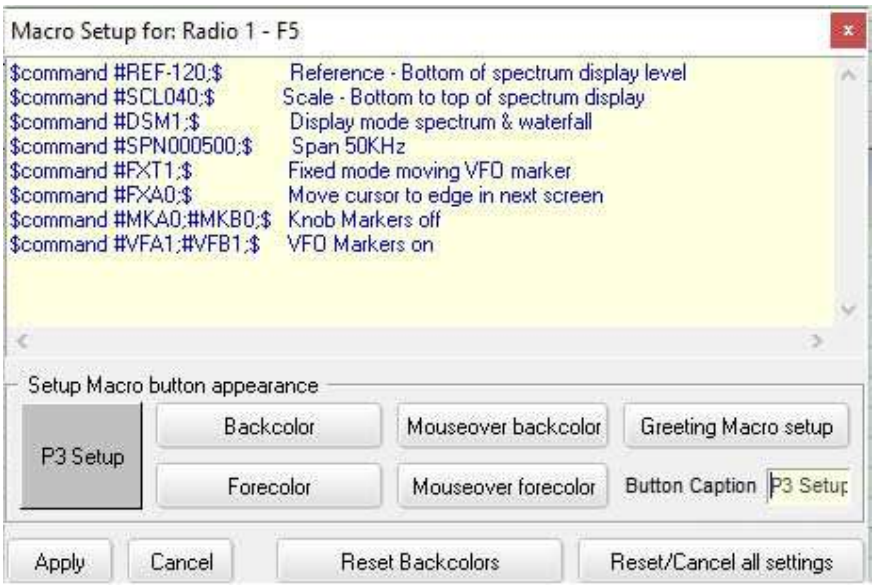
Copyright © 2026 Robert C. Furzer K4CY

36.3.6 Document your macros!

The macro examples above work with the specified radios. Check your radio operating or CAT manual for other models as the commands and parameter formats vary.

Most of the examples above have comments – arbitrary text *outside* the dollar pairs. If you are cutting-and-pasting the example macros into your RCP or directly into *C:\Logger32\RadioPanel.ini*, feel free to include the comments and make changes as you wish.

Here is a commented macro to setup the Elecraft P3 panadapter ▶



Hinson tip: it is good practice to document your code. Trust me, you *will* forget the fine details at some future point, so brief notes explaining each step are a worthwhile addition, especially in any complex sequence of commands. Although it is easier for us to read, comment and understand multi-step macros if the commands are split across multiple lines, conCATenating the commands into longer lines cuts down the polling delays and so speeds up delivery of the macro to the radio, so you might like to conCATenate time-critical macros after writing and debugging them.

36.4 Automatic DX spot split operations

36.4.1 #DXSpotSplit# function

This function tries valiantly to interpret and apply any split details found in the comments field of DX spots. Once configured, you can click a DX spot containing a recognized split designation and the split function will be active, even when the RCP is closed.

The main challenge with **#DXSpotSplit#** is the non-standard way in which DX spotters report splits. With some exceptions, the function correctly interprets spot comments such as:

- up 1
- up 5-10
- up 01
- up 10
- DN 01
- DOWN 01
- DOWN1
- QSX 21260.50
- SPLIT +5
- Tnx up up up 5
- 59 qsx 21285-21290
- qsx 21084.00
- up-2
- dn-2
- Tnx 4 nu band! Up 1.45
- PSE QSX 1810-1825
- LONGPATH UP 3-5
- qsx 7081 Alain
- WRK CQ,CQ UP 7012.9
- RX 18.159.0
- QSX 14.033.35
- 59 qsx 21.285-21.290
- QSX UP 1.25KHZ 599

Logger32 *assumes* that split frequencies reported are in kHz.

Copyright © 2026 Robert C. Furzer K4CY

The comments below are not understood (by Logger32 at least):

- LP up I (that's a capital i instead of the figure 1!)
- tnx band #4 - up 107 LP
- Up 111
- QSX 1.55
- QSX 111

There are, no doubt, many other bizarre variations that don't work automatically either, just as human DXers are sometimes confused by weird spots. Logger32 is clever but not omniscient.

The command syntax is the same as **#Keyboard#**, except one function derives the split from what you type, the other from comments in a clicked DX spot message.

Two user-defined macros are used in this process, one to be used if a split is detected in the comments text and the other to restore the radio to simplex operation otherwise. Each can be configured and disabled if you don't want to use either or both of them.

These macros are entered into the Setup RCP macros window seen below. The left-hand side of the table is for Radio 1 and the right for Radio 2. The middle sections are for macros to undo split, while the lower sections are to enter split operation.

Logger32 uses the median (mid-point) for spots giving a *range* of QSX frequencies or offsets. For example "59 QSX 21285-21290" uses 21287.5 kHz. The median calculation may not function as expected if there are additional characters (*e.g.* extra spaces) between the two figures, in which case the first of the two QSX frequencies is used. In any event, the QSX frequency is merely a suggestion, a starting point, somewhere probably within the DX station's spotted listening range: you should listen to the pile and tune around to find the best place to call the DX.

36.4.2 #ModeModifier#

The **#ModeModifier#** function modifies commands according to the radio's mode of operation. For example, you may want to change the receiver's bandwidth when you click a DX spot for station using a different mode. If the CAT command for bandwidth is of the form BWxxx; where xxx is the bandwidth in Hz, you could set up a macro including $**command BW#modemodifier#**;$ with mode-specific variables defined for Radio 1 (CW=100, SSB=2800, RTTY=3000, *etc*.) and for Radio 2 (CW=250, SSB=3000, RTTY=3000, *etc*.). Then, when you execute your macro with Radio 1 currently active, the CAT command BW100; would be sent to the radio for a CW spot, whereas if Radio 2 was active, the CAT command BW250; would be sent.

Aside from changing receiver bandwidth, you can do more creative things such as selecting different AGC settings or transmit filter bandwidths for different modes, and, for digital modes, turn off speech processing, reduce output power, disable the microphone and enable accessory audio input. Use your imagination.

#ModeModifier# must be embedded in a macro as shown in the DX spot split command (setup window - center section). The macro will vary according to the radio type and CAT command structure. The CAT command sent to the radio is constructed using parameters from the bottom section of the setup window as shown below. A comma must follow each entry in the table, including the last,

A typical macro to set the bandwidth on a Yaesu radio would be:

Macro: $**HexCommand 00 00 00 #ModeModifier# 8C**$

Filter table: **SSB=01,CW=02,RTTY=03,PSK31=04,**

A typical CI-V macro for ICOM radios would be:

Macro: $**HexCommand FE FE xx E0 06 #Mode# #ModeModifier# FD**$

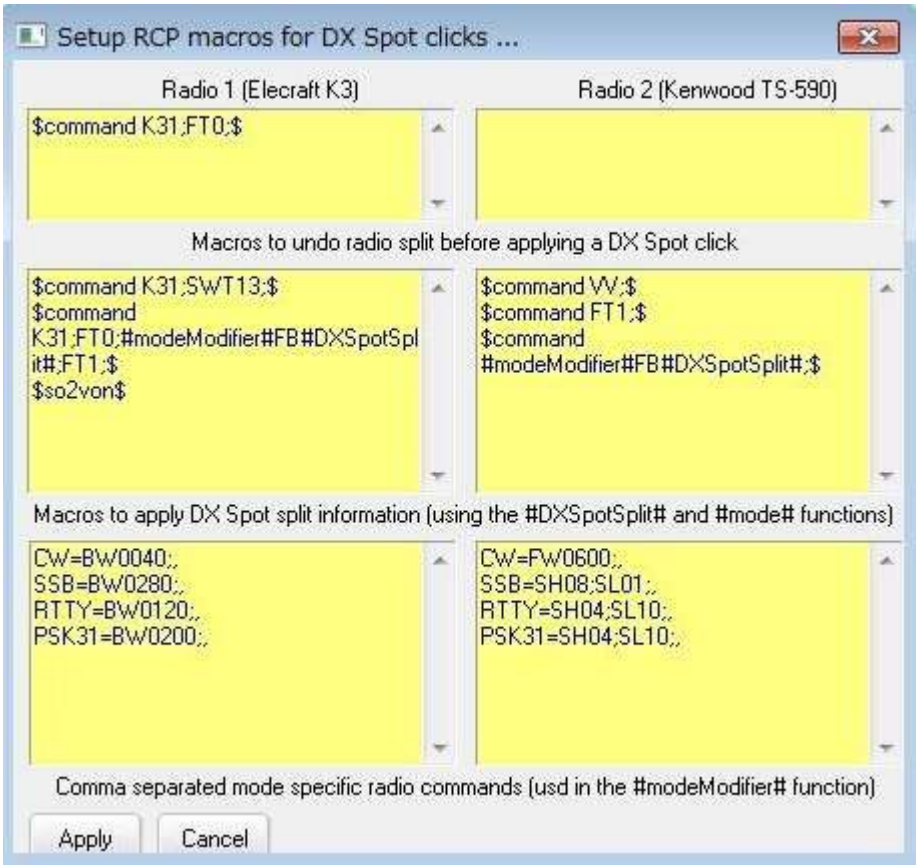
Copyright © 2026 Robert C. Furzer K4CY

where **xx** is your ICOM's CI-V address.

Filter table: **SSB=01, CW=03, RTTY=02, PSK31=02,**

ICOM uses the mode command to set the bandwidth. The #**Mode**# function reads the operating mode listed in the log entry pane and sends the mode *and* bandwidth information to the radio in a single CAT command.

Here is an example of the split setup for an Elecraft K3 and a Kenwood TS-590 in an SO2R station ▶



The CAT commands and hence macros to set and un-set split are radio-dependent. The examples given below illustrate the general approach. Check your radio's manual for the CAT functions, commands and control codes you need.

Copyright © 2026 Robert C. Furzer K4CY

Radio Make	Macro to UNDO radio split	Macro to apply split
Kenwood TS-850	$command FB#split 00#;$	$command FB#DXSpotSplit#;$ Put split freq in B $command FR1;$ Swap to VFO B $command MD#mode#;$ Make MODE same as VFO A $command FR0;FT1;$ Set RX on VFO A and TX on VFO B
Kenwood TS-590	$command FT0;$	$command W;$ $command FT1;$ $command FB#DXsPOTsPLIT#;$
ICOM IC-735	$hexcommand FE FE 04 E0 07 00 FD$ VFO A	$hexcommand FE FE 04 E0 07 01 FD$ VFO B $hexcommand FE FE 04 E0 05 #DXSpotSplit# FD$ Set freq $hexcommand FE FE 04 E0 07 00 FD$ VFO A
Please note:- The IC-735 is a first generation CI-V protocol and does not support the full complement of CI-V commands. This requires the operator to manually place the radio into and out of the SPLIT mode. The above setup will properly set both VFO-A and VFO-B, leaving the receive frequency on VFO-A and transmit on VFO-B		
ICOM IC-7600	$hexcommand FE FE 7A E0 0F 00 FD$ Split Off	$hexcommand FE FE 7A E0 07 b1 FD$ vfo A=B $hexcommand FE FE 7A E0 0F 01 FD$ Split ON $hexcommand FE FE 7A E0 05 #dxspotsplit# FD$ UP-DN(nn) $hexcommand FE FE 7A E0 07 b0 FD$ vfo A/B
ICOM IC-756 Pro III	$hexcommand FE FE 6E E0 0F 00 FD$ Split Off	$hexcommand FE FE 6E E0 07 b1 FD$ vfo A=B $hexcommand FE FE 6E E0 0F 01 FD$ Split ON $hexcommand FE FE 6E E0 05 #dxspotsplit# FD$ UP-DN(nn) $hexcommand FE FE 6E E0 07 b0 FD$ vfo A/B
Yaesu FT-920	$hexcommand 00 00 00 00 01$ Undo split	$hexcommand 00 00 00 02 10$ Poll radio $hexcommand 00 00 00 8#mode# 0c$ Set VFO B mode $hexcommand 00 00 00 01 01$ Place in split mode $hexcommand #DXSpotSplit# 8A$ Apply split freq to VFO B
Yaesu FT-1000mp	$hexcommand 00 00 00 00 83$ $hexcommand 00 00 00 00 01$ split and dual OFF	$hexcommand 00 00 00 00 85 $ Modes made the same for both vfos $hexcommand #dxspotsplit# 8A$ Take VFO A freq, add the split offset and push result into VFO B $hexcommand 00 00 00 01 83$ Put dual and split functions on $hexcommand 00 00 00 01 01$
Yaesu FT-2000	$command FT2;$ Turn off TX on B VFO $command FR0;$ Turn off Sub receiver	$command AB;$ Copy A vfo to B vfo-To make frequencies and more importantly the MODES the same $command FB#DXspotSplit#;$ Take A vfo freq add the dxspot offset and push result to the B vfo $command FT3;$ TX on the B VFO $command FR2;$ Turn ON both the main and sub receivers and push to the B vfo
Elecraft K3	$command K31;FT0;$	$command K31;FT0;FB#DXSpotSplit#;FT1;$

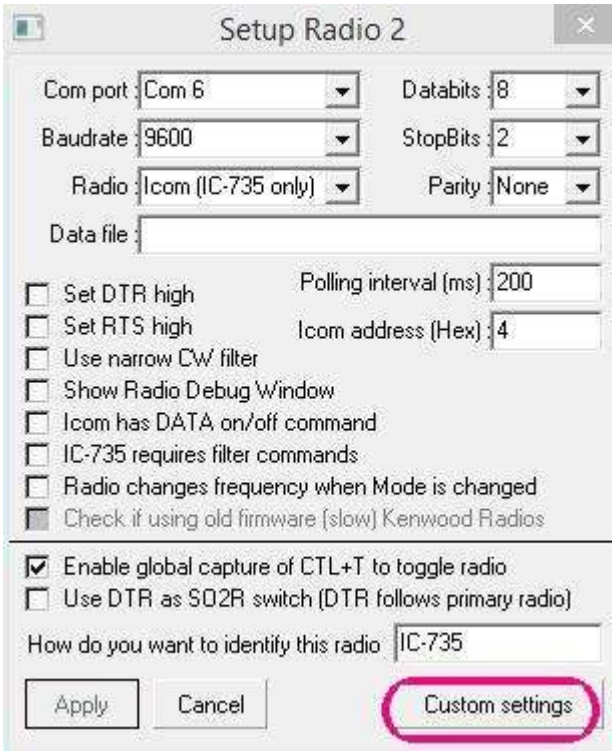
Copyright © 2026 Robert C. Furzer K4CY

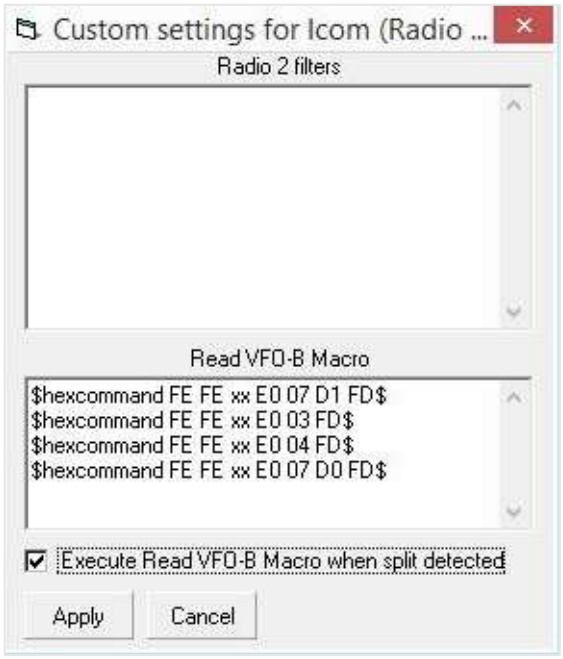
36.4.3 $IcomVFOB$

ICOM radios don't have a specific CAT command to read the frequency and mode of VFO B (the sub-VFO) as other radios do. The $**IcomVFOB**$ macro can read the VFO B frequency and mode, but can't track the frequency in normal split operation while we tuning up or down the band, hence you may need to use the $**IcomVFOB**$ macro again to update the information.

To configure the macro, use **Setup ⇨ Radio ⇨ Radio 1|2 configuration ⇨ Custom Settings ▶**

This opens a setup widow where you can see four ICOM commands. Replace xx with your ICOM radio model's hexadecimal CI-V address.





◀ This shows the split macro being setup for an IC-7600.

The default CI-V commands shown initially are for the IC-7000 and should work with most late model ICOM radios. Verify the actual commands for your radio. They are in order:

1. Switch to VFO B
2. Poll frequency
3. Poll mode
4. Switch to VFO A

Setup the macro with the correct address for your radio and if you like select <**Execute Read VFO-B macro when split detected**>. Your ICOM (if it is a more modern one) will hopefully read the VFO B frequency/mode every time you put the radio in split. Also, if you setup a macro button on the RCP with the $**IcomVFOB**$ macro, you can update the VFO B reading at will.

The RCP panel shows
both VFOs in operation ▼



Copyright © 2026 Robert C. Furzer K4CY

36.5 Alerts for split operation

Logger32 can alert you visually and/or audibly when the radio is set to operate in split (with different TX and RX frequencies) after clicking a DX spot with comments containing usable split frequency information.

36.5.1 Visual split alert

To setup the visual alert function, you have two choices:

1. Configure it through the **R**adio **C**ontrol **P**anel[431]:
   - Right-click the RCP side bar and click **Setup split alerts ⇨ Visual alert font & color** -or- **Visual alert backcolor** to configure the text and background colors.
   - The visual alert will remain until the split function is disabled, *unless* <**Limit visibility to 5 seconds**> is selected.
   - Enable the visual alert function by ticking <**Enable visual notification when radio split detected**>.
   - In order to position the alert anywhere, select <**Unlock Visual Alert from Logbook Entry Window**> and drag the alerting window into place.
   - To position the alert within the log entry pane's caption, *de*select <**Unlock Visual Alert Window from Logbook Entry Window**>

When split visual alert has been triggered by clicking a DX spot with meaningful QSX details in the spot comment, the log entry pane may look something like this ▶

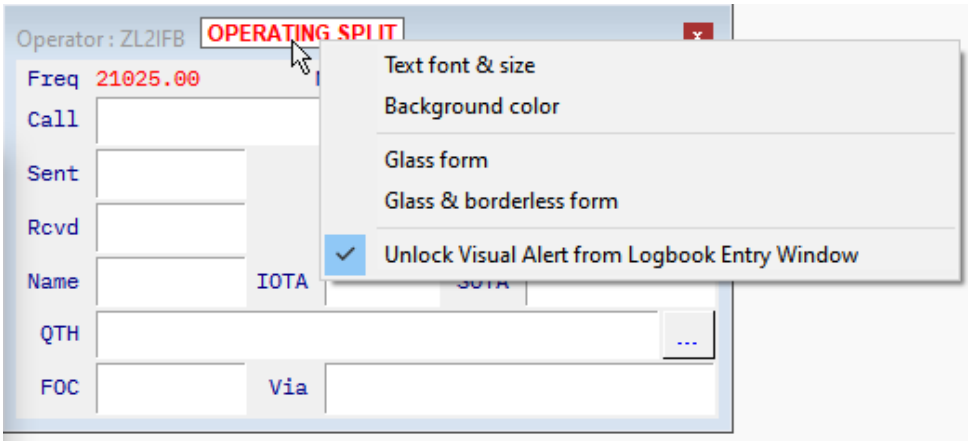


[431] You can select not only the forecolor (for the text) and the background color (for the field) but also the font name, style and size. The visual alert window is automatically resized to suit.

Copyright © 2026 Robert C. Furzer K4CY

2. Configure the visual alert message directly:

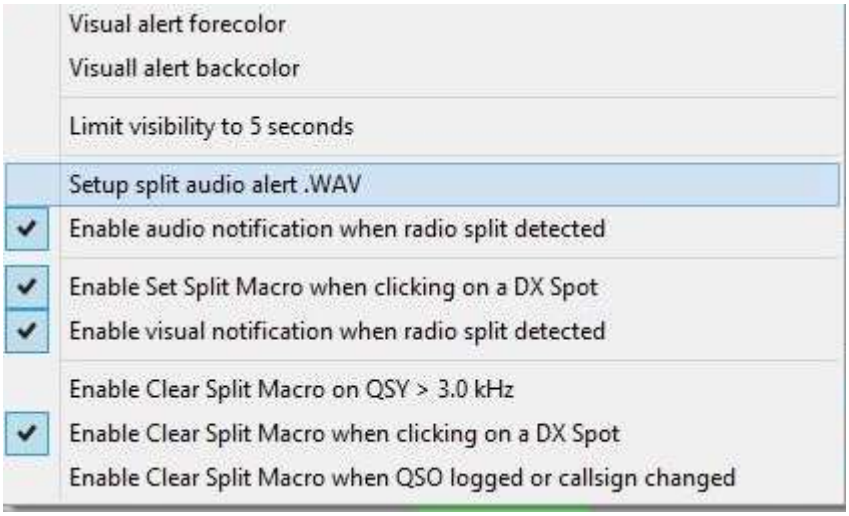
With the split alert message visible (*i.e.* with the radio operating split), right-click the message for yet more options ▶



- **Text font & size**: setup the alert message text font, color and size.
- **Background color**: choose the alert box background color.
- **Glass form**: make the alert box transparent with just a thin border/frame.
- **Glass & borderless form**: make the alert box transparent without a border/frame.
- **Unlock Visual Alert from Logbook Entry Window**: allows the alert box to be dragged anywhere on your screens. Put it somewhere obvious as a reminder when you are split.

36.5.2 Audio split alert

To setup which audio alert to play when you use split, click **Setup ⇨ DX spot Macros ⇨ Setup audio alert .WAV** then find an audio file containing the alert sound you want ▶



Click <**Test**> to hear the alert play and, if it is OK, click <**Apply**> to save the setting.

Tick <**Enable audio notification when radio split detected**> to activate the alert with the chosen sound.

36.5.3 Enable Clear Split Macro on QSY > 3.0 kHz

If this option is selected, the log entry pane will clear automatically if you QSY beyond the stated range and, if you were operating split, the radio will revert to simplex. The 3 kHz range should let you tune around a typical pileup to find a suitable transmit slot without erasing the DX callsign or cancelling split … but may not be wide enough for straggly or unruly pileups chasing rare or inept DX stations.

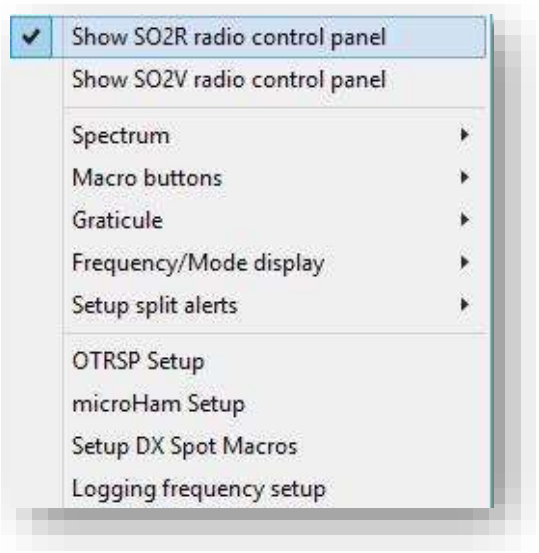
Hinson tip: this setting can cause grief when, later on, you've forgotten about it, and discover perplexing weirdness when idly tuning around. It may *seem* cool, but is it, really?

Copyright © 2026 Robert C. Furzer K4CY

36.6 SO2R and SO2V RCP operation

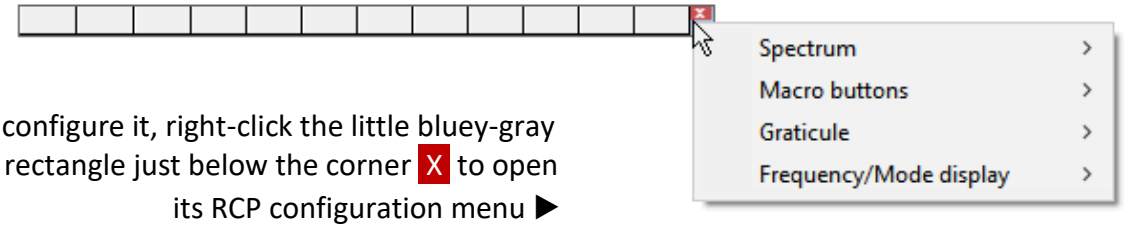
Logger32 supports SO2R and SO2V operation similarly to the Sound card data window, with a pair of RCPs.

Only radios with a second/sub-receiver are capable of SO2V. A radio with two VFOs but just one receiver cannot do SO2V: *you need to be able to listen to both frequencies at the same time*.

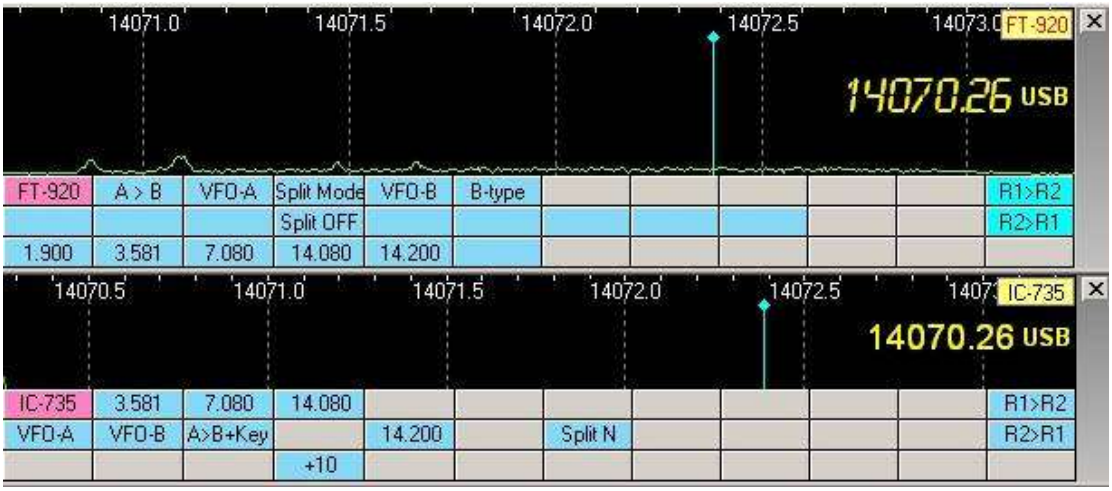


To enable SO2R and/or SO2V, right-click the right side of the RCP, then click <**Show SO2R radio control panel**> or <**Show SO2V radio control panel**> to open the second RCP ▶

That initially opens a minimalist RCP for the second VFO/radio, with just a single row of 12 blank macro buttons ▶



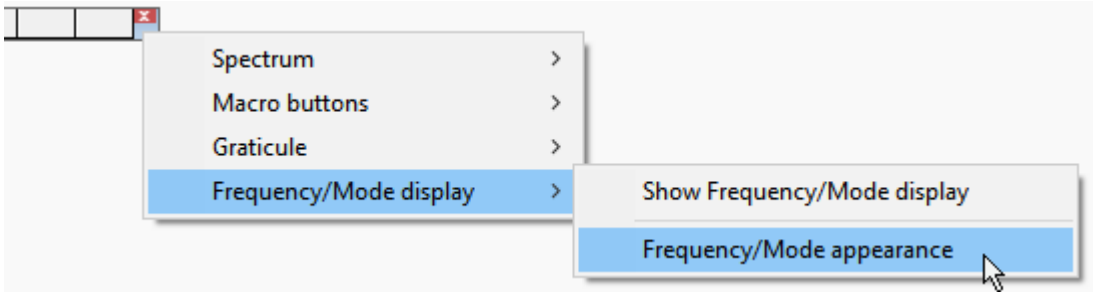
To configure it, right-click the little bluey-gray rectangle just below the corner X to open its RCP configuration menu ▶

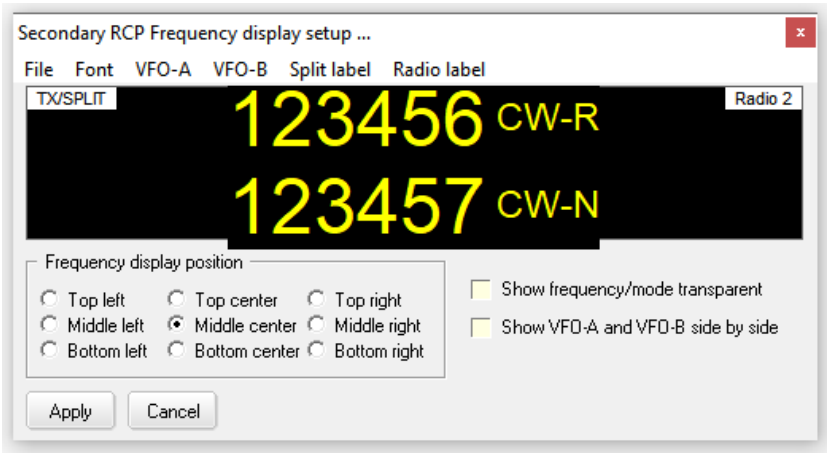


▲ The VFO frequencies and radio modes can optionally be displayed on the spectrum areas for each RCP, in our choice of fonts, colors, sizes and positions. We can even configure the radio name labels in the top right corners (such as “LEFT” and “RIGHT” if we have a pair of radios of the same type sitting side-by-side on the shack desk in front of you).

Copyright © 2026 Robert C. Furzer K4CY

To configure the frequency and mode text, right-click the rectangle below the corner X, then click **Frequency/Mode display ⇨ Frequency/Mode appearance ▼**

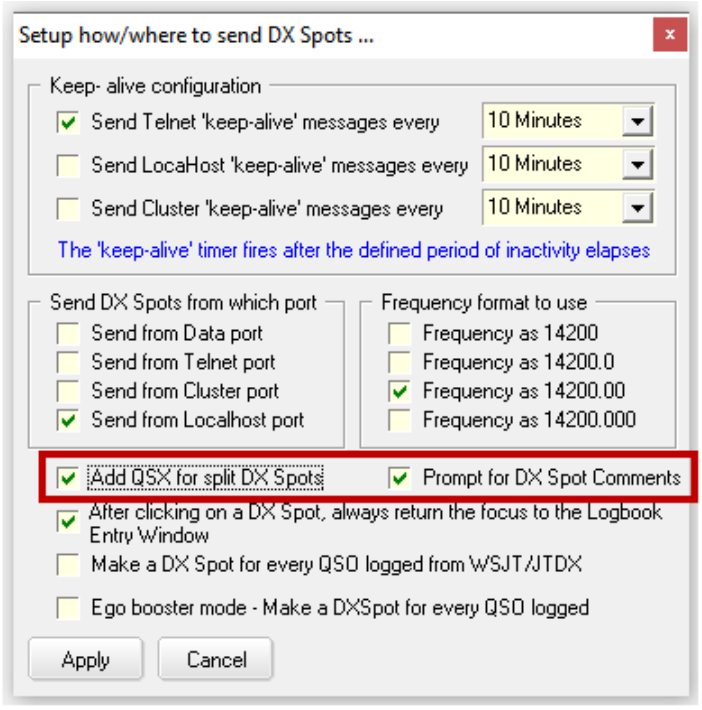




◀ The configuration is quite straightforward so we won't describe it here: simply work your way systematically through the menus.

36.6.1 Setup DX spot QSX reporting

With some limitations (*e.g.* not on older ICOM radios), Logger32 can optionally add the QSX frequency to the comments field when you post DX spots for DX stations working split.



◀ If you choose to do so, open **Setup ⇨ DX spot** and select both options indicated here.

Given that splits (QSX offsets) greater than 100 kHz are probably user errors, Logger32 does *not* report them in the generated DX spot comments, although we are free to edit the comments as

Copyright © 2026 Robert C. Furzer K4CY

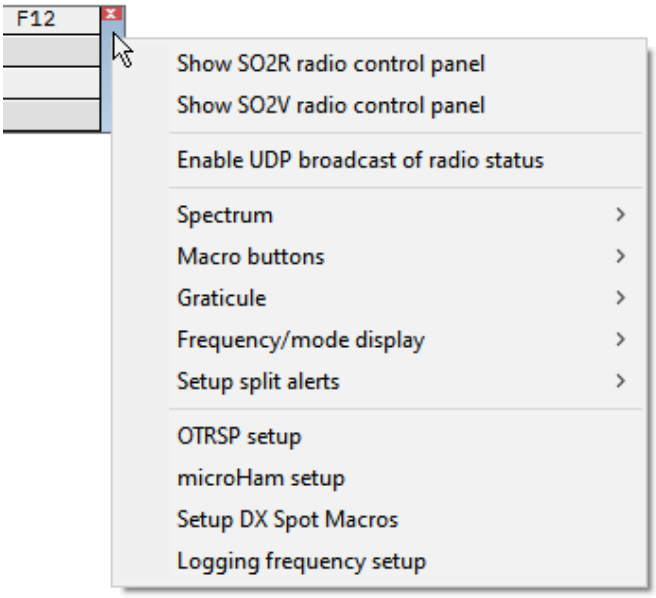
we wish. Regardless of that, however, QSOs *can* be logged with greater splits, including cross-band operation (transmitting and receiving on different bands), as we'll see in just a moment …

Hinson tip: keep an eye on your transmit VFO frequency, *especially* near the band edges, in order to avoid accidentally transmitting out of band (if your radio doesn't automatically prevent that: check the manual) or beyond a designated mode or license class segment.

36.6.2 Split frequency operation: logging TX *and* RX frequencies

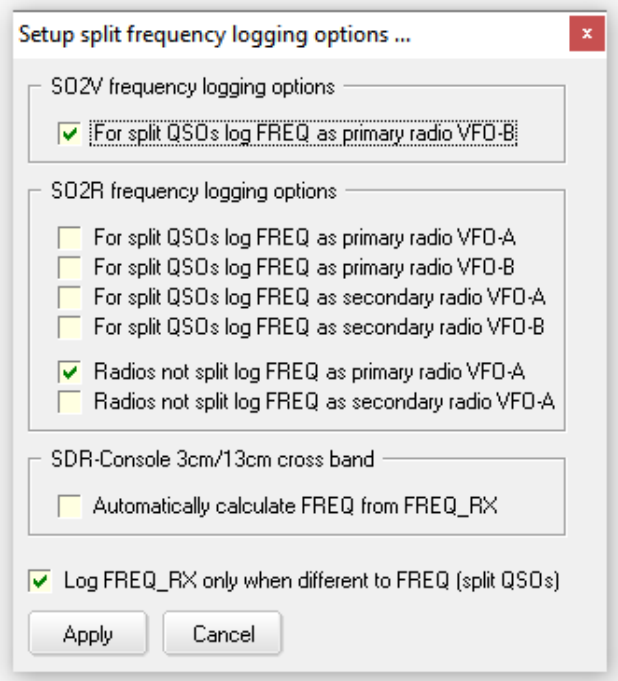
When we are operating split or cross-band, perhaps using SO2V or SO2R, how do we want Logger32 to log the frequencies of our QSOs?

The convention is to log **the TX frequency on the primary radio**, hence Logger32 needs to know which VFO and radio that is.



◀ Right-click the bluey-gray rectangle below the corner X on the RCP to open this menu.

◀ At the bottom, click <**Logging Frequency Setup**> to open the setup form ▼



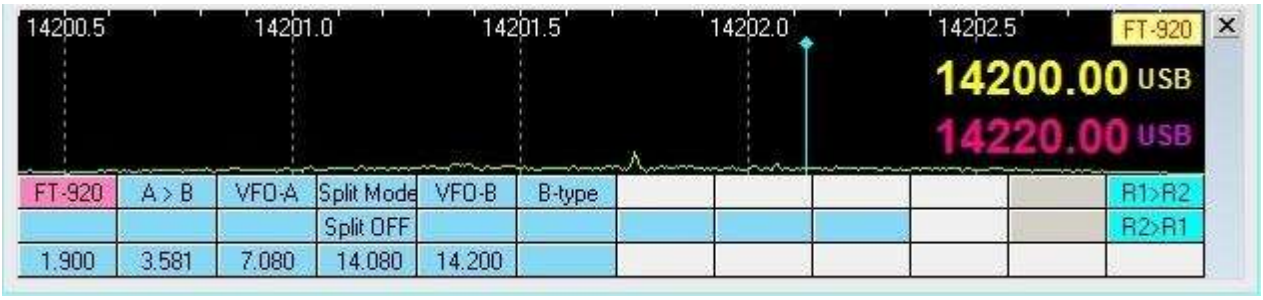
Copyright © 2026 Robert C. Furzer K4CY

Unless we tick the last option on the <**Logging Frequency Setup**> form, Logger32 *routinely* logs *both* our TX *and* RX frequencies:

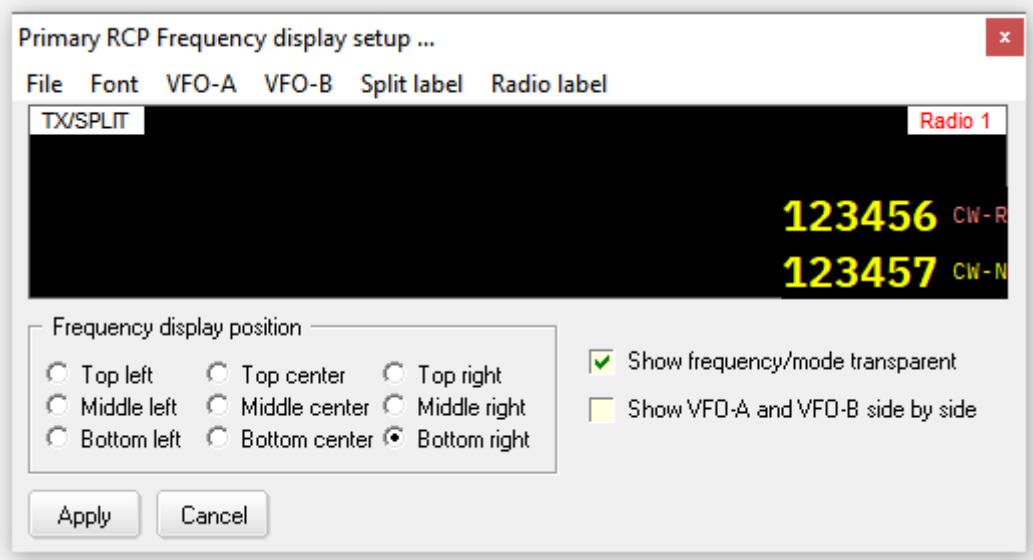
- The TX frequency and band are normally shown at the top of the log entry pane and in our logbook, using the correct but ambiguously-named ADIF fields “FREQ” and “BAND”.
- The RX frequency is quietly logged in the ADIF field “FREQ_RX”, while the RX band is discreetly logged in the ADIF field “BAND_RX” although neither are normally shown …
  - To show *both* frequencies in the log, simply display both “FREQ” *and* “FREQ_RX” columns.
  - To see *both* logged bands for any cross-band activities, display the “BAND” *and* “BAND_RX” columns in the logbook.
  - Feel free to rename those logbook columns as you wish *e.g.* “TX freq” would be a more useful label for the “FREQ” column.

36.6.3 Flags and Indicators

Here is a typical RCP window showing the frequencies and modes of VFO A (in yellow) and VFO B (in red) ▼

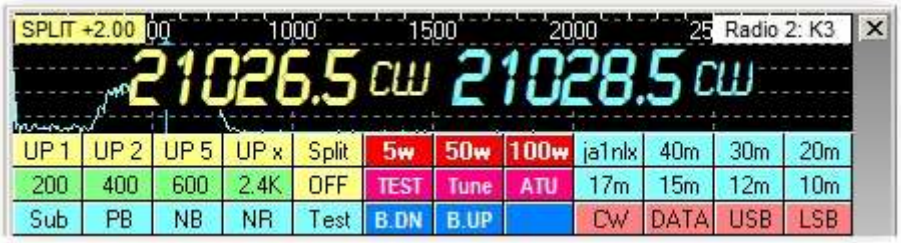


The VFO B frequency can only be displayed *if* the radio CAT poll replies supply the data (some ICOMs do not). The appearance (size, color) and position of the frequency and mode values can be configured using the RCP right-click menu **Frequency/mode display ⇨ Frequency/mode appearance** ▼



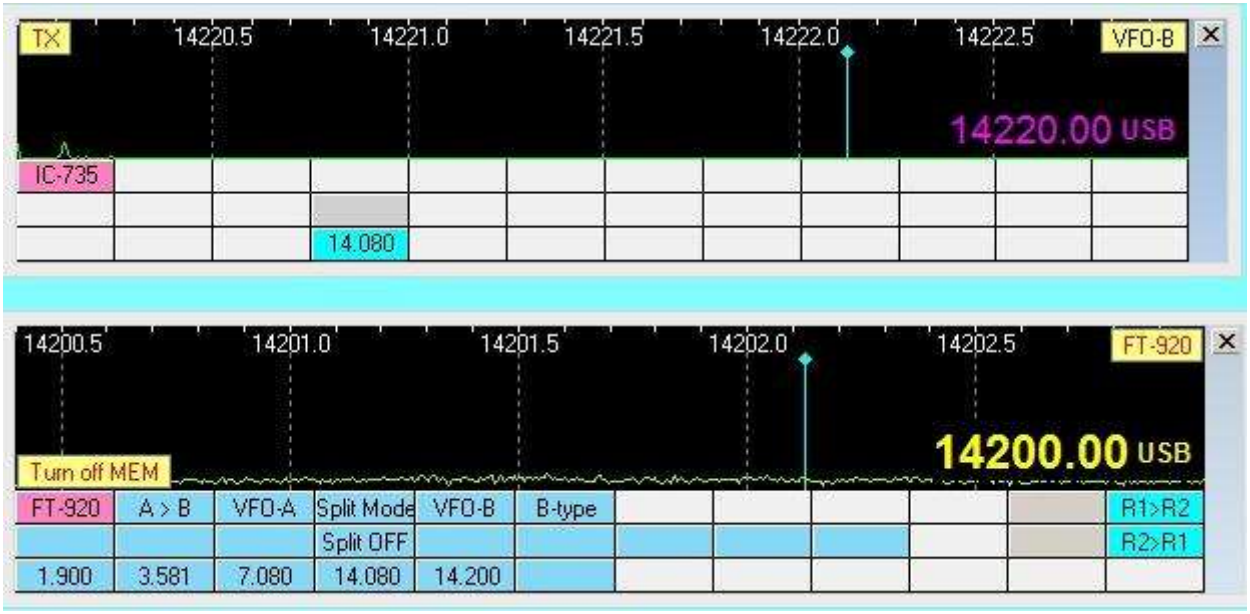
Copyright © 2026 Robert C. Furzer K4CY

Here, the radio is in split mode as indicated in the top left corner ▼



The RCP shows "SPLIT" and the TX offset in kHz. You can choose the colors for the split label. The macro buttons have also been grouped together and color-coded in that example.

Below ▼ is an SO2V setup showing VFO A below (by user preference) and VFO B above.



The radio is operating split, transmitting on VFO B as indicated by the "TX" flag in the top left corner. The blinking message "Turn off MEM" at the bottom left corner of the spectrum warns the user that the receiver is currently locked to a fixed memory channel and is *not* using VFO A[432].

36.7 RCP FAQs

Q. Can I have more than 48 radio macros?

A. Yes. If you have been incredibly creative, programming more than 48 radio macros, you will have run out of buttons on the RCP.

However, if you only use one radio, you can define the Radio 2 COM port and CAT parameters the same as Radio 1, then you can notionally 'swap radios' using <**Ctrl**+**T**> to access *another* set of up to 48 macros for the 'other' radio, sending the commands via the same port to the same radio.

[432] Some radios only return the VFO A data in response to Logger32's polling. If the radio is tuned to MEM, M-Tune or a QMB channel, the frequency displayed in Logger32 may be wrong.

Copyright © 2026 Robert C. Furzer K4CY

Alternatively, you can:

- Set up different Logger32 configurations with different macros for each – perhaps sets of macros for regular QSOs, DXing, contesting and ragchewing.
- Open the Data Terminal (TNC) function and exploit its 48 macro buttons as well as the RCP's 48.
- Stop playing around, review and rationalize your selection of radio macros, trim the least useful ones … and spend more time making QSOs!

Q. I have spent weeks patiently designing, testing and refining a *fabulous* set of RCP macros for one of my SO2R radios, but they all disappear when I swap to the other radio. Can I transfer my fab macros to the other radio?

A. You could of course write down a fab macro from radio 1 on a scrap of paper, then swap to radio 2 and type it in to the **R**adio **C**ontrol **P**anel, and repeat until done … but it may be a tedious and error-prone job to replicate all your macros this way and – more importantly – the macros will need to be modified unless both radios use the same commands.

The macros are stored in a plain text configuration file *C:\Logger32\RadioPanel.ini* under the sections headed "[Radio 1 macros]" and, yes you guessed it, "[Radio 2 macros]". So, if both radios are identical or at least similar (*e.g*. radios from the same manufacturer of about the same vintage), you can simply copy and paste macros from one section to the other using a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad, adjusting them if necessary in the editor or later through Logger32.

If your radios are completely different (*e.g*. an ICOM and a Yaesu), you will need to consult the radio manuals and this tome to develop equivalent macros for each – although that may prove tricky or impossible in the case of complex or unusual macros if the radios don't offer the same capabilities. As a simple illustration, your cunning radio 1 macro to select a narrow crystal filter to listen intently to a genuine DX station and a wide filter when checking through her pileup won't work the same way if radio 2 does not have a narrow crystal filter installed. Can you achieve a similar effect using 'slope tuning' or digital filtering instead?

Hinson tip: keep notes! As with software source code, it pays to document your macros, explaining what each command or function does. This makes it easier to understand and correct, modify or replicate the macro later on, especially if you don't do this often and have a poor memory for obscure syntax.

Hinson tip: use the same or similar RCP button labels and positions for the same or similar macros on each radio, unless you relish the mental challenge whenever you swap radios.

Q. In the RCP I have F12 set up to trigger the $ClearLog$ macro. Sometimes, after I've logged a QSO, I need to click on the log entry pane before I can type the next callsign, wasting valuable clicks and time …

A. Try appending $**FocusToCallsignField**$ to the $**ClearLog**$ macro to make it focus directly on the call field every time, without need for extra clicks.

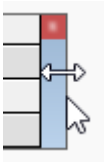
ConCATenate the macros on the same line to skip the polling interval and so shave a few precious milliseconds *i.e*. $**ClearLog**$$**FocusToCallsignField**$

Copyright © 2026 Robert C. Furzer K4CY

Q. I can't move the RCP window. I don't see the four-way arrow cursor. What am I doing wrong?

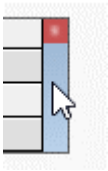
A.

The left-right arrow appears when **pointing** the mouse cursor at the vertical line at the far right edge of the blue-gray rectangle ▼

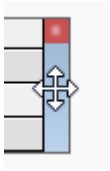


From there, I can click and drag the right edge of the RCP to the left, shrinking the RCP, or out to the right to widen it.

When **pointing** near the middle of the blue-gray rectangle, I see my usual mouse cursor ▼ ...



… and in that position, when I **click and hold** the mouse button, the mouse cursor turns into the four-way arrow ▼



While still holding the mouse button down, I can now drag the RCP around the screen like a small dog on a leash, then drop it wherever I like by releasing the button

Admittedly, the blue-gray rectangle is a fairly small target (especially if the RCP only has one row of 12 macro buttons and no audio waterfall), requiring decent eyesight and a steady hand. Anyone with cataracts or Parkinsonism needs patience, luck or an able-bodied helper.

Hinson tip: if you are left-handed and have configured your mouse to reverse the mouse buttons, *try* clicking the 'wrong' (other) button instead. The button-reversal function is not entirely consistent *e.g.* it *may* not work on the panels at the bottom of the Logger32 window. The issue relates to how the mouse driver interacts with Windows: it's not a Logger32 bug.

Copyright © 2026 Robert C. Furzer K4CY

37 Sound card-radio interfacing

> "The best interface
> is no interface"
>
> *Golden Krishna*

There are two things you must do to interface a radio to a PC sound card/audio subsystem:

1. Get audio from the radio to the PC so that Logger32 can decode received signals.
2. Send audio from the PC to the radio to generate and transmit a digimode signal.

All other interfacing (PTT, CAT radio control and interaction with Logger32) is covered elsewhere in this User Manual.

Depending on your station configuration and operating preferences, you may want to refine your setup. For instance, if you have the money, you may prefer to buy a commercial interface rather than construct your own.

37.1 Interfaces

37.1.1 Receive interface

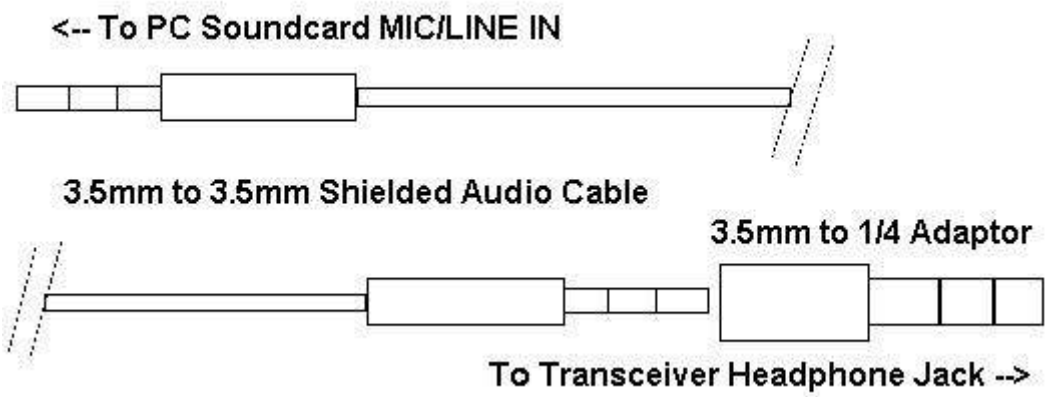
To receive and decode digital mode signals, connect the receive audio to either the PC sound card Line In or Mic In port.

The simplest way to do this is with a shielded stereo audio cable between the receiver's headphone jack (possibly a ¼" mono jack on the front panel of ancient radios, a ¼" stereo jack on more recent radios and often these days a 3.5mm stereo jack on the front and/or rear panel) and the sound card's Line or Mic in jack (generally 3.5mm stereo, sometimes a pair of phono sockets for the left and right channels). Be sure to use a decent quality *shielded* cable to reduce the possibility of RF interference. If you make your own, invest a little in metal plugs as well.

If you have the choice, the sound card's Line in jack is a better choice than the Mic in jack, because the line input is harder to overload, but either should work. Some sound cards only have a Mic in jack, or a shared jack that is configured in the sound card software for either a microphone or line level input. Either way, adjust the Windows recording level and/or the radio audio output as described below.

Copyright © 2026 Robert C. Furzer K4CY

Here is a receive audio cable using a commercial or homebrew 3.5mm-to-3.5mm stereo audio cable and a 3.5mm to 1/4" stereo adapter if needed at the transceiver end ▼

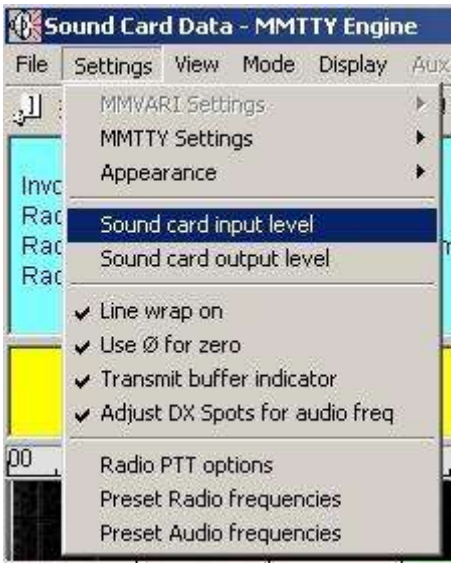


In some setups, with the headphone plug part of the way in, you can monitor transceiver audio through the radio speaker while also sending the audio to the sound card. This is a great aid in tuning signals by ear while monitoring them on Logger32's tuning display.

With this simple cable, you're ready to try to receive digital mode signals with Logger32.

37.1.2 Set receive audio levels

It is important to adjust your sound input and output levels using the Recording and Speaker level controls in Windows. You can also adjust these levels within MMTTY and MMVARI.



Click **Settings ⇨ Sound Card input level** to open the Recording mixing console ▶

Click **Options ⇨ Properties**, then select the recording radio button. Make sure the input(s) you are using (Mic or Line) are selected, then click <**OK**> to open the Recording mixer window.

For the next adjustments, arrange your Logger32 window and the Recording Mixer Window so that you can see and switch them both. Set your transceiver's volume (AF level) to a comfortable listening level.

Depending on your setup, adjust the Mic or Line level input controls and the overall recording level using the slider. One approach is to tune your radio to an area of a band with digital mode activity (*e.g.* 14074), then click the area of the strongest activity as shown in the spectrum display (lower segment of Logger32's main operating window) to direct Logger32's attention to that signal.

Start by setting the overall recording volume in the center of its adjustment range. Adjust the input level on the Mic or Line input until you can clearly see the signals on the spectrum display, but they do not turn the display red. Signals will not reach the very top of the spectrum display if they are adjusted properly.

Copyright © 2026 Robert C. Furzer K4CY

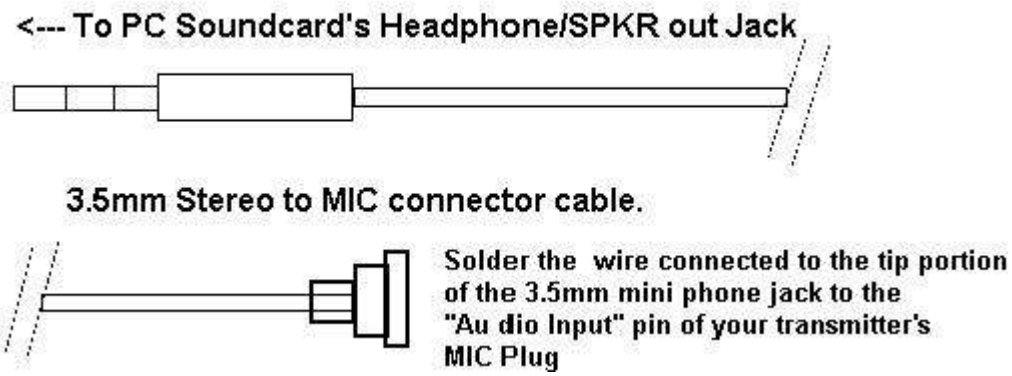
Overdriving inputs will severely degrade the copy of all digital modes. Adjust for the minimum record levels, while still providing a good display in Logger32. It may be necessary to reduce the overall recording level as well as the Mic or Line inputs to achieve this. It may also be necessary to attenuate the signal between the transceiver and sound card, especially if you're using the Mic input of your sound card and a fixed output level from the radio.

If you don't see any receive activity on Logger32's displays, make sure that your Mic or Line input control is not muted ("Mute" is a check box next to the slider in your record mixer) and also double-check all connections. It is possible to overdrive the sound card, and may need to attenuate the input signal. It is also possible that the sound card is incompatible with Logger32, so you may need to install a separate sound card/audio device.

After these receive adjustments are made, try to copy some QSOs by tuning them in using Logger32's tuning indicator(s) as shown above.

37.1.3 Transmit interface

To transmit digital signals, you need to connect the sound card output (often through an isolation transformer, or 100:1 attenuator) to the transmitter microphone or AFSK input. The following picture shows a direct connection, but many transceivers and sound cards will need the voltage divider shown later in this section: please check the instructions for your radio. Some radios have a divider in the rear audio input, but not in the microphone connector.



Turn off all transmitter speech processing in the radio. Any type of speech processing will distort the audio and so reduce copy at the far end.

Overdriving your transmitter audio stage in PSK31 creates big IMD (intermodulation distortion) problems, and is a major cause of interference in this mode. It is very important to get this right. Be sure to check in your first QSOs to make sure that you do not have a high IMD. Many PSK programs, like Logger32, read this value out on received signals. Send about five seconds of idle signal to get a stable IMD reading.

With the above interface connected, and your transceiver's antenna jack connected to a dummy load, set the audio output level of your PC sound card to match your transceiver's input circuit.

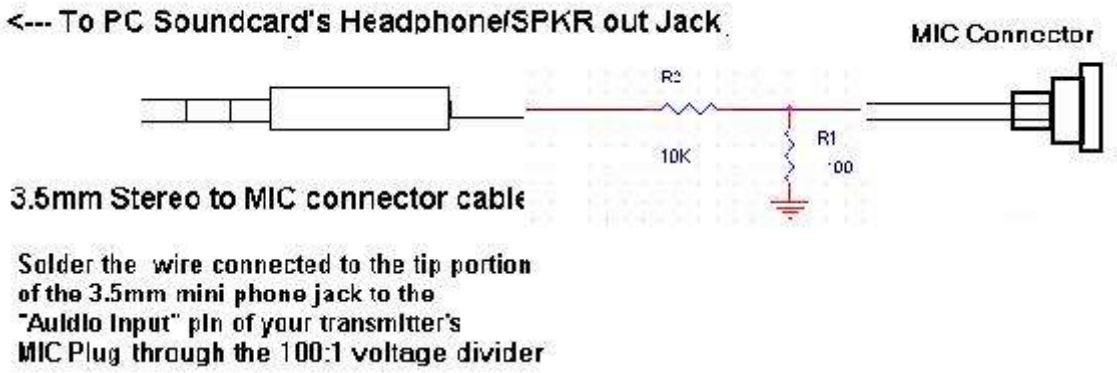
37.1.4 Reducing audio drive for a cleaner signal

In most cases the sound card provides too much drive for the radio's audio stage. Even by setting the computer mixer output control to its lowest level, you may still have a high IMD value. You are actually overdriving your radio's microphone input stage before the Mic gain control, so no matter how low you set the Mic gain, you are already distorting. It is strongly recommended that you use

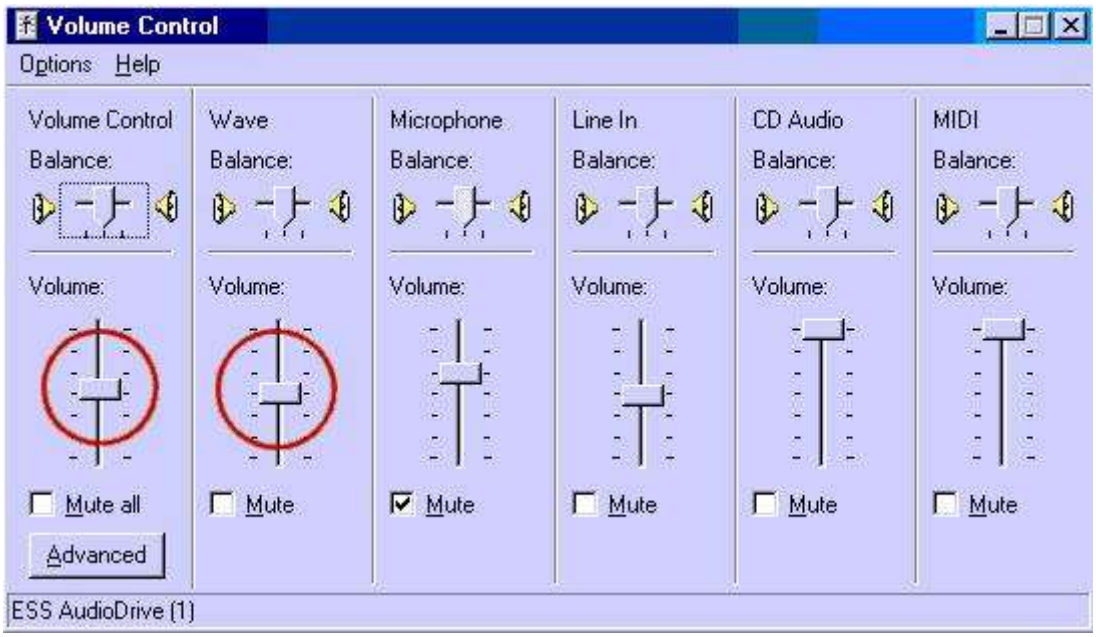
Copyright © 2026 Robert C. Furzer K4CY

a voltage divider to reduce your sound card output. You can make or purchase add-on hardware designed for PSK and other digital modes that includes this circuitry.

Here is a voltage divider. If you want greater control, you can replace the series resistor with a potentiometer:



Again, it is extremely important to match your sound input and output levels. To set the output level, click **ToolBox ⇨ Sound card output level** to open the volume control mixer ▼



37.1.5 Set transmit level

Audio output adjustments are best made with your transceiver connected to a dummy load. Set your transceiver Mic gain control slightly above its minimum setting and if available set your radio's metering to display ALC. If you're using VOX, set it as per other modes. Otherwise, you can make these adjustments by manually engaging transmit on the radio at the same time that you tell Logger32 to generate and transmit audio.

37.1.6 Prepare to adjust the PC

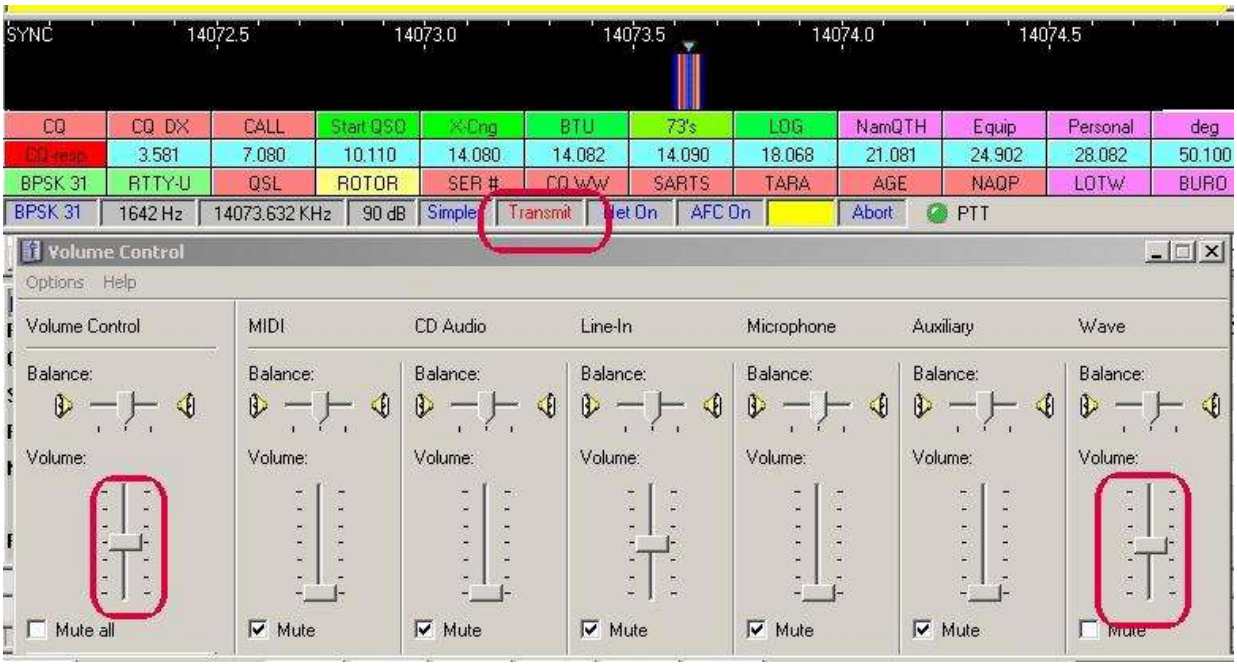
With Logger32 running and the volume control mixer open, click **Tool Box ⇨ Sound Card output level**. For now, slide the Windows volume control to a minimum (all the way down), and the wave slider just above the minimum setting.

Copyright © 2026 Robert C. Furzer K4CY

Click the small blue **Receive** panel in the middle of the MMTTY status bar, turning it to Transmit (red). Logger32 transmits a single tone in the Tune mode. To return to receive mode simply click the same panel again[433].

Don't lose track of how long you are transmitting. Long transmissions may overheat your finals.

Here is what this setup should look like with both windows visible ▼



Notice that other audio sources are muted in this example, to avoid transmitting music *etc*. ▲

37.1.7 Adjust radio audio and VOX

There is a lot of interaction between the VOX level setting and the audio setting.

Slowly raise the Volume control on the mixer until the radio's VOX circuit engages and the radio starts transmitting. If VOX has not engaged by the time the Volume control is midway up the slider scale, then raise the radio's Mic gain slightly and try again. Watch the radio's ALC indicator. You want a minimum reading here, indicating that there is just enough audio to drive the radio, but not so much that you run the risk of overdriving the Mic input. Overdriving the Mic input circuit is a common cause of distorted and wide signals when using this type of sound card set up. In fact, any operation of the radio's ALC means that it is distorting your transmitted signal. PSK31 is especially sensitive to these settings, but all the digital modes will suffer. Voice signals are not as distorted by the action of ALC.

If VOX does not seem to operate at low enough audio levels, you may want to set the levels by manually putting your radio into transmit, then setting the levels to optimize your audio signal (again you should just see your ALC indicator moving). After this, reset your VOX circuit to trip at that level of input. If you normally use VOX, then the voice setting should be OK, but if you have not set your VOX level; you may have to do so. You can do this by manually putting your radio into transmit, adjusting the transmit level for the above no-ALC level, and then adjusting your VOX gain so the radio goes into transmit when you send it a tone.

[433] You can also toggle between transmit and receive by pressing the Pause/Break key on your keyboard.

Copyright © 2026 Robert C. Furzer K4CY

If your radio does not return to receive, but keeps cycling between receive and transmit when you use VOX, then another input is operating the VOX circuit. Go back to the Volume control mixer, bring up all the controls, and mute them all. Now, uncheck the one(s) you need to output a tone, and the one you need to get a display on the Logger32 screen. Leave all the others muted or turned all the way down. Remember, you can hear the output tone in your speakers.

The optimal setting when using the above interface will usually set your sound card output (Volume and Wave Control settings) very low, and your radio's Mic gain control is at a little lower than normal for SSB operations. If you are unable to control the audio using these controls at reasonable levels, you very likely need to add attenuation between the sound card output and the radio's Mic input. You can also try using a direct audio input on your accessory jack or rear panel (if your radio is so equipped). This may avoid your Mic pre-amp circuit and be a better choice for signal matching, however this may also make it impossible to use your VOX circuit for engaging your transmit and receive modes. Some radios do not connect VOX to the rear input.

Once you've optimized these settings, make a note of the positions of your radio controls as well as the Windows Volume and Recording mixer positions. You may make changes for other software and operating modes, and this makes it easier to reset things for Logger32.

There is an additional adjustment that you can make to tell Logger32 the actual frequency of the sound card's clocking generator. This calibration adjustment holds only for RTTY.

Note: some interfaces are powered from the computer's DTR, DSR, or RTS lines. Logger32 may not work properly with this type of interface. Modify the interface so that it gets power directly from the radio or a separate power supply instead.

37.2 Audio sources

When using digital modes, it is important to avoid transmitting audio from the rig's microphone, or alerts and messages generated by other software such as Windows itself, your email program, web browser or even Logger32.

Modern radios typically offer a DATA mode that automatically disconnects the microphone input/s, enables the DATA or LINE IN input/s, and disables any transmit audio shaping. Check your radio's manual for details.

The microphone and rear panel inputs may both be active at the same time on older radios, typically sharing the same audio input circuit, so you should disconnect the microphone from the radio whenever you operate digimodes. Consider buying or building a switch that makes it easier to select between audio sources. You may also need to disable any transmit audio shaping such as compression/speech processing and narrow transmit filtering to avoid distorting your digimode signal and perhaps generating spurious artifacts, reducing copiability.

On a few strangely-designed radios, the microphone input remains active if you change to transmit using a CAT command, but not if you ground the PTT control line … so consider using the latter to control transmit/receive instead of CAT.

Copyright © 2026 Robert C. Furzer K4CY

38 Single Operator Two Radios (SO2R)

Logger32 supports one operator operating two radios simultaneously, tracking both radios and routing various commands and signals to whichever radio is deemed "active" at that moment.

> "Busy as a one-legged man
> in an ass kicking contest"
>
> *Stephen King*

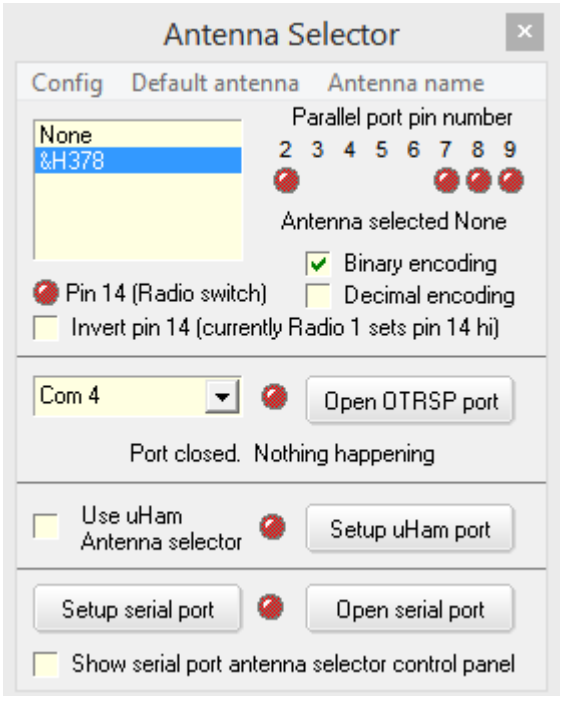
Hinson tip: be aware that when you are transmitting, there will be lots of RF in the air, possibly enough to release the smoke from any sensitive receiver front end, especially one connected to an antenna at that point. Logger32 is not responsible for any smoke damage. It's down to you.

38.1 SO2R radio changeover

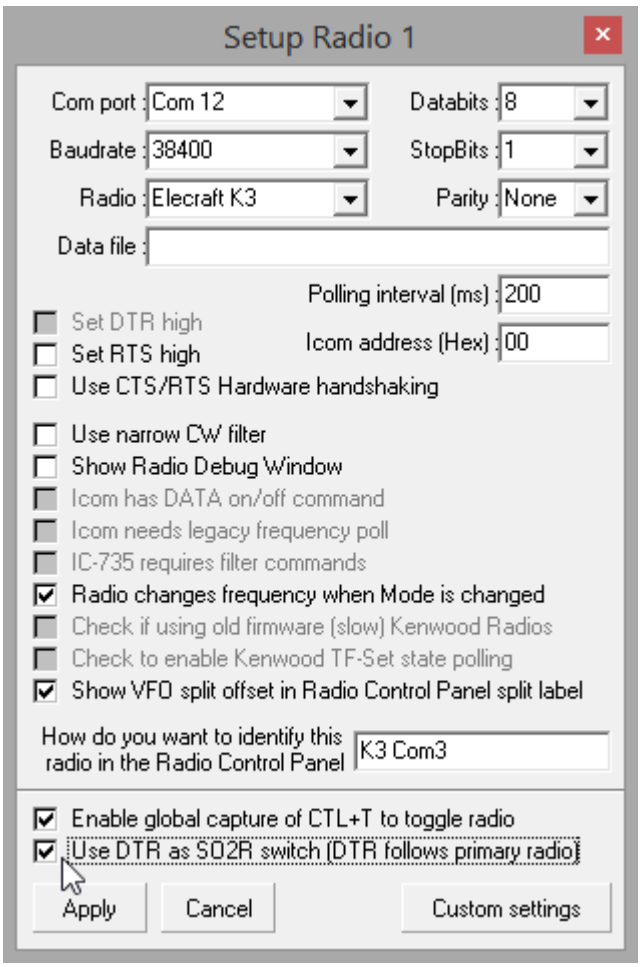
38.1.1 Hardware switching

With a parallel port set up as detailed in the Antenna switching chapter, pin #14 can operate an external relay to swap the PTT, audio and keying connections between radios. By default, Radio 1 is selected by sending a logic high signal (+5v) to pin #14, whereas Radio 2 is selected by grounding pin #14.

Selecting <**Invert pin 14 …**>
under **Setup ⇨ Antenna Selector**
reverses that default ▶



Copyright © 2026 Robert C. Furzer K4CY



Alternatively, the DTR modem control line on one of the CAT serial ports can be set/*un*set according to which radio is selected.

Obviously enough, the single DTR line on a given serial port can *only* be used for PTT control, CW sending *or* SO2R radio selection. Pick one, any one.

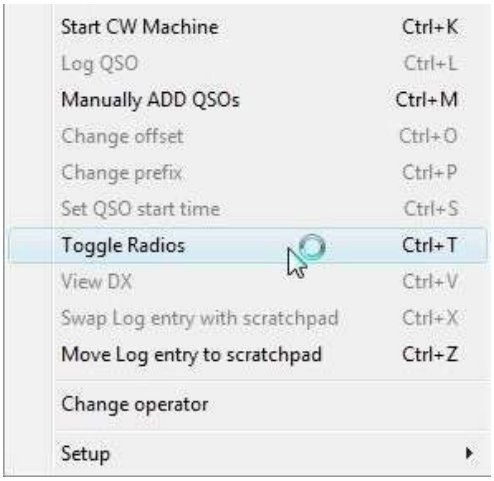
◀ Under **Setup ⇨ Radio ⇨ Radio 1|2 configuration** enable <**Use DTR as SO2R switch (DTR follows primary radio)**>.

Finally, a WinKeyer with dual outputs can divert the PTT line *and* CW keying to the "primary" (active) radio under software command from Logger32 (more below).

38.1.2 SO2R shortcuts and commands to swap active radios

The keyboard shortcut <**Ctrl+T**> swaps over the active and inactive radios[434].

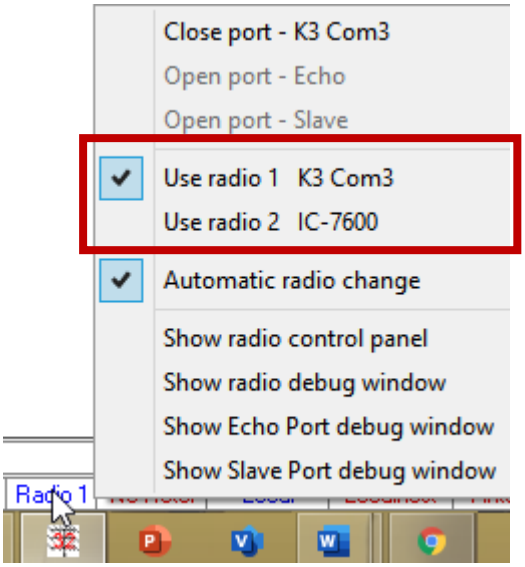
You can also swap radios by right-clicking the log entry pane then clicking <**Toggle Radios**> ▶

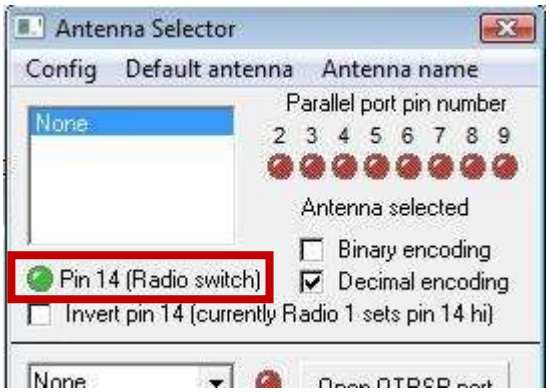


[434] This can be defined as a global shortcut, in which case <**Ctrl+T**> toggles the active radio regardless of which window or program has focus. If the scratchpad has focus and the shortcut is *not* defined globally, <**Ctrl+T**> sets the selected scratchpad entry's QSO start time.

Copyright © 2026 Robert C. Furzer K4CY

Right-click the Radio panel on the status bar ▼ then click to choose a radio[435].





▲ A red/green "LED" in the antenna selector shows which radio is active if you are using pin 14 on the parallel port to control it.



◀ A tooltip when mousing-over the **Radio** panel in the status bar says which radio ports are open.

38.1.3 SO2R toggle radio macro

The immediate macro $**ToggleRadios**$ switches between radios from the Soundcard Data window, Data Terminal and CW Machine window.

38.2 Quick switch

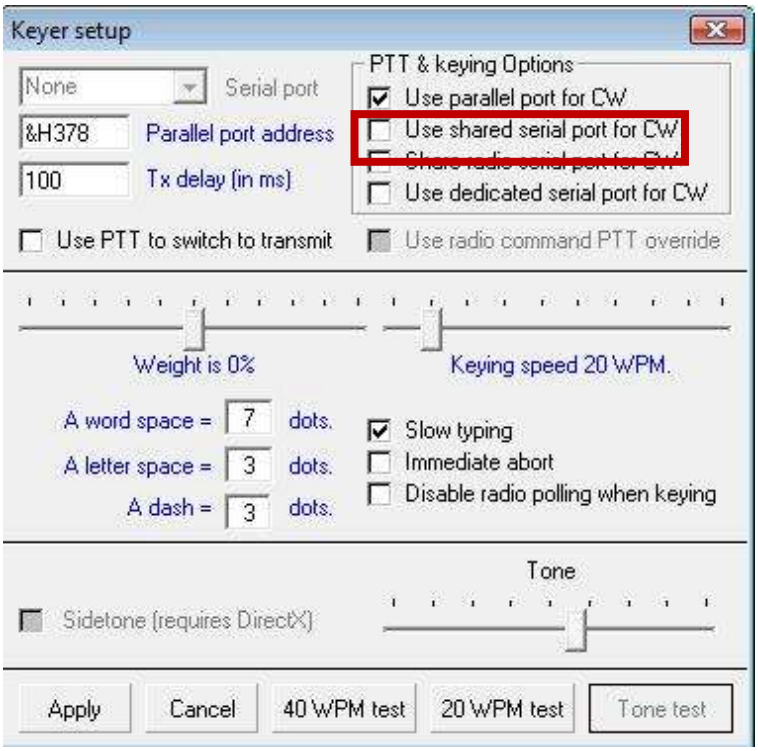
Logger32 can QSY radio 1 or radio 2 when you click DX spots, CAT-commanding whichever radio is specified for the corresponding band in your Bands & Modes table to the spotted VFO frequency and mode. With Quick Switch enabled, it is *also* able to launch the CW machine or JTDX to handle CW and FT# spots, respectively.

[435] 'Automatic radio change' selects whichever radio is defined for the current band/segment in your Bands & Modes table.

Copyright © 2026 Robert C. Furzer K4CY

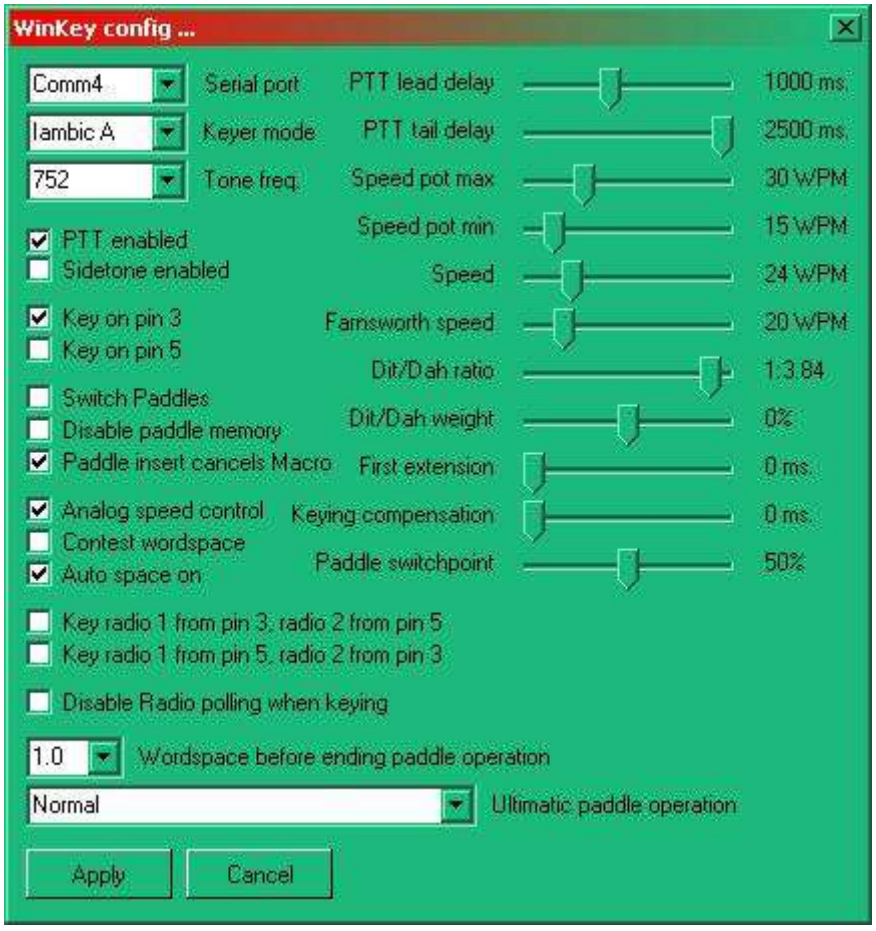
38.3 CW Machine

The CW Machine (*software keyer only*) supports SO2R with PTT and CW keying when using a shared serial port with the radios.



Enable <**Use shared serial port for CW**> to have the RTS (PTT) and DTR (CW keying) signals from the software keyer switch between the two radio COM ports following a <**Ctrl**+**T**> ▶

38.4 WinKeyer



Logger32 supports WinKeyer (both the original ▶ and WinKeyer2). WinKeyers can switch the keyed outputs between two pins on the device. Under normal use the keyed CW is available on pin #3 and pin #5 is a PTT line, but this can be changed such that pin #5 becomes the keyed line by using a particular command.

As an alternative, Logger32 allows the user to set up for the CW keyed output to appear on pin #3 when Radio 1 is in use and to automatically change to pin #5 when Radio 2 is in use. Additionally, the reverse of this is available. If either of these two alternatives is selected, then the “Key on pin #3” and “Key on pin #5” check boxes will be grayed out. Consult the WinKeyer documentation for more.

Copyright © 2026 Robert C. Furzer K4CY

38.5 Additional macros

Macros are sent to either Radio 1 or Radio 2, whichever is selected/active at the time. This means that, when running dual radios, the buttons designed for one radio may not work with the other, particularly if the radios are from different manufacturers with different command sets.

To address this, the $**HexCommand**$ and $**Command**$ macros can include a vertical bar | [ASCII 124] character to divert command text to the left of the bar to Radio 1 and command text to the right to Radio 2, accordingly.

The macro format is:

$**HexCommand [radio1text]|[radio2text]**$

- or -

$**Command[radio1text]|[radio2text]**$

where [radio1text] and [radio2text] are the parameters for each radio. Do not include "[" and "]" in the macro! Logger32 does not strip out spaces so avoid unnecessary spaces either side of the | character as well.

Complex commands can contain both $**HexCommand**$ and $**Command**$ macros *e.g.* if Radio 1 is a Yaesu (requiring $**HexCommand**$) and Radio 2 is a Kenwood (uses $**Command**$), a macro button for a specific function (such as selecting a 500 Hz filter) would need to send different commands to the radios:

- Radio 1 needs $**HexCommand 00 00 00 02 8c**$
- Radio 2 needs $**Command FL009007;**$

The combined macro is $**HexCommand 00 00 00 02 8c**$**|**$**Command|FL009007;**$

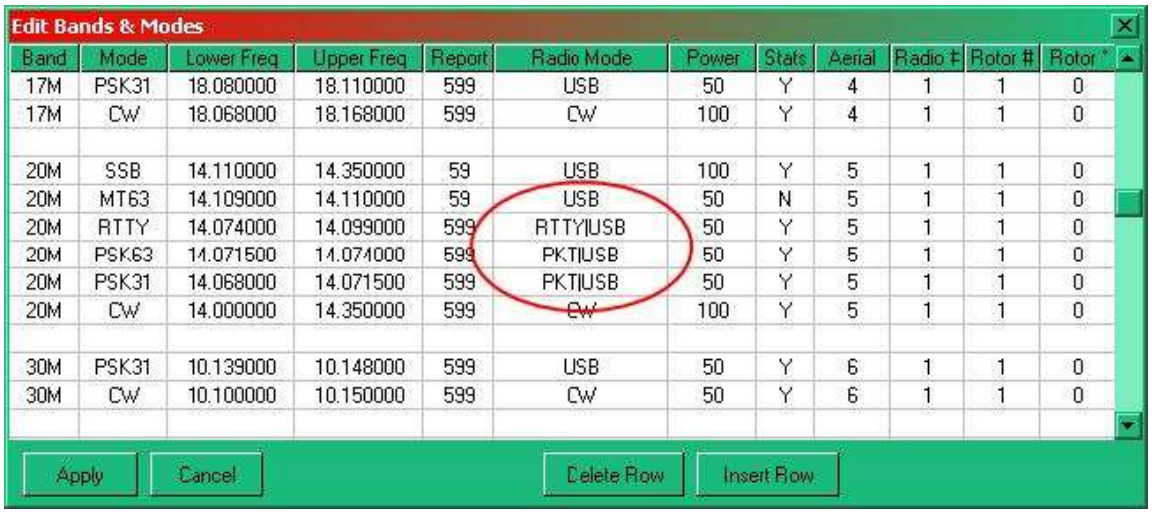
The hex command string 00 00 00 02 8c is sent to Radio 1, while nothing is sent to Radio 2 (to the right of the first | character), followed by a command of nothing for Radio 1 (there's nothing to the left of the second | character) and FL009007; to Radio 2.

If both radios need (say) $**HexCommand**$ instructions but they are different, the combined macro becomes $**HexCommand xx xx xx xx xx|yy yy yy yy yy**$ where xx xx xx xx xx is the command string for Radio 1 and yy yy yy yy yy is the string for Radio 2.

Copyright © 2026 Robert C. Furzer K4CY

38.5.1 Using the Bands & Modes table

The Bands & Modes table supports SO2R radio modes with the same syntax as $**Command**$ and $**HexCommand**$ macros *e.g.* if the Radio Mode column contains RTTY|USB, the radio mode would be set to RTTY on Radio 1 and USB on Radio 2 ▼



As with the $**Command**$ and $**HexCommand**$ macros, if no | is included in the string, the same mode is used by both radios.

38.6 Sound card enhancements for SO2R

The Sound card data window's support for SO2R includes dual displays for both MMTTY ▼ and MMVARI ▼





▲ Click SO2R on either menu to open a secondary sound card display, the captions showing which radio is which[436] ▼





[436] Usually the displays are driven from the left and right audio channels, respectively. As Logger32 cannot determine the audio settings for MMTTY, the audio channel in use is only displayed in the window caption for MMVARI.

Copyright © 2026 Robert C. Furzer K4CY

Under <**Settings**> in the SO2R window is an option to enable or disable the SO2R CAT port:

- When disabled, the SO2R sound card window displays an audio frequency ribbon.
- When enabled, the RF frequency of the second radio is displayed.





It doesn't matter whether you configure ▲ MMVARI or MMTY ▲ since they share this setting.

38.7 Preliminary setup of MMTTY for SO2R

To fully utilize SO2R capability with MMTTY, two different copies of *MMTTY.EXE* need to run simultaneously.

It is necessary to create a new folder *e.g. C\Program Files\MMTTY2*. Into the new folder **copy** from your Logger32 folder the files *MMTTY.EXE*, *USERPARA.INI* and *MMTTY.INI. I*n addition, check that the Logger32 folder contains the file *XMMT.OCX*.



◀To use the full capabilities of SO2R MMTTY, click **Settings ⇨ Setup path to SO2R MMTTY.EXE**, then specify the folder where you have copied the copy of *MMTTY.EXE* and its supporting *.INI* files ▼



MMVARI does not require a special setup for SO2R.

Copyright © 2026 Robert C. Furzer K4CY

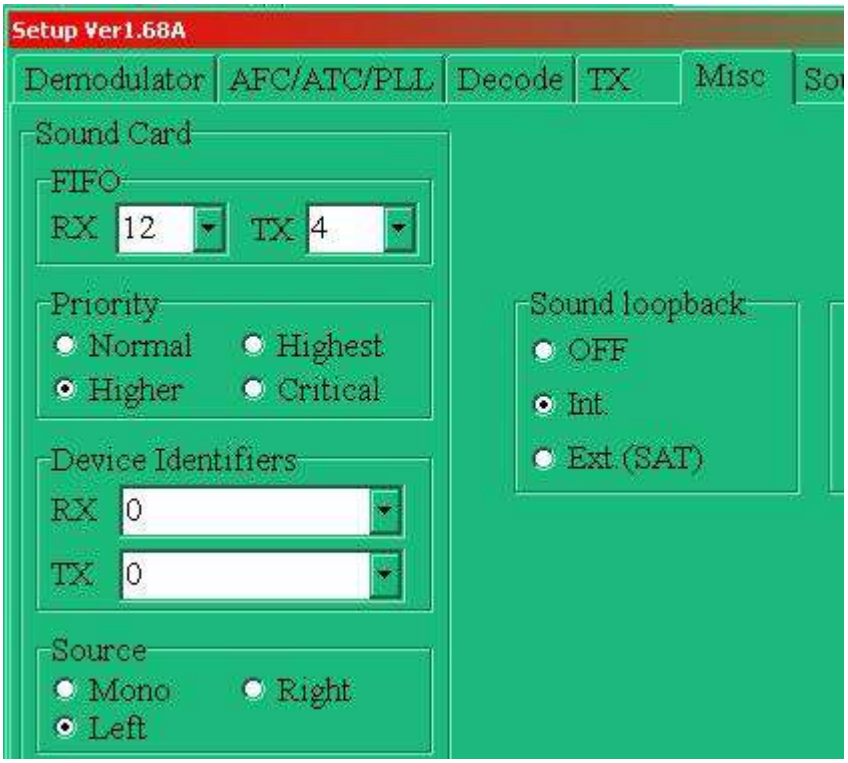
38.8 Sound card selection

Logger32 can use different sound cards for different digimode engines ▼





The device numbers will be required for the MMTTY set-up.



38.9 Selection of audio channel

Under normal use (single radio input), the soundcard generally uses the left channel. For SO2R both channels are used independently, usually one from each radio. Logger32 gives you the option to configure the sound card to use either the left or the right channel for Radio 1. Radio 2 simply uses the other.

Copyright © 2026 Robert C. Furzer K4CY

38.9.1 MMVARI

See **Settings ⇨ MMVARI Settings ⇨ SO2R Audio channel**. This option only becomes available if SO2R has been selected, and applies to both Radio 1 and 2 windows: if Radio 1 is set to use the left channel, Radio 2 use the right channel and *vice versa* ▼



38.9.2 MMTTY

In MMTTY, each radio has to be individually associated with the left or right channels.

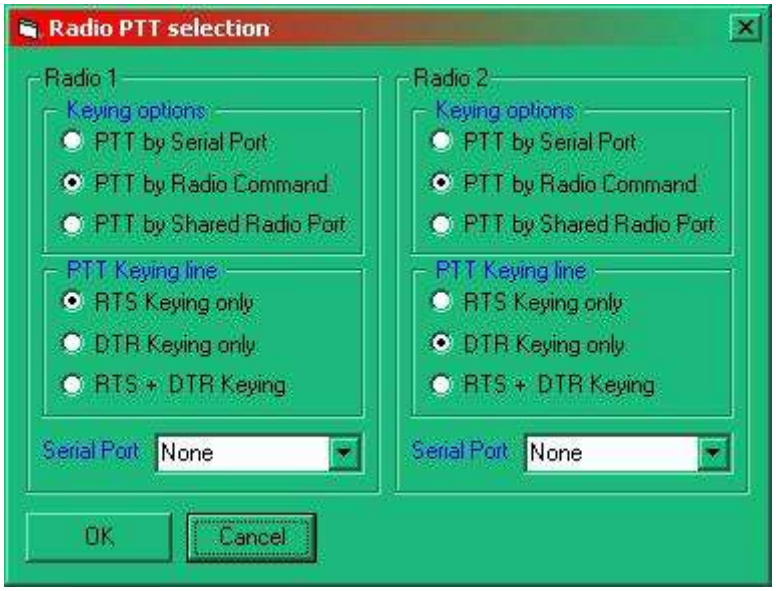
For Radio 1, click the spanner icon in the Radio 1 window and select Mono/Left/Right on the <**Misc**> Tab. Repeat for Radio 2 from *its* window.

38.10 Radio PTT port selection

38.10.1 MMVARI and MMTTY

The radio PTT selection caters for SO2R operation.

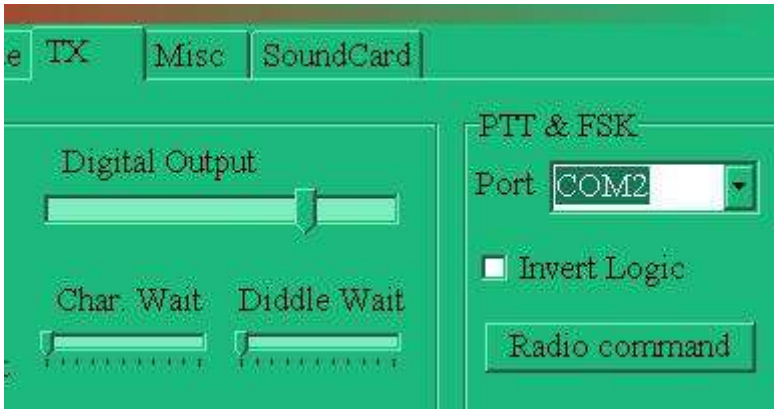
It is configured from the Soundcard data window menu **Settings ⇨ Radio PTT options** ▶



Both MMTTY and MMVARI PTT are configured at the same time. Do not attempt to change the port settings from MMTTY setup (TX) panel as these settings will not be saved.

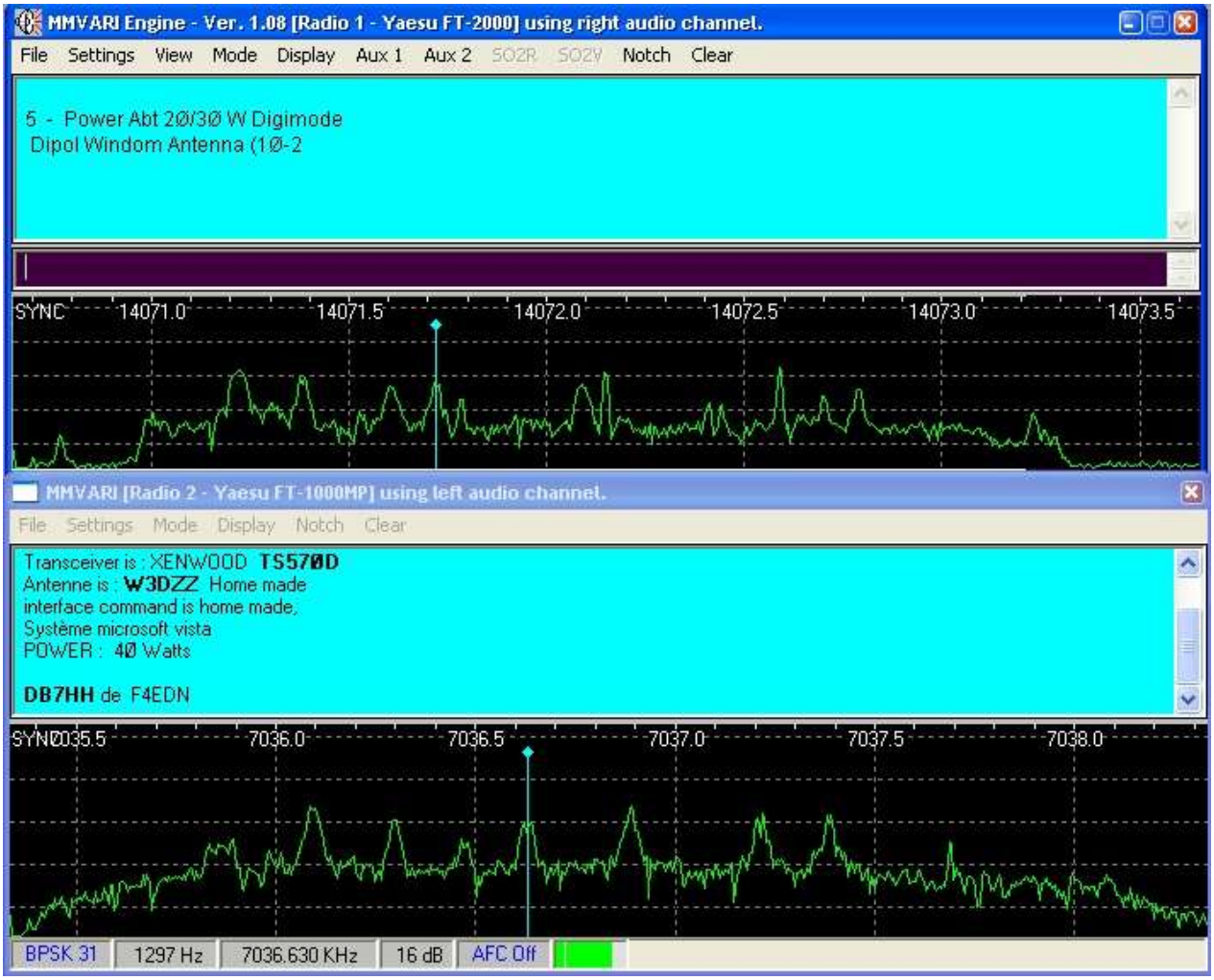
38.10.2 MMTTY only

For MMTTY, as with the left and right audio channels, the PTT ports have to be set for each radio from each of the SO2R windows. For Radio 1, click the setup spanner icon in the Radio 1 window and select the COM port on the TX tab. Repeat for Radio 2 from its window ▶



Copyright © 2026 Robert C. Furzer K4CY

The screen shots show the sound card being used with MMVARI in SO2R ▼



The main and secondary screens are receiving different audio streams from two different receivers - one tuned to 14 MHz and the other to 7 MHz.

The main and secondary screens can receive different modes. In the upper screen shot, a PSK signal from Radio 1 is being decoded in the main window while the secondary window shows a RTTY signal being decoded from radio2.

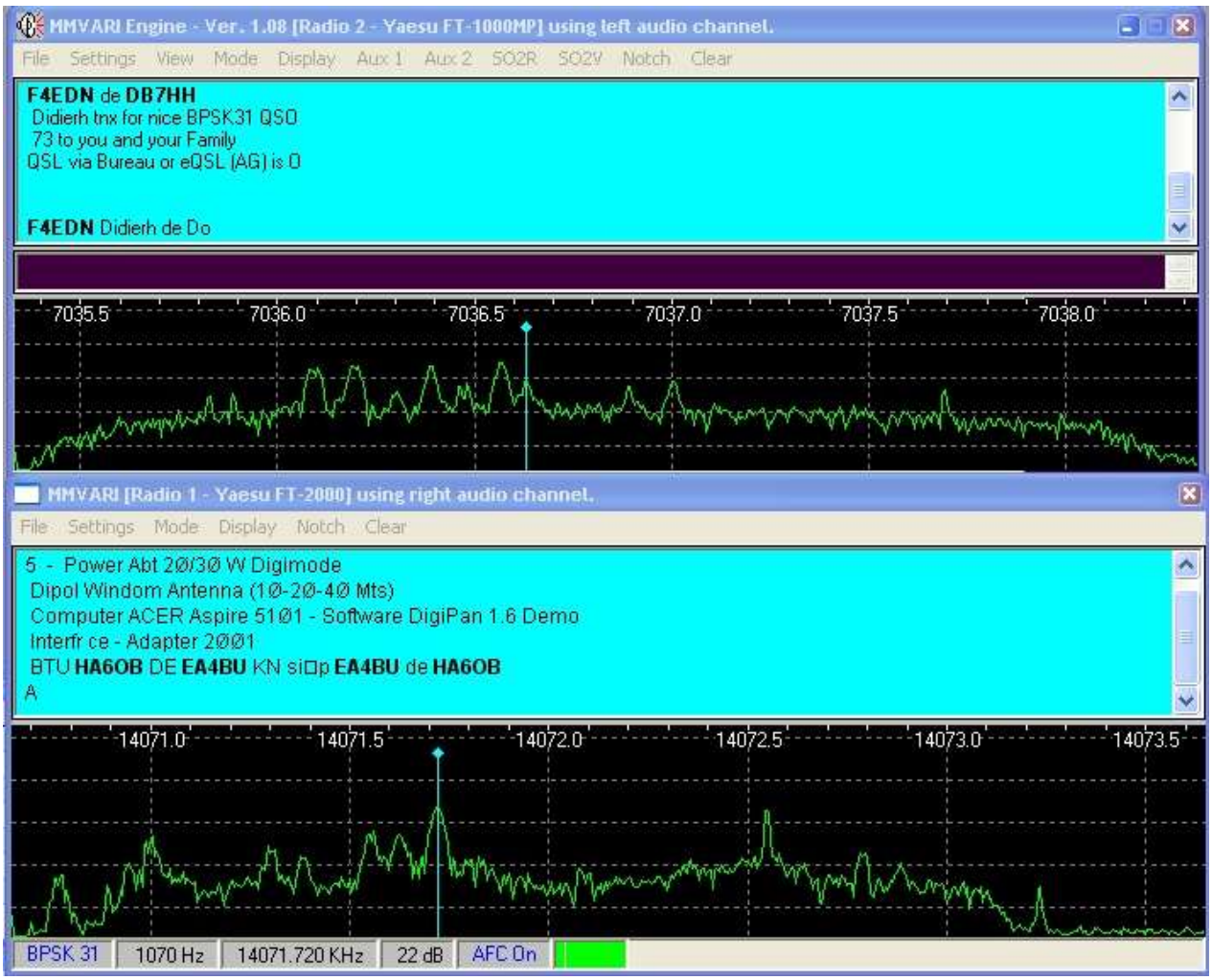
Only the primary SO2R window has macro keys.

The frequency ribbon in the SO2R window will only display an RF frequency if:

- There is a functioning radio connected to a second radio port; and
- The SO2R port is activated.

The screen shot below has been taken using the same set up as the picture above except that in this case the Radio 1 and Radio 2 signals have been swapped using either the <**Ctrl**+**T**> function with the focus on the log entry pane OR selecting a different radio using the **Radio** panel in the status bar. Not only has the audio channel been swapped but the received text AND the mode settings have changed over as well.

Copyright © 2026 Robert C. Furzer K4CY



These two digital windows do not have to be derived from physically separate radios of course. There is no reason why they should not be fed with audio from the main and sub-receivers on a fancy transceiver. However, although a <**Ctrl**+**T**> will switch the radio panes on the screen, the radio itself will not change to transmit from the other VFO. This is useful when chasing DX working split: you can monitor the DX station in one pane and tune around using the second VFO or receiver, looking for the stations he is working.

38.11 Setting up to use two radios

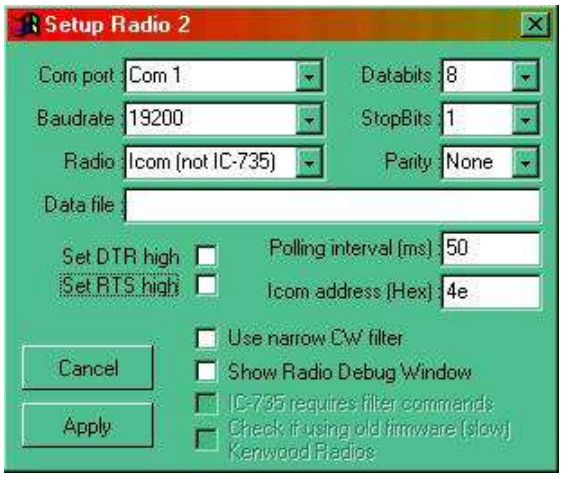
Logger32 supports the transfer of frequency, mode and commands to either of two radios via CAT, with manual and automatic switching between the two. However, Logger32 cannot automatically switch your audio and/or other hardwired radio connections (*e.g.* the CW paddle or keyer and PTT controls): you will have to handle this separately.

Hinson tip: be careful how you use this setup. Using the Radio Reset menu item (right-click an entry in the DX Spots pane), or **Setup** ⇨ **Radio**, can cause you to log QSOs on the wrong band or mode. It's very easy with two radios running to make a QSO on one radio (*e.g.* on 20m CW) but accidentally log it while Logger32 remains connected to the *other* radio (perhaps on 6m FT8). You can correct your log later, provided you notice and remember the details.

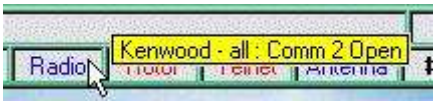
Copyright © 2026 Robert C. Furzer K4CY

1. Setup the configuration details for each radio. Use **Setup ⇨ Radio ⇨ Radio 1|2 configuration** ▼ to display the Setup Radio 1|2 form ▼





2. Click to tick **Setup ⇨ Radio ⇨ Use Radio 1|2**. Mouseover the **Radio** panel in the status bar for a tooltip telling you what's going on ▶



3. Right-click the **Radio** panel and change radios by selecting the appropriate menu item <**Use radio 1|2**> ▶



4. With <**Automatic radio change**> enabled here, Logger32 automatically commands whichever radio is identified in the **Radio #** column of the Bands & Modes table for a given frequency *e.g.* when you click DX spots.



Every band segment entered in the bandplan ▲ *must* have an entry in the "Radio #" column.

In this example, Radio 1 is configured for HF while Radio 2 is configured for VHF/UHF. Selecting any 6m, 2m, or 70cm spot switches automatically to Radio 2, whereas selecting spots for any 10m segment switches to Radio 1.

You are not limited to just the Band column. Any defined band segment can use either radio. You might, for instance, dedicate one radio for legacy modes such as AM and another for digimodes. You can also manually switch between Radio 1 and Radio 2 with <**Ctrl**+**T**>.

Copyright © 2026 Robert C. Furzer K4CY

38.12 SO2R FAQs

Q. I have *lots* of CAT-capable radios. If I connect them all to my shack PC, can Logger32 switch between them on demand?

A. No, at least not as simply as it can switch to either of two radios using the SO2R functions in this chapter. There is no obscure SO3R, SO4R or SO#R function lurking within Logger32. By design, Logger 32 *presumes* you have no more than two ears, two eyes, two arms, two legs and two hemispheres of the brain, barely sufficient (in my case at least) to cope with SO2R.

Instead, you can:

- Setup separate configurations according to whichever *pair* of radios you want to use at the time.
- Use additional shack PCs also running Logger32 to control and log QSOs made on additional radios (you may want to merge the logs or perhaps try parallel logging).
- Install hardware switches to select between your radios – preferably identical models of radio or at least those with similar CAT command structures, avoiding the need to reconfigure Logger32 after switching radios.
- Stop greedily acquiring and using so many radios at once! Pick your favorites and connect those. Use the others like we always used to, pre-CAT. Twiddle your knobs. Write QSO details in an actual log book.

Q. I have decided to define my shiny new radio as "Radio 2" and no longer even power up the tired old one, still known as "Radio 1" for sentimental reasons. I have configured the port correctly and when I *open port* for Radio 2, I see its VFO frequency on the log entry pane. Fantastic! So … how come when I click a DX spot, I get an obscure error message, the CAT connection to the new radio is dropped and it does not QSY to the spot frequency as expected?

A. When you click a DX spot, Logger32 quickly checks which band it is on, selects whichever radio you have defined for that band, and attempts to command it to the spotted frequency using its CAT connection. Let me guess:

1. Your Bands & Modes table still lists the tired old Radio 1 against every frequency in the Radio # column, hence Logger32 is *trying* to command Radio 1 by default.
2. You have, at some long since forgotten prior point (perhaps before you even owned two radios), enabled <**Automatic radio change**> from the <**Radio 1|2**> panel on the status bar.

To stop this behavior, your options include:

- Redefine your shiny new radio as "Radio 1". Retire the old one. Get over it. Or …
- Diligently update your Bands & Modes table to refer to "Radio 2" for all the bands and modes you intend to use on your shiny new radio. If you ever want to revert to the tired old radio, you will have to reverse these changes just as diligently. Or …
- Right-click the <**Radio 1|2**> field on the status bar, then *de*select <**Automatic radio change**>, leaving radio 2 selected in the same menu, with its port open. This is easy to do and reverse! Also, right-click your BandMaps to check that "Use radio 2" is selected.

Copyright © 2026 Robert C. Furzer K4CY

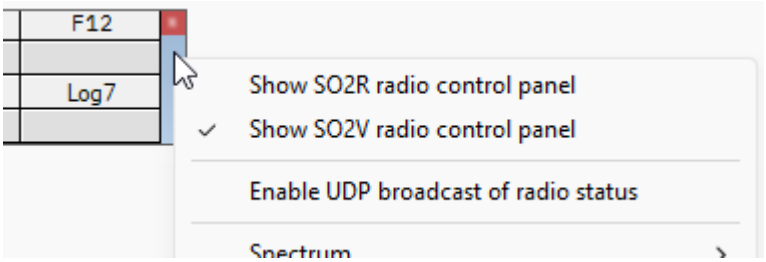
39 Single Operator Two VFOs (SO2V)

> "The best things in life come in pairs"
>
> *Anonymous*

Basic support for SO2V is provided in Logger32. There is no A-B switching and no display of the TX or RX VFOs *etc*. It is assumed that VFO B tunes a sub-receiver.

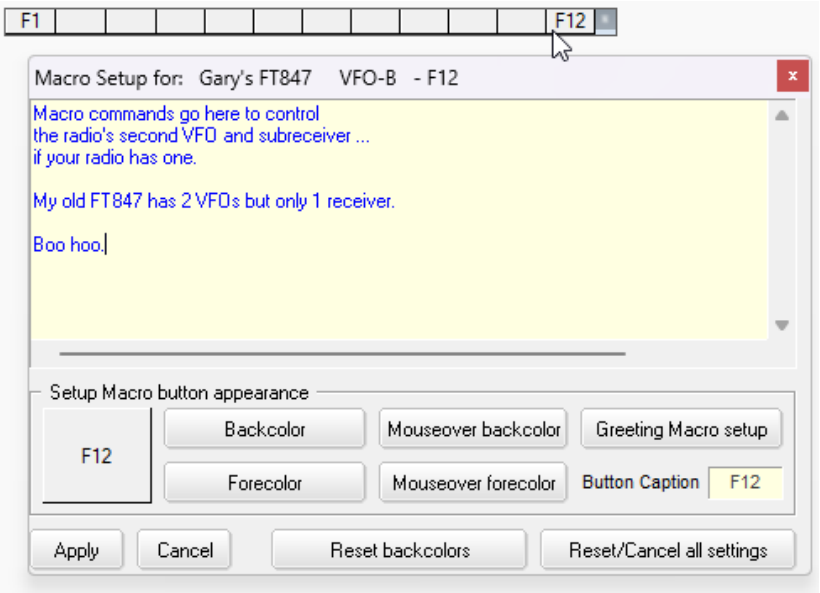
Your SO2V radio must output stereo audio using one channel for the main receiver and the sub-receiver on the other.

39.1 Operating SO2V



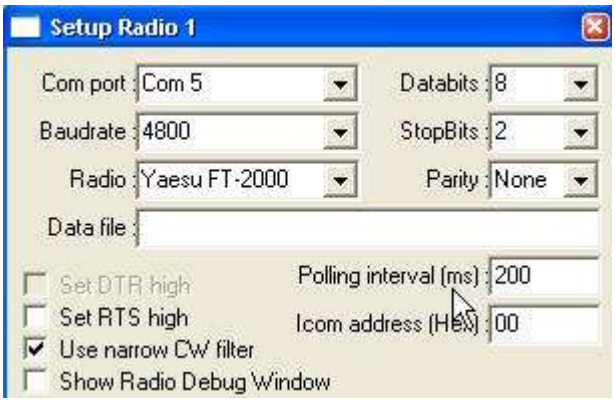
◀ To launch Logger32's SO2V function, open the Radio Control Panel, click the area just below the RCP's corner X to open the menu, then click <**Show SO2V radio control panel**> - basically a set of macro buttons ▼

Having opened the SO2V radio control panel, you can right-click any button then compose macros to control VFO B, for example using the appropriate CAT commands to change the VFO B filtering or audio balance in the same way as for the normal RCP.



Hinson tip: as with other sets of macro buttons in Logger32, labelling unused buttons with their function key numbers is a handy prompt if you are as scatterbrained as me.

Copyright © 2026 Robert C. Furzer K4CY



◀ If you experience CAT problems when using SO2V, check the CAT data using the radio debug window. You may need to increase the polling interval through **Setup ⇨ Radio 1|2 ⇨ Radio 1|2 configuration**.

39.2 Sound card SO2V

While using MMVARI or MMTTY, another SO2V menu option is available there too▼



Click <**SO2V**> on the menu to open a second sound card display. The window captions show the radios and (in MMVARI but not MMTTY) the audio channels being used. The primary (VFO A) display has some additional options▼



Normally, one of the displays is driven from the left audio channel and the other from the right. If you have *two* SO2V-capable radios and suitable changeover switching, it is possible to run either SO2R or SO2V by using the <**Ctrl**+**T**> changeover control as described in the SO2R chapter.

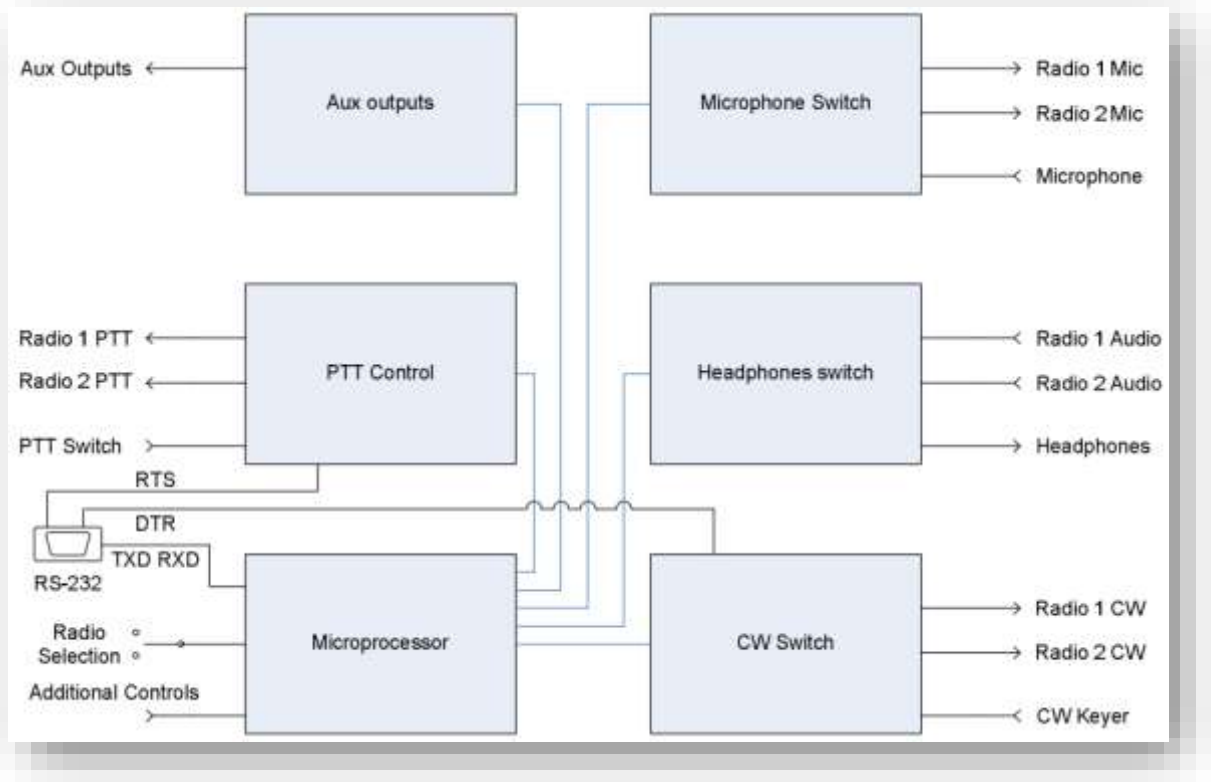
Copyright © 2026 Robert C. Furzer K4CY

40 Open Two Radio Switching Protocol (OTRSP)

> "Open protocols are the glue that connects the internet, enabling seamless communication and collaboration"
>
> *Vint Cerf*

The **O**pen **T**wo **R**adio **S**witching **P**rotocol is a standard that defines commands to configure and operate a **S**ingle **O**perator **2** **R**adio station through a serial link from the PC to an intelligent SO2R interface ▼

OTRSP control macros are available in the **R**adio **C**ontrol **P**anel, the Sound card data window and the CW Machine, and Logger32's automatic antenna switching function supports ORTSP.



Copyright © 2026 Robert C. Furzer K4CY

40.1 About OTRSP

OTRSP commands are used to control functions such as:

- **Input selection**: enabling the microphone or CW keyer for the active radio, while disabling it for the other radio.
- **Stereo audio selection**: sending receiver audio[437] from the non-active radio to the headphones while muting or reducing the level of the active radio, and/or changing the mixture of audio sent to the left and right ears.
- **PTT**: routing the Push To Talk lines from, say, a footswitch and Logger32's CW Machine to switch *one* of the radios to transmit, with an XOR lockout automatically preventing the *other* radio from transmitting simultaneously.
- **CW keying** of the active radio by grounding the DTR line on the serial link.
- **Voice keying** of the active radio, if the SO2R box contains a DVK or can trigger those built into some radios.
- **Aux** controls for the auxiliary functions on some SO2R setups, such as automatic antenna switching and reconfiguring modes, filters, sub-receivers *etc*. for either radio.

An OTRSP command string of up to 16 UTF-8 characters is typically comprised of:

- Optionally a question mark ? denoting a query typically anticipating a current status response, as opposed to a command to change something.
- A keyword of up to 6 letters denoting the function *e.g.* TX means ground PTT to transmit.
- Any parameters (items and values) *e.g.* which radio will transmit – either 1 or 2.
- A carriage return (character 13) concluding the command string.

The OTRSP serial link operates at 9600 baud with no parity, 8-bit data and one stop bit. It is a full duplex link: the SO2R interface receives and actions commands from the PC at the same time as it responds to queries sending status information back, with buffers at both ends allowing a few commands and responses to be queued-up if necessary.

The system is designed for responsiveness. OTRSP commands should be executed or passed-on to the radios and other devices, within a quarter of a second of being sent to the SO2R controller.

40.2 OTRSP macro command format

Two OTRSP macro command formats are supported.

40.2.1 $OTRSP xxx|yyy$

This is the main OTRSP macro to send one or more commands to the OTRSP port *e.g.*

$OTRSP TX1|RX1$

Sends an OTRSP command to the SO2R interface to put the transmit and receive focus on Radio 1.

[437] Many of today's radios output multichannel audio *e.g.* from the main and sub-receivers, possibly with digital signal processing to generate pseudo-stereo effects by controlling the phases and frequency ranges sent to each ear. With two of such radios in use, the audio switching and mixing becomes complex and can be confusing for the operator, especially when busy or exhausted: this is where the SO2R controller, plus OTRSP, can really help through automation.

Copyright © 2026 Robert C. Furzer K4CY

40.2.2 $onRxOTRSP xxx|yyy$

This is a special macro for the CW Machine and Sound card data window. A typical script (command sequence) would be:

$**OTRSP TX1|RX2**$

CQ CQ DE K4CY K4CY

$**onRxOTRSP TX1|RX1**$

This may also be written on one line as:

$**OTRSP TX1|RX2**$ CQ CQ DE K4CY K4CY $**onRxOTRSP TX1|RX1**$

This script entered in one of the CW Machine or Sound card data window buttons will:

- Put the transmit focus on Radio 1 with the headphones connected to Radio 2, typically listening on another band.
- Transmit the CQ message on Radio 1 … while the operator listens to Radio 2.
- At the end of transmission, switch the receive audio back to Radio 1 to listen for replies to the CQ.

40.3 OTRSP command examples

As many commands as are required can be strung together, separating each command with a vertical bar "|" (pipe symbol, typed as <**Shift** + **Backslash**> on most keyboards).

Do not add <**CR**> at the end of the OTRSP command: Logger32 does that for you.

The following common OTRSP commands can be sent to the SO2R interface through the OTRSP port. These macros have been tested and work OK on the YCCC SO2R+ interface box.

40.3.1 OTRSP TX commands

- **TX1**: switch transmit focus to Radio 1.
- **TX2**: switch transmit focus to Radio 2.

40.3.2 OTRSP RX commands

- **RX1**: connect Radio 1 to both headphones (left and right ear).
- **RX2**: connect Radio 2 to both headphones.
- **RX1S**: connect Radio 1 to the left headphone and Radio 2 to the right headphone (stereo). If stereo is not possible connect Radio 1 to both headphones.
- **RX2S**: connect Radio 1 to the left headphone and Radio 2 to the right headphone (stereo). If stereo is not possible connect Radio 2 to both headphones.
- **RX1R**: connect Radio 1 to the right headphone and Radio 2 to the left headphone (reverse stereo). If reverse stereo is not possible connect Radio 1 to the left headphone and Radio 2 to the right headphone (stereo). If stereo is not possible connect Radio 1 to both headphones.
- **OTRSP RX2R**: connect Radio 1 to the right headphone and Radio 2 to the left headphone (reverse stereo). If reverse stereo is not possible, connect Radio 1 to the left headphone and Radio 2 to the right headphone (stereo). If stereo is not possible either, connect Radio 2 to both headphones.

Copyright © 2026 Robert C. Furzer K4CY

40.4 <Ctrl+T> and grave accent key switching

The <**Ctrl+T**> hotkey[438] or $**ToggleRadios**$ macro is normally used to switch the transmit and receive focus between radios. You can change the active radio *and* change other settings at the same time by defining additional programmable parameters.

The ` grave accent key (to the left of the 1 on most keyboards) is a hotkey that toggles between two OTRSP commands. In contest or pileup situations, this can be useful to toggle between two operating arrangements *e.g.*

Active radio audio to both ears ⇄ Left radio to left ear, right radio to right ear

For instance, most of the time you may prefer to concentrate fully on a weak DX station operating split, listening to the DX in both ears (perhaps with diversity reception or pseudo-stereo DSP effects). Occasionally, however, you may want to listen on a second receiver monitoring a *different* DX pileup on another frequency or band.

Set the commands for these two features in the right-click menu of the **R**adio **C**ontrol **P**anel as shown below, similar to the DX spot macros menu.

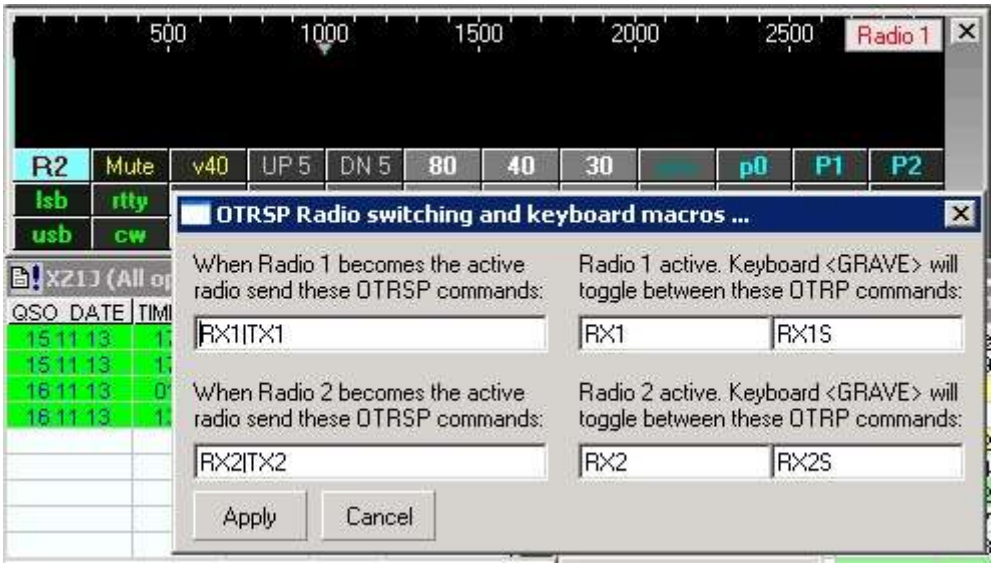


◀ Click <**OTRSP Macros**> on the menu, then define the OTRSP commands for each radio when toggling radios and when using the grave accent key to toggle between the two macros (successive presses alternate between them).

[438] If the scratchpad has focus, <**Ctrl+T**> normally sets the QSO start time for the selected row *unless* <**Ctrl+T**> is configured as a global hotkey to toggle radios.

Copyright © 2026 Robert C. Furzer K4CY

In this example ▼ ...



... the following commands are defined:

Function	OTRSP command string	Description
When Radio 1 becomes the active radio	RX1|TX1	Puts the receive and transmit focus on Radio 1 when that radio is made active
When Radio 2 becomes the active radio	RX2|TX2	Puts the receive and transmit focus on Radio 2 when that radio is made active
Grave accent key toggle with Radio 1 active	RX1 ⇄ RX1S	Toggles between Radio 1 audio to both ears, or Radio 1 to one ear and Radio 2 to the other
Grave accent key toggle with Radio 2 active	RX2 ⇄ RX2S	Toggles between Radio 2 audio to both ears, or Radio 1 to one ear and Radio 2 to the other

These commands are enabled when you click <**Apply**>, and remain active whether the **R**adio **C**ontrol **P**anel is open or is closed.

Copyright © 2026 Robert C. Furzer K4CY

41 Antenna rotators (rotors)

> "Sometimes a change in direction leads to the most beautiful destination"
>
> *Anonymous*

Logger32 automatically calculates the heading and distance from your home QTH to the DX station when there is a recognizable prefix or callsign in the Call field of the log entry pane (*e.g.* while you are logging someone, or after you have clicked a DX spot).

If a supported and connected rotator is configured for the band or band segment you are currently using, Logger32 turns the rotator to the directions shown on **DX information bar** the when you hit the <**Crtl**+**A**> or <**Alt**+**A**> shortcuts for short path or long path, respectively.



Logger32 communicates with the rotator through a serial (COM) port, sending it the commands to turn the rotator and receiving back the azimuth (heading).

41.1 Supported rotators

Logger32 supports azimuth control (not elevation) for the following popular digitally-controlled auto-positioning rotators, plus others that use the popular **HyGain DCU-1** protocol (*e.g.* the Ultrabeam UB-ONE control set includes DCU-1 commands):

- Alfa SPID (*some* models use the HyGain DCU-1 protocol – check the manual)
- EA4TX ARS
- EZ Rotator Control ERC-M
- Green Heron Engineering RT-20 or RT-21 (use HyGain DCU-1)
- HyGain DCU-1
- Idiom Press - RotatorCard DXA, SDX or Rotor-EZ (all use HyGain DCU-1)
- M2 RC2800 series
- ProSis Tel C or D
- *Micro*HAM ARCO (networkable smart rotator controller)
- Yaesu GS-23 (GS-232A or B)
- Yaesu GS-232B AZ/AZ A or B
- Zelpro SAK-232

Logger32 even supports automatic rotator positioning for old analog rotators through a digital interface such as the cool €89 Arduino-based digital rotator controller kit from RemoteQTH.com.

Copyright © 2026 Robert C. Furzer K4CY

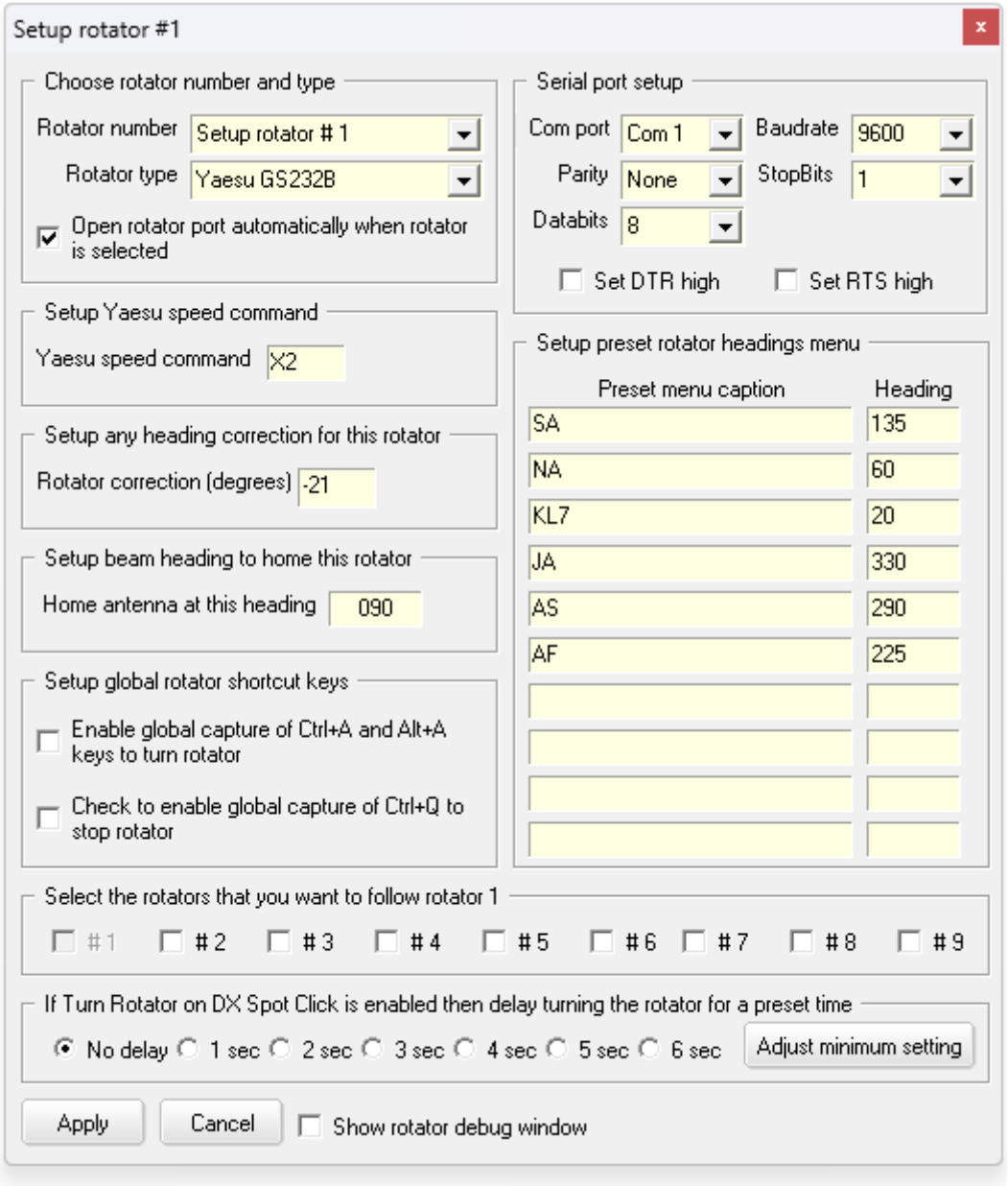
41.2 Which rotator for which band?

Logger32 will only attempt to turn rotators on the bands for which you have specified them … so first tell Logger32 which rotators (if any) are used for each band or band-segment using **Tools ⇨ Setup Bands & Modes**. See the Bands & Modes table section for instructions.

If the radio is tuned to a band where (according to your Bands & Modes table) you do *not* use a rotator, the rotator control serial port has nothing to do and (if so configured) is closed automatically. It magically springs back to life if you QSY to a band *with* a defined rotator.

41.3 Rotator configuration

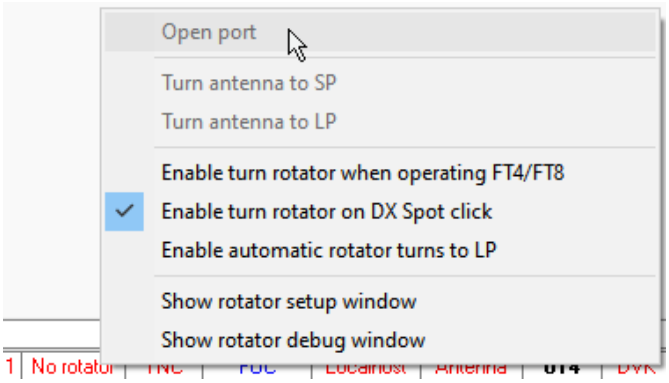
Configure a rotator using **Setup ⇨ Rotator ▼**



- **Rotator number**: choose which rotator to set up. If you have an extensive antenna farm, Logger32 can control up to 9 rotators, so there are up to 9 setups to complete.

Copyright © 2026 Robert C. Furzer K4CY

- **Rotator type**: specify the rotator make and model which determines the protocol and hence control command set required.
- **Yaesu speed command**: check the manual for your rotator to see if it supports the Xn rotation speed command, where n is between 1 (slowest) and 4 (fastest). If you are turning a big stack, the rotator needs longer to get up to speed, turn and slow down approaching the set point. Attempting to turn it too fast increases strain and wear.
- **Open rotator port automatically when rotator is selected**: this is the easy and obvious option. If *un*ticked, you have to open or close the rotator port manually by right-clicking the rotator panel on the status bar and then clicking <**Open|Close port**> ▶



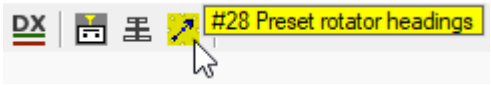
- **Serial port setup**: set the serial communications parameters for the COM port that connects to your rotator. You may need to check in Windows Device Manager or (Shock! Horror!) read the rotator manual for basic information such as the default baudrate.
- **Set DTR|RTS high**: select the appropriate one if you need the PC to supply power to your rotator interface, or if the rotator itself needs hardware handshaking.
- **Rotator correction (degrees)**: Logger32 always calculates headings relative to **true North**. Use the rotator correction to compensate if your antenna/rotator does not actually point to true North when indicating North on the rotator control box *e.g.* if you aligned the antenna and rotator to *magnetic* North, or if the stub mast has slipped in the rotator, or if it is mounted crosswise or backwards to the indicated direction. For example, if your antenna actually points to 21° (North North East) with 0° (North) showing on the control box, correct the offset with *minus* 21° (-21). If it actually points West when showing North, use plus 90° correction[439].
- **Home antenna at this heading**: is the azimuth where you want to 'park' your antenna at the end of the day *e.g.* towards the East (90°) if you normally beam towards the rising sun most mornings, or downwind if you live in a wind tunnel. With the focus on the log entry pane, <**Ctrl**+**H**> turns the rotator to the specified position, sending the antenna **H**ome.
- **Setup global rotator shortcut keys**: <**Ctrl**+**A**> and <**Alt**+**A**> (for 'turn to the short path' and 'turn to the long path', respectively), plus <**Ctrl**+**Q**> (for **Q**uit turning), may be configured for global capture so you can turn or stop the rotator assigned to the current band using those hotkeys, *regardless* of which window or program has focus at the time.

> **Hinson tip**: global capture may result in your triggering Logger32 to turn a rotator, even if Logger32 is running quietly in the background and you actually meant to do something in the foreground program – such as marking all the text in a document or email using <**Ctrl**+**A**>.

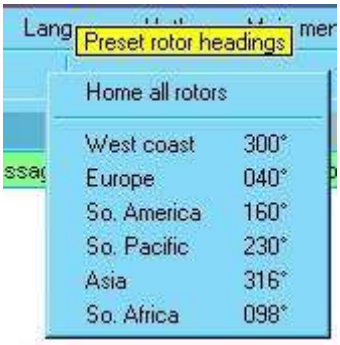
[439] Remember that the direction pointer/wedge and the digital readout show the antenna direction, not necessarily the rotator direction. Logger32 takes account of any fixed offsets applied to the rotator setup.

Copyright © 2026 Robert C. Furzer K4CY

- **Rotator presets**: if your beam isn't a huge multi-element monster on a tall tower, the half-power beamwidth may be 60° or more[440], so there's no need to 'precisely point' towards each DX station when 'roughly right' is generally good enough. Rotator presets let you turn the antenna to your preferred directions (including 'Home'), easily.



◀ Click Toolbar icon #28 then click your chosen preset ▶



- **Show rotator debug window** opens a diagnostic window showing the commands being sent to the rotator, and the rotator responses *e.g.* ▶

  The debug window can also be opened by right-clicking the **Rotator** panel in the status bar.

  If you just mouseover the **Rotator** panel, a tooltip tells you which (if any) rotator is active on the current band, and whether the associated port is open ▼





41.4 How to rotate the antenna

Having decided whether to try the short or long path directions, rotate the antenna for the current band manually using hotkeys or macros, or by clicking the map, or have Logger32 figure out the direction to the DX and turn it automatically.

41.4.1 Log entry pane - hotkeys

With the focus on the log entry pane and a callsign in the Call field, the antenna for that frequency can be turned using:

- <**Ctrl**+**A**> - turns to the short path direction *i.e.* the direction of the shortest great circle distance from your station location to the DX, the most direct "as the crow flies" heading.
- <**Alt**+**A**> turns to the long path *i.e.* the longest great circle distance to the DX (the opposite or reverse heading to the short path, 180° different).

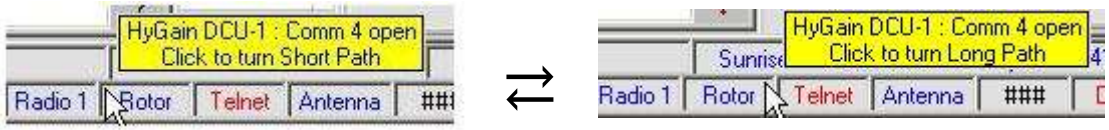
41.4.2 Mouse clicks

This method works with the focus in any Logger32 window.

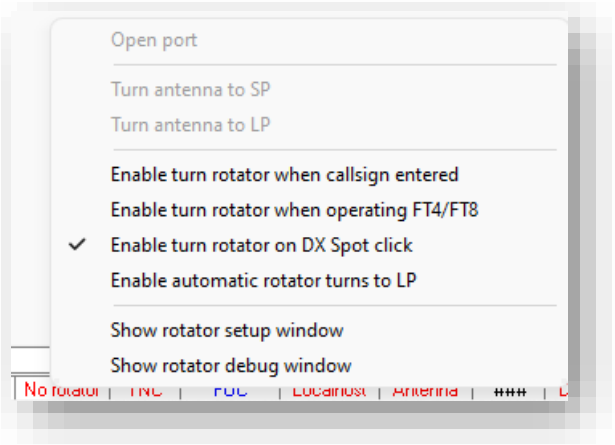
[440] A rotatable dipole or delta loop less than a wavelength above ground may have a 'beamwidth' of more than 90° - possibly almost omnidirectional with weak nulls 'off the ends'.

Copyright © 2026 Robert C. Furzer K4CY

Point the mouse cursor at the **Rotator** panel on the status bar – more specifically, point at the left side of the panel to bring up a tooltip with <**Click to turn Short Path**> or point at the right side of the panel for the long path option▼ Either way, click to send the command.



You can also *right*-click the **Rotator** panel, then click <**Change antenna direction SP|LP**>[441] ▶



Having right-clicked the **Rotator** panel on the Status bar across the bottom of your screen, you have a few options:

- **Open|Close port:** manually enables or disables the serial port used to connect to and control your rotator – more specifically, the rotator specified for the present band in your Bands & Modes table, and its corresponding port configured using the instructions above.
- **Turn antenna to SP:** automatically turn the antenna configured for the present band to the calculated short path direction from your station location (home QTH) to the DX station currently in the log entry pane. See Auto-pointing for more.
- **Turn antenna to LP:** ditto except it turns the antenna to the opposite direction (180˚ different) for the long path.
- **Enable turn rotator when callsign entered:** having entered a callsign in the Call field of the log entry pane, moving the focus away from that field can trigger Logger32 to turn the antenna towards the station you are logging.
- **Enable turn rotator when operating FT4/FT8**: while using JTDX|WSJT-X through the UDP BandMap, this option tells Logger32 to beam towards the DX stations you are working.

> **Hinson tip**: FT4 and FT2 QSOs may be over before your beam has turned. Why bother?

- **Enable turn rotator on DX Spot click**: while chasing DX stations spotted on your BandMaps, this option tells Logger32 to turn your rotator and beam towards them. Handy that.
- **Enable automatic rotator turns to LP**: when it turns your rotator automatically, Logger32 normally points your antenna for the present band along the short path to the DX, *unless* you enable this option to point it to the long path instead.

> **Hinson tip**: enable this option only while you believe or know the long path is open, and remember to disable it when the long path fades away. Keep an eye on the rotator direction indicator while operating in case you forget.

- **Show rotator setup window**: does what it says on the tin. This is another way to open the rotator configurator.

[441] I don't have a computer-controlled rotator so these options are grayed out for me. Boo hoo.

Copyright © 2026 Robert C. Furzer K4CY

- **Show rotator debug window**: use this to monitor the commands being sent to the selected rotator through the selected serial port, and the responses sent back to Logger32 by the rotator.

41.4.3 Sound card, CW Machine and data terminal windows

With the focus on the Sound card data window, CW Machine window or Data Terminal, and a callsign in the log entry pane's Call field, the antenna can be rotated by macros:

- $**Rotator**$ turns the antenna to the short path.
- $**RotatorLP**$ turns it to the long path.

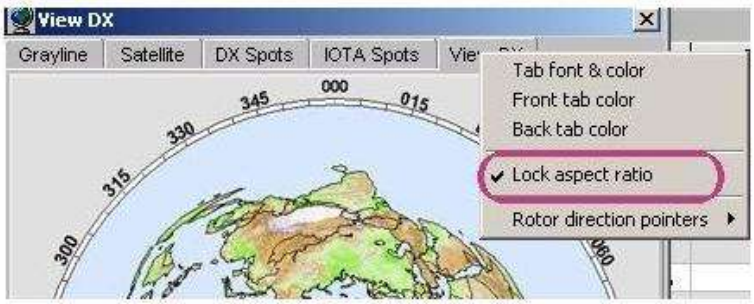
These macros can be assigned to programmable buttons and hotkeys in the usual way.

41.4.4 Tracking window

For rotator controllers that provide directional data feedback, Logger32 can display the real-time direction as overlays on the Tracking window's <**Azimuthal Equidistant Projection**> (great circle) maps, on all *except* the Satellite tab.

Furthermore, the rotator can be controlled by mouse from the Tracking window. Real-time position data will not appear unless you have a suitable rotator installed with its support software running (if applicable).

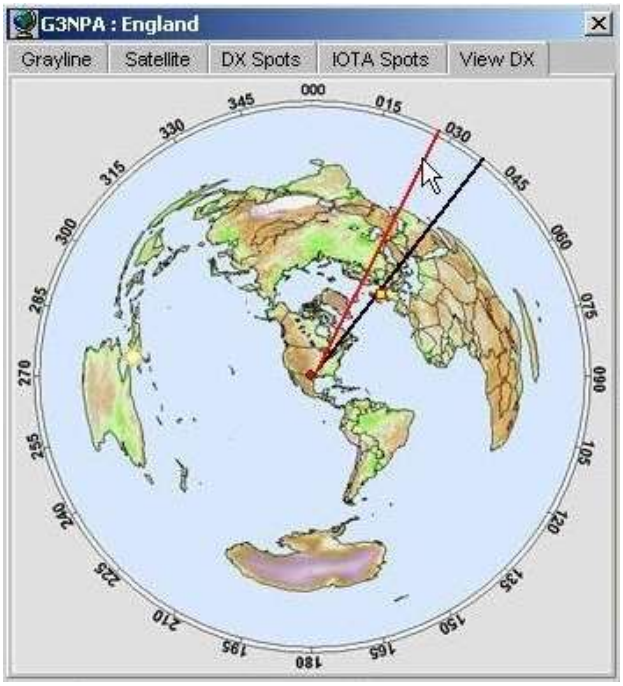
Enable <**Lock aspect ratio**> for a truly circular display ▶



By polling the rotator controller, Logger32 determines the antenna's current direction and displays a line in a color of your choice from the center of the map to the perimeter of the circle. The line turns as the antenna rotates.

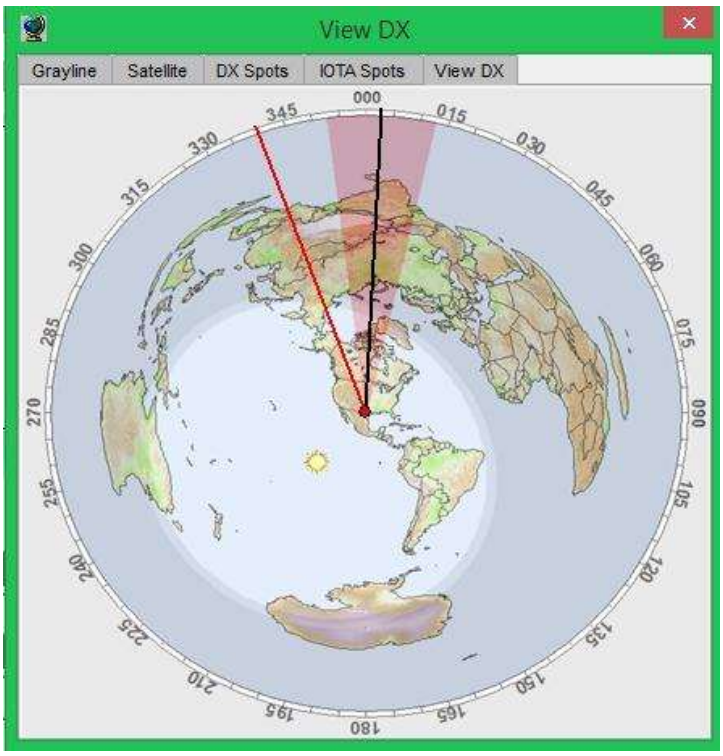
Moving the mouse cursor into the circular azimuthal map adds a red line extending from the center of the map through the mouse cursor to the compass. The line can be rotated around the compass simply by pointing within the confines of the circular map.

When you have the red line in the desired direction, click the mouse button to turn the antenna in the direction of the red line, or right-click to point the antenna 180˚ *away* from the red line ▶



Copyright © 2026 Robert C. Furzer K4CY

The most accurate positioning occurs when clicking near the periphery of the circular map, with the red line passing through the target DX location or spot … but on HF, it takes a sizeable array to generate a narrow beamwidth, and skew paths occur (*e.g.* along the grayline) so there's really no point being obsessive about this.

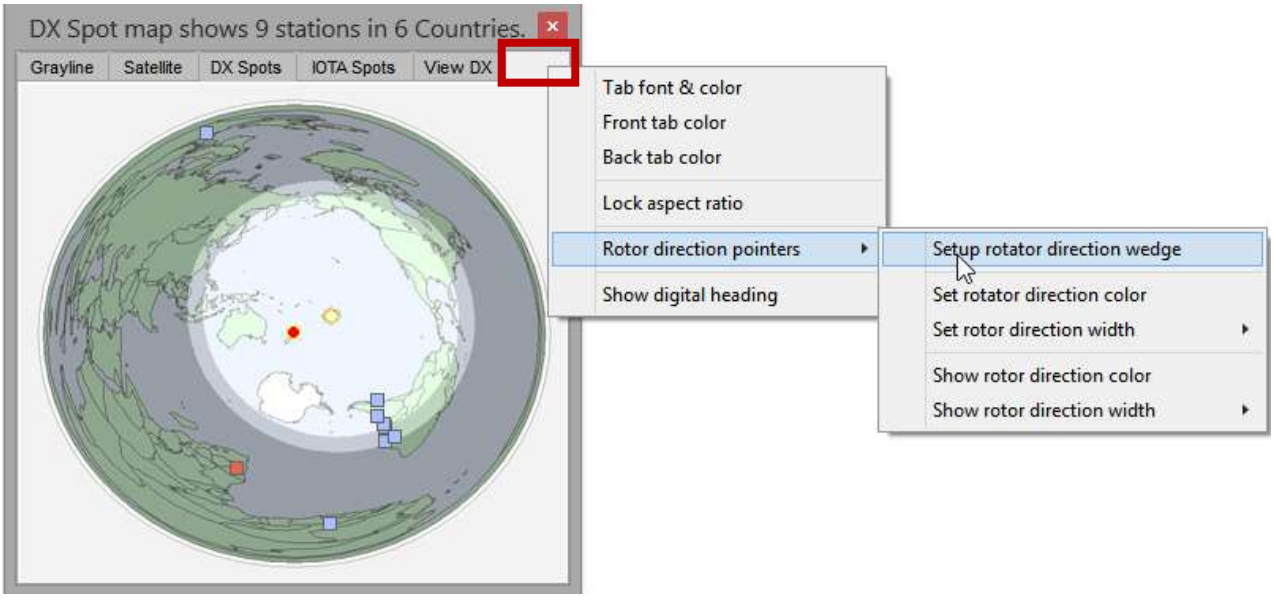


The rotator display can show a wedge representing the anticipated beam pattern.

◀ The pink wedge reflects an optimistic view of the forward beam pattern for a reversible SteppIR antenna.

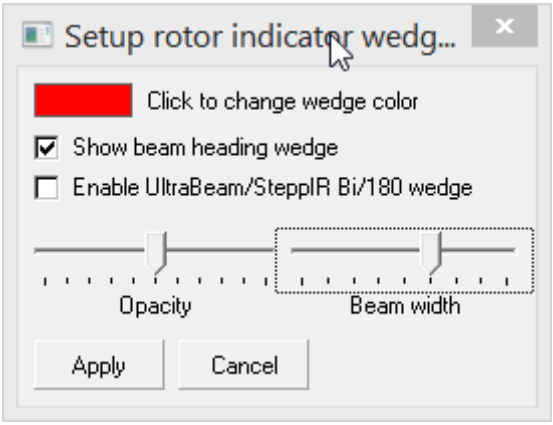
In reality, the beamwidth between (say) the -3 dB (half-power) points for a single SteppIR beam installed on a typical amateur tower (*i.e.* between 1 and 2 wavelengths above ground - definitely not in free space!) is likely to be somewhat greater than the 25° used in this example.

Configure the rotator direction pointers by right-clicking the header area just to the right of the <**View DX**> tab ▼



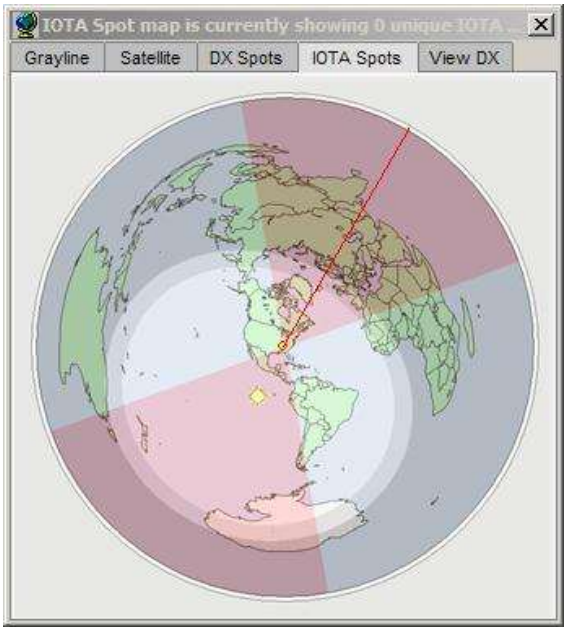
Copyright © 2026 Robert C. Furzer K4CY

You can change the appearance of the lines.



Choose the rotator indicator wedge color and opacity, and its width to indicate your beam width ▶

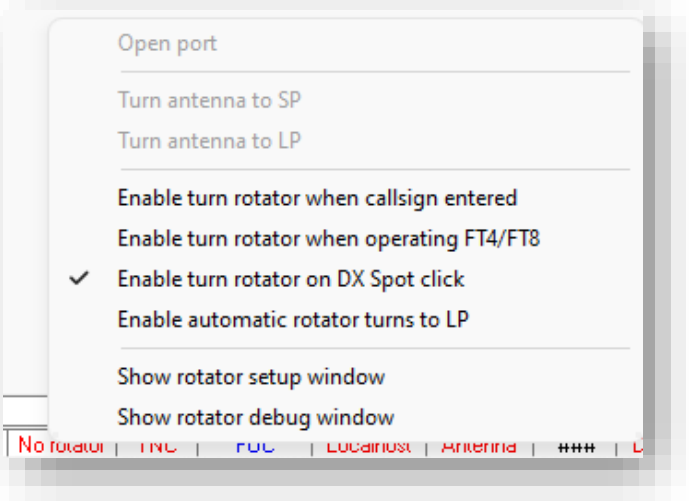
If you have a SteppIR, UltraBeam or similar with a forward/reverse/bidirectional capability - or even a plain rotatable dipole - <**Enable SteppIR Bi/180 wedge**> shows the coverage in both directions simultaneously ▼



Say, for example, you have set up a 40˚ wedge. SteppIR 180 will show a 60˚ wedge (150% of normal).

◀ In the bidirectional setting, it shows two 80˚ wedges (200% of normal).

41.4.5 Auto-pointing



Right-click the **Rotator** panel in the status bar, then tick <**Turn rotator on DX Spot click**> to have Logger32 point your beam automatically towards the stations you are calling/working ▶

Logger32 determines the heading from your QTH to the location of the station currently in the log entry pane and turns the active rotator to the short path when:

1. Focus moves off the Call field in the log entry pane (*e.g.* when you <**Tab**> after typing in a callsign).
2. Working someone on FT#.
3. Clicking a DX spot in the DX Spots pane.
4. Clicking a spot in the BandMaps.

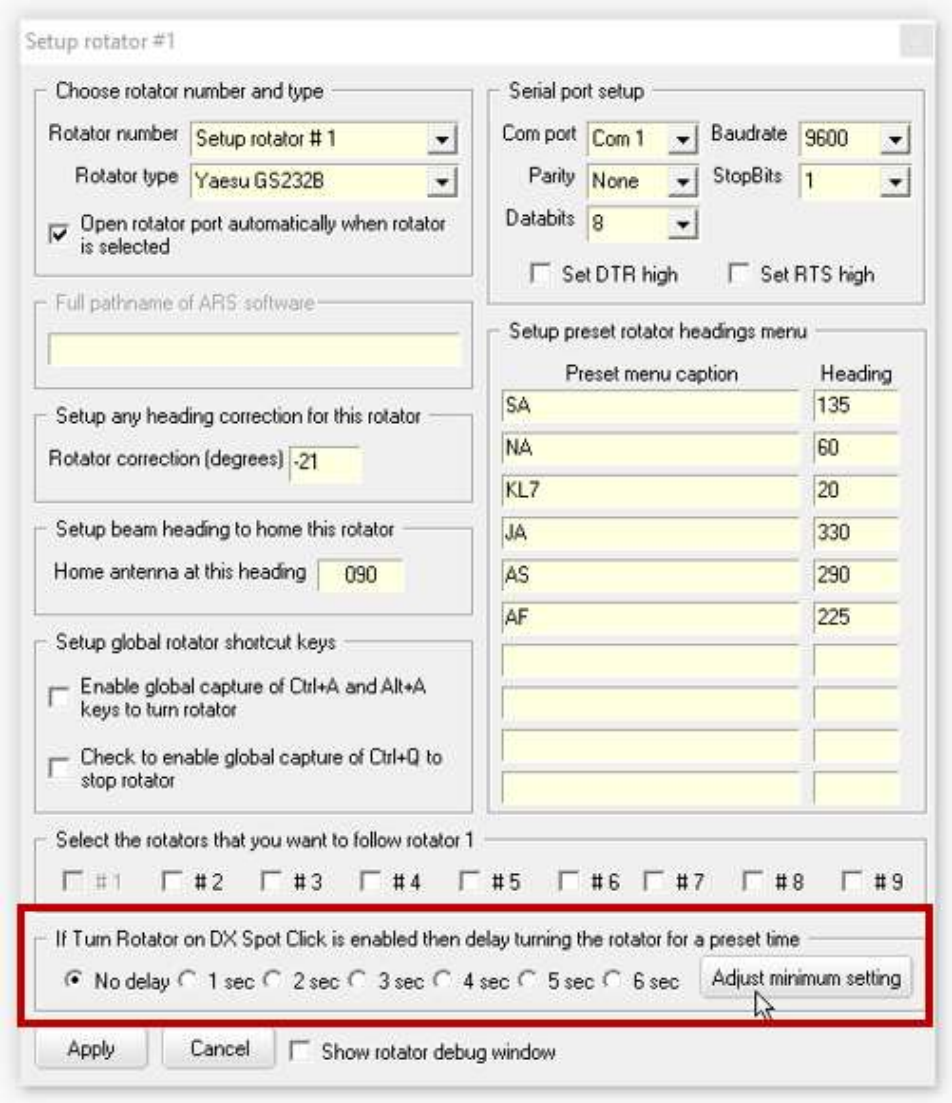
Copyright © 2026 Robert C. Furzer K4CY

5. Entering a callsign in the CW Machine's Call field.
6. Clicking a callsign in the Sound card data window decodes.

Hinson tip: while this is a cool function, frequent minor adjustments to the antenna direction create wear on the rotator bearings with little discernable difference to signal strengths, given the beamwidth of typical antennas. Unless you are using a microwave dish or a large stacked-and-bayed multi-element array, it may not be worth it. Either use <**Ctrl**+**A**> or <**Alt**+**A**> to rotate the antenna manually, or configure the rotator delay and tolerance values.

41.4.6 Setting rotator delay and tolerance

If you have selected <**Turn rotator on DX Spot click**> to enable auto-pointing, Logger32 can *pause* after you click a spot before sending the heading information to the rotator. It draws breath long enough to refine the heading based on more precise DX location information provided by the previous QSOs and/or online callsign lookup. Allow up to 6 seconds' delay for the lookup and heading refinement process to complete ▶

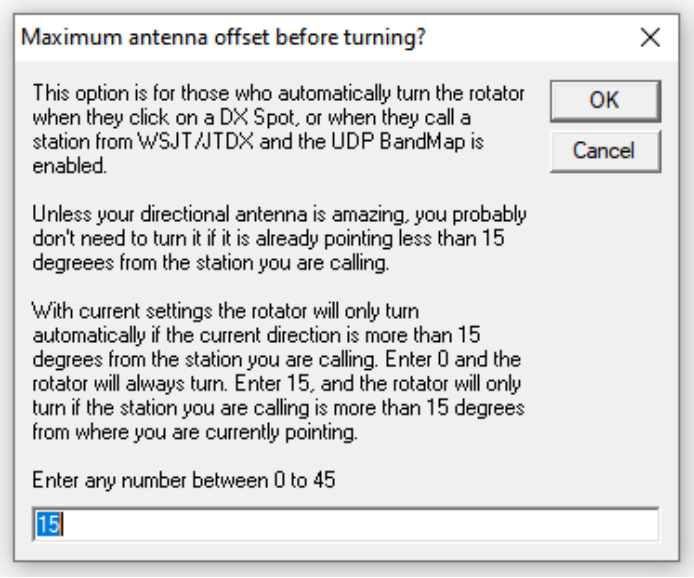


I think the rotator tolerance value affects the volume of the response after you shout "Come on! Hurry up!" at the computer while the antenna is turning oh so slowly towards some just-spotted ultra-rare DX …

Gary ZL2iFB

Copyright © 2026 Robert C. Furzer K4CY

The <**Adjust minimum setting**> button opens another form to configure a tolerance value for the offset between DX and antenna directions ▼



If your antenna is already pointing 'close enough' to the desired heading (meaning within 15˚ by default), Logger32 does not command the rotator to turn.

Although this imprecision *may* cost you a tiny amount of gain, it saves unnecessary wear and tear on the rotator.

Hinson tip: frankly, with my HF beams being so close to *terra firma* and most DX signals being subject to QSB, I am lucky to detect any consistent difference in signal strength so long as I am beaming towards the correct quadrant (*i.e.* 45˚ tolerance).

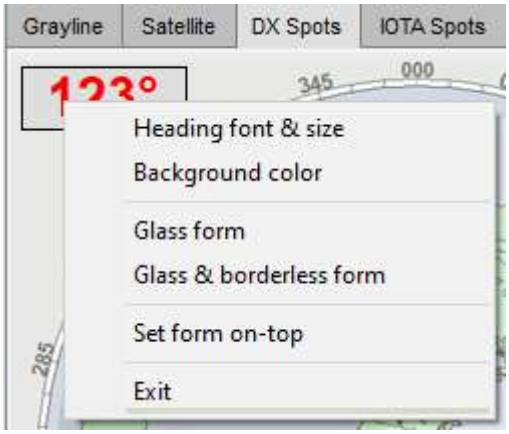
41.4.7 Show digital heading readout



◀ This digital readout tells you the rotator's azimuth setting in degrees, relative to *true* North remember.

To show it, right-click any of the map tabs and click <**Show digital heading**>.

Right-click the floating display for configuration options similar to the floating clock and floating callsign ▶



41.5 Linked rotators

Linking rotators lets us align beams for the same band on different rotators, for example to:

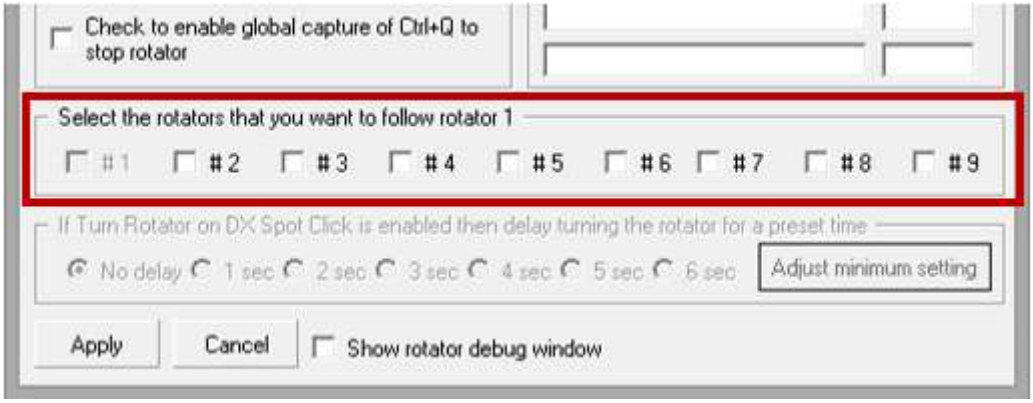
- Point phased antennas on different towers, or side and top-mounted on the same tower, towards the same DX station.
- Compare signals on different beams *e.g.* a high beam versus a low one, switching the receiver between them.

Copyright © 2026 Robert C. Furzer K4CY

- Use diversity reception with one beam and receiver feeding audio to one ear, another beam and receiver feeding another ear *e.g*. listen on the short path with one antenna feeding one ear and the long path on another by setting a 180˚ offset on the long path antenna.

41.5.1 Configure the linkage

Near the bottom of the **Setup** ⇨ **Rotator** configuration screen is a set of buttons to link other (slaved) rotators to the one we are presently configuring (the master) ▼



Simply click to tick the rotator/s that should be linked (aligned) to the present one (which is grayed out). Now, when the present antenna is rotated, the others follow suit.

Hinson tip: if you only want to link rotators for multiband antennas on particular bands, first configure the rotators separately (*e.g*. rotator #1 and rotator #2), then configure a third fake setting (rotator #3) with identical settings to rotator #1 but this time selecting rotator #2 to follow it. In your Bands & Modes table, select rotator #3 for the band/s on which the rotators should be linked, otherwise rotator #1 or #2 as appropriate.

41.5.2 Synchronize linked rotator

If you reposition individual rotators manually, or if the antennas spin in the wind, they can become misaligned. To synchronize (re-align) a linked rotator to another, click Toolbar icon #28 to open a pull-down menu ▼



Click the <**Sync Rotator …**> line to perform the sync operation.

In the example shown here, the primary antenna on rotator #1 is currently pointed west at 274˚. The other antenna on rotator #2 is presumably aimed somewhere else – maybe 30˚ or 300˚. When you click the <**Sync Rotator …**> line, the secondary antenna turns clockwise or counterclockwise to reach 274˚ as well[442].

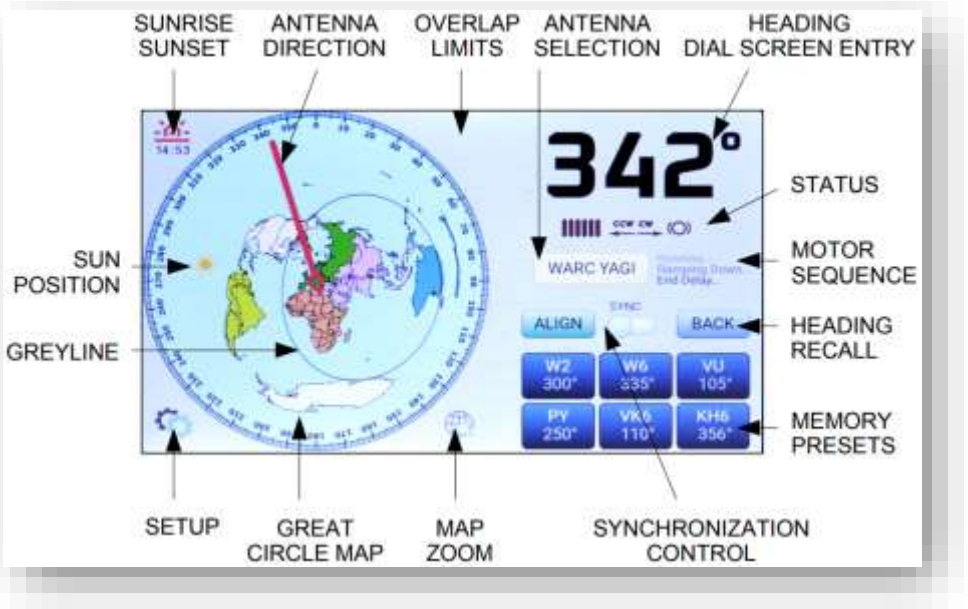
[442] Under **Setup** ⇨ **Rotator**, tick <**Open rotator port automatically when rotator is selected**> when using multiple rotators, otherwise the COM port for the slave rotator may be fast asleep and you may not notice it gently snoring.

Copyright © 2026 Robert C. Furzer K4CY

41.6 *Micro*HAM ARCO controller



The *micro*HAM ARCO Smart Antenna Rotator Controller is designed to control virtually *any* rotator with AC, DC or stepper motors, various position sensors and stops *etc*. using its two microcontrollers. Once connected through its serial/USB interface, or using TCP/IP over an Ethernet/WiFi LAN, or remotely over a WAN (using a client-server pair of ARCOs), the ARCO can be configured through its web interface or using its front panel.

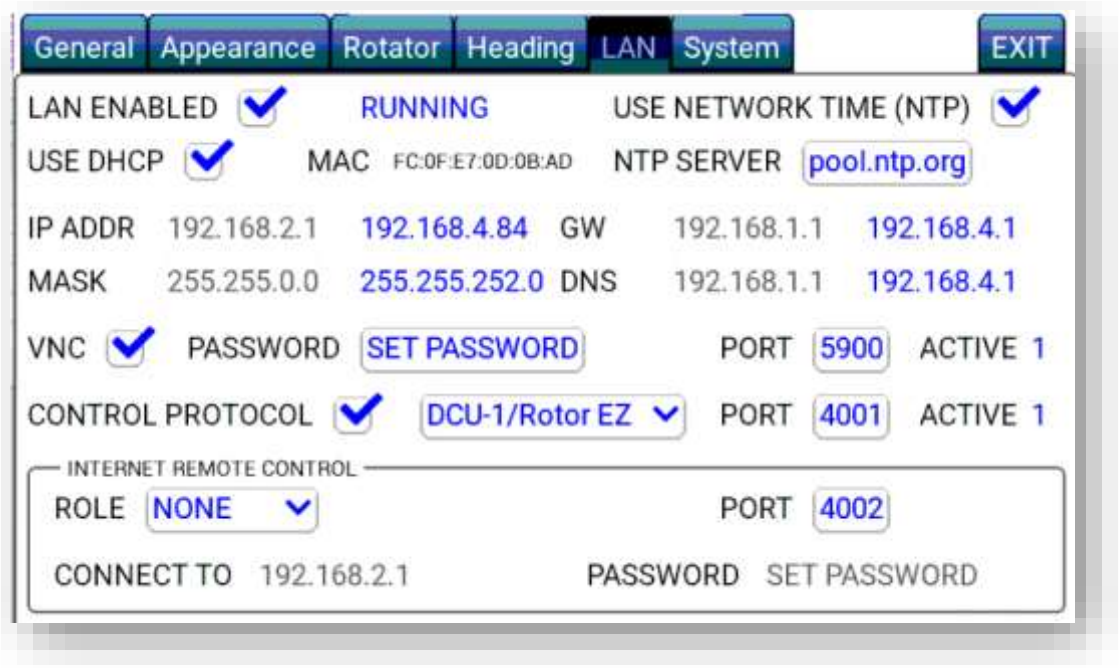


◀ The ARCO's pretty front panel 7" color touchscreen.

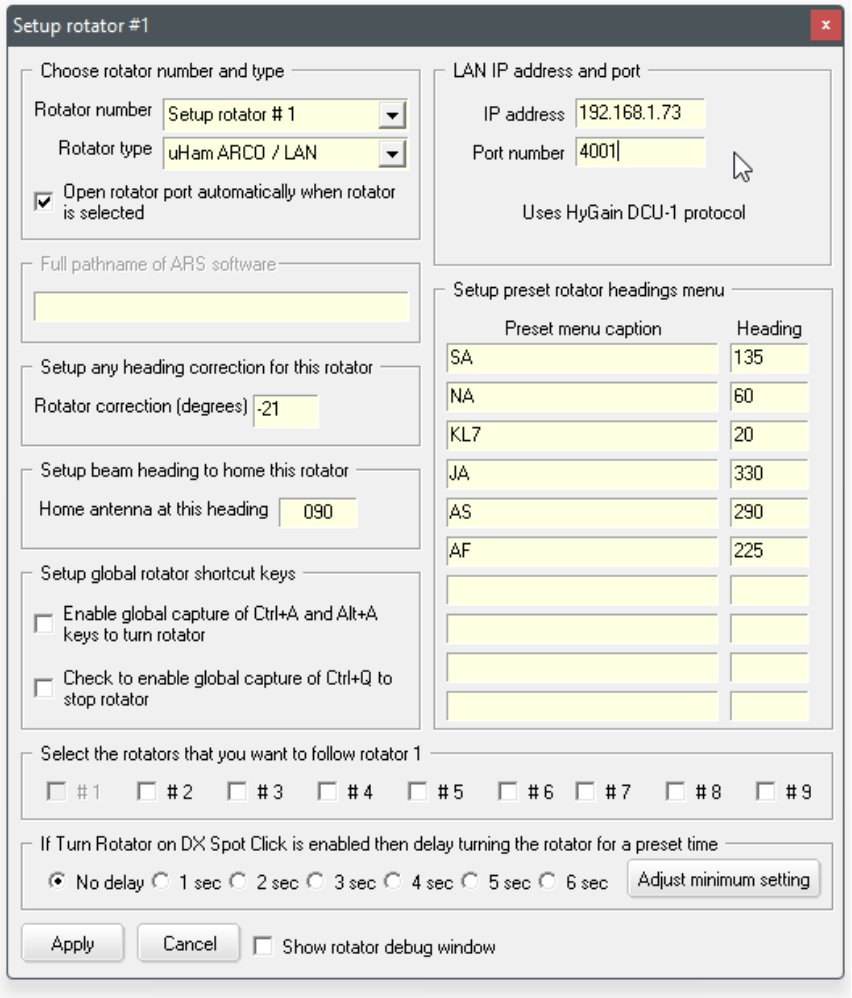
Before trying to get it going in Logger32, configure the ARCO controller for the HyGain DCU-1 protocol on the port/s you are using. To do that, consult the ARCO manual.

Copyright © 2026 Robert C. Furzer K4CY

Here's a clue about the ARCO LAN (TCP/IP) configuration ▼



To configure the ARCO as a network device in Logger32 under **Setup ⇨ Rotator**, you'll need its IP address and port ▶

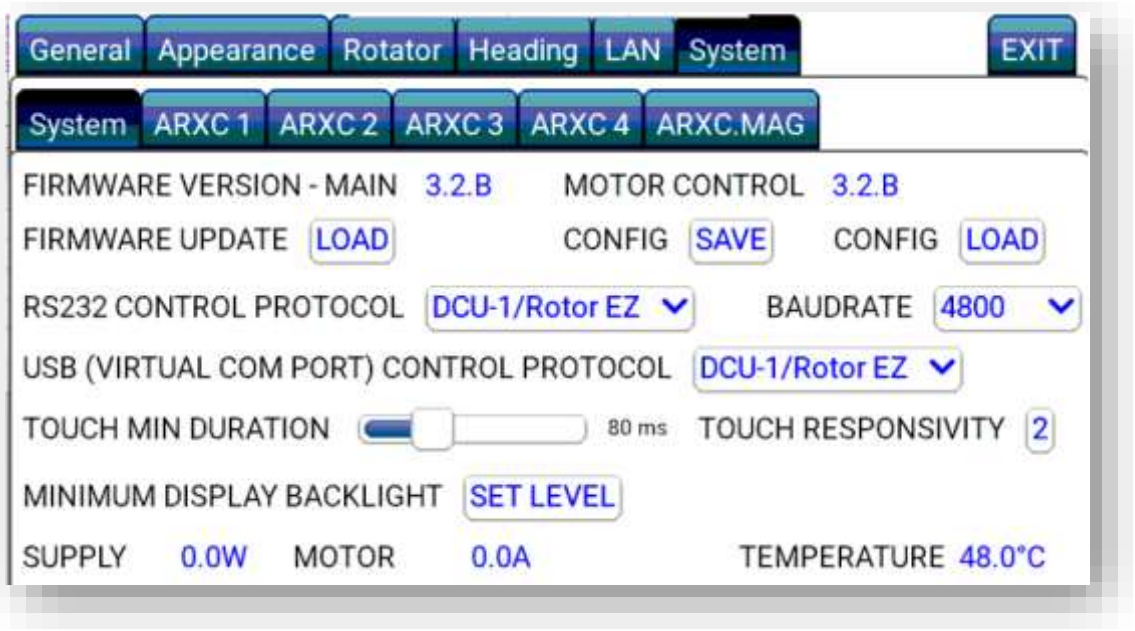


Whenever you change the ARCO LAN setup, it unticks the enabled box, so don't forget to tick and enable it again when you are done configuring.

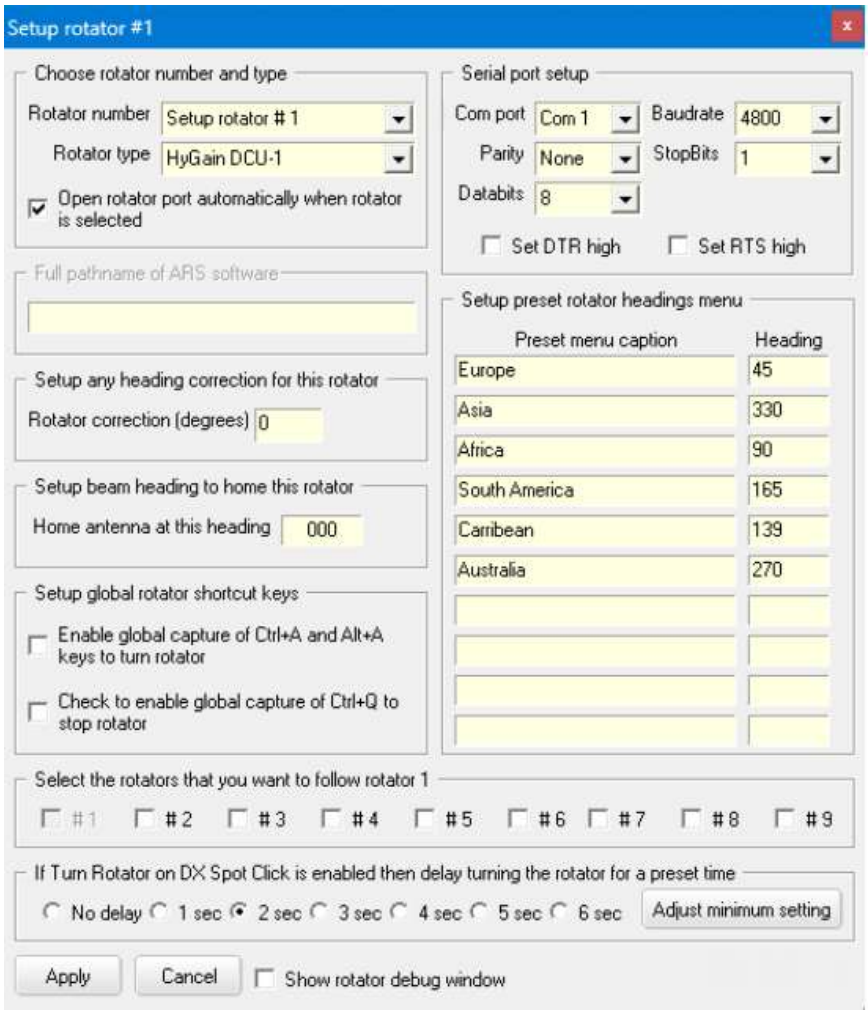
Jim W4UCK

Copyright © 2026 Robert C. Furzer K4CY

And here's the ARCO RS232 (serial) setup ▼



To configure the ARCO as a serial device in Logger32 under **Setup ⇨ Rotator**, you'll need its control protocol (*e.g.* HyGain DCU-1) plus the COM port number (from Windows device manager) and baudrate (from ARCO)▶



Copyright © 2026 Robert C. Furzer K4CY

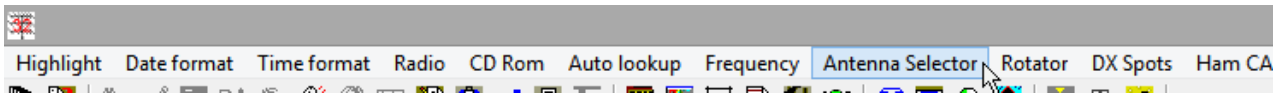
42 Antenna switching

> "With aerials, we can defy gravity, challenge our limits, and reach for the stars"
>
> *Yuri Gagarin*

Logger32 supports several methods of automatically switching to the antenna that has been defined for the band or sub-band in use in the <**Aerial**> column of your Bands & Modes table:

- **Parallel port**: Logger32 controls remote antenna relays, using pins on a parallel port to select the relays. Bit-wise and **B**inary **C**oded **D**ecimal encoding techniques are both supported.
- **Serial port**: Logger32 uses the OTRSP protocol, *micro*HAM functions or command interface to control a remote antenna switch via a serial port.
- **The antenna switch built-in to some radios**: if your radio doesn't automatically select the appropriate antenna connector when you change band, Logger32 can send the appropriate CAT commands to the radio if so defined in the Bands & Modes table.

Configure antenna switching using **Setup ⇨ Antenna Selector ▼**



You can configure any antenna (*e.g.* a multiband vertical or doublet with auto-tuner) as the default antenna to be selected whenever the radio is tuned to a frequency outside the defined bands, or if a specific antenna is not defined for the particular frequency in use.

An automatic antenna switch can also be switched manually in order to check/test the configuration and hardware.

42.1 Parallel port antenna switching

Parallel ports were originally used to connect line-printers – often noisy monochrome dot-matrix devices early in the PC age[443]. Typical computers back then had a 25-pin DB25 socket for the "LPT" port, connected to a special double-sided 36-way "Centronics" socket on the printer. With 8 digital data lines (one per bit) in parallel, data was sent to the printer a whole byte at a time. Gosh.

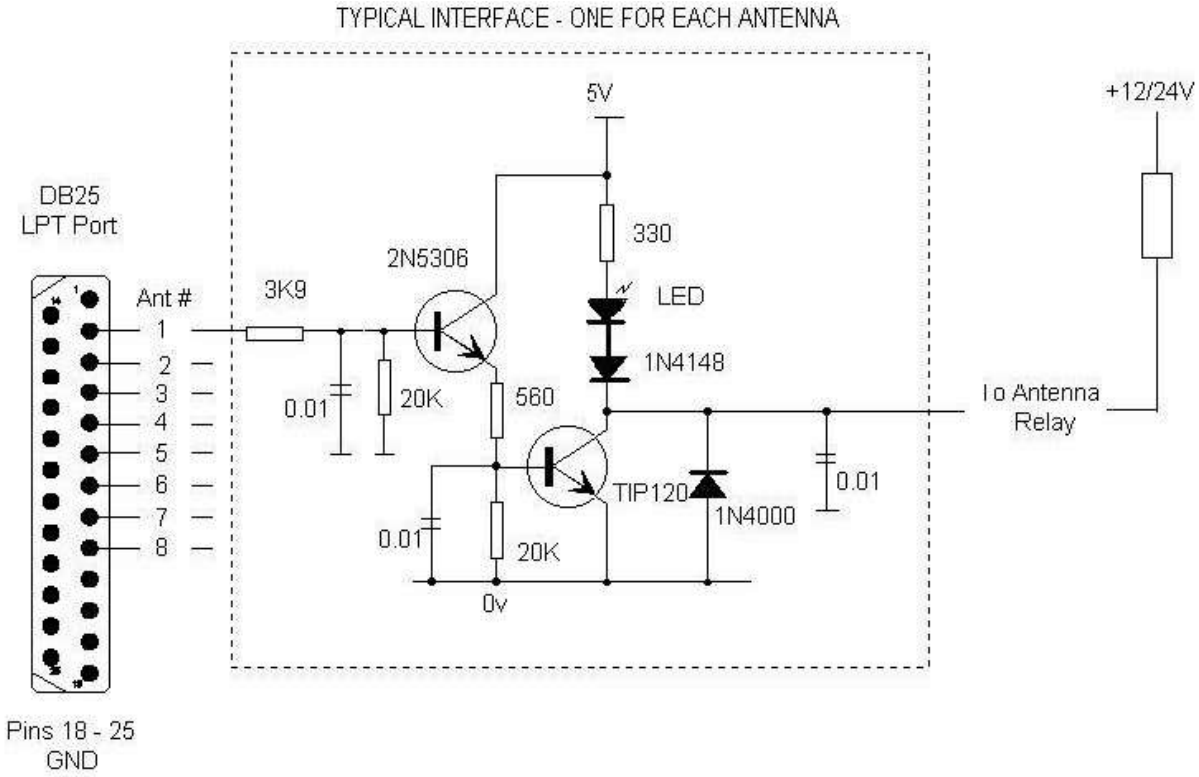
[443] Parallel ports are rarely provided on modern computers and printers but, fortunately, serial-to-parallel converter devices or cables are readily available, emulating LPT ports in software. Add-in PCI cards offering one or more parallel ports are available for laboratory equipment control and robotics.

Copyright © 2026 Robert C. Furzer K4CY

42.1.1 Parallel port bit-wise antenna control

An 8-bit byte of data sent to the parallel port will assert (set *e.g.* raise to +5V or ground) or release (leave floating) specific pins, facilitating the direct control of external devices by up to 8 relay or transistor switches connected to those 8 pins (pin-wise control).

Parallel port data lines need to be isolated and level-converted to operate typical antenna relays. The following hardware interface design isolates the parallel port data lines and supplies sufficient power to drive most antenna or equipment control relays.



The TIP120 transistor passes a little more than the relay current continuously while the relay is powered, and should be mounted on an adequate heat sink. If yours gets hot after, say, a minute or three on a given band, you probably need better heat sinking to avoid releasing the smoke.

A duplicate interface circuit is required for each antenna/equipment relay.

The LED is useful for fault-finding, just in case Logger32's representation of an "LED" for that pin proves misleading.

The circuit shown suits antenna switches such as the 5-way Ameritron RCS-8 or 8-way RCS-10, where the remote relays are powered by individual wires, with minor changes.

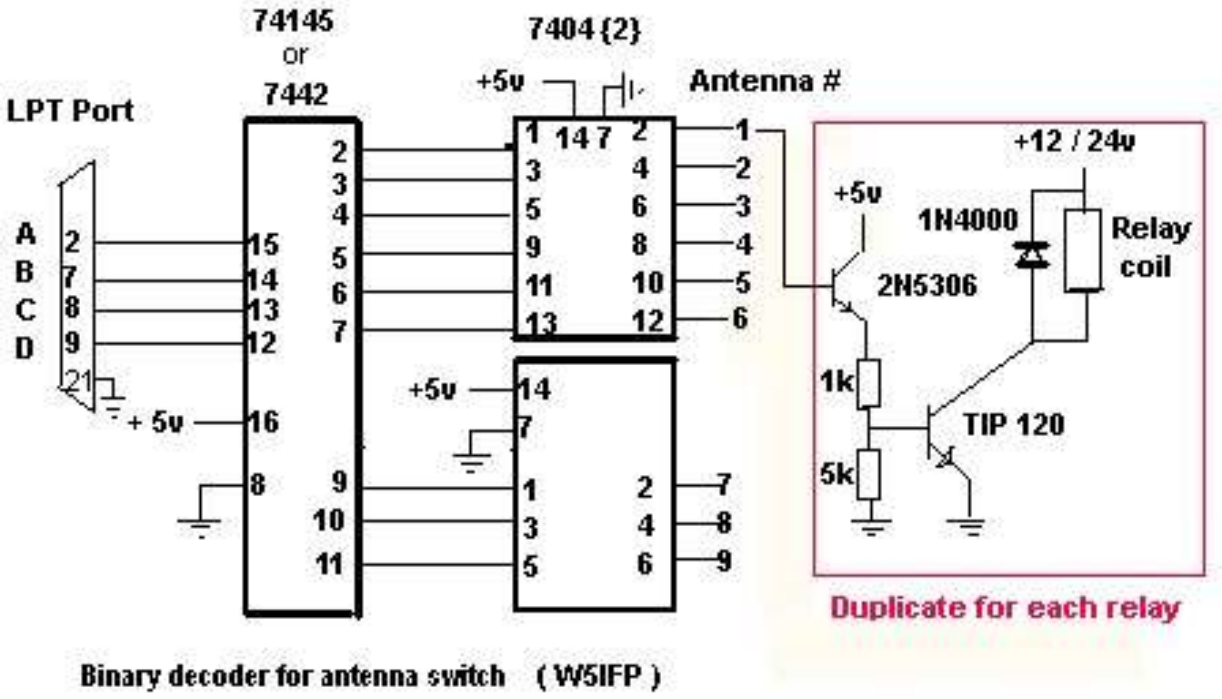
The Ameritron RCS-8 manual control box can be modified by the addition of a mini DPDT switch to disable the existing common connections on the LED and relay switch wafers. VCC power to the interface can be supplied by switching the common to the relay wafer to the interface. The LED and relay lines from the interface circuits can be connected directly to each LED switch position. This enables switch between manual and automatic modes.

The Ameritron RCS-4 manually controls 4 remote relays by feeding control voltages through the coax: it can be modified to work automatically with Logger32 by adding a remote-control cable and minor wiring modifications to both the control box and the remote relay box.

Copyright © 2026 Robert C. Furzer K4CY

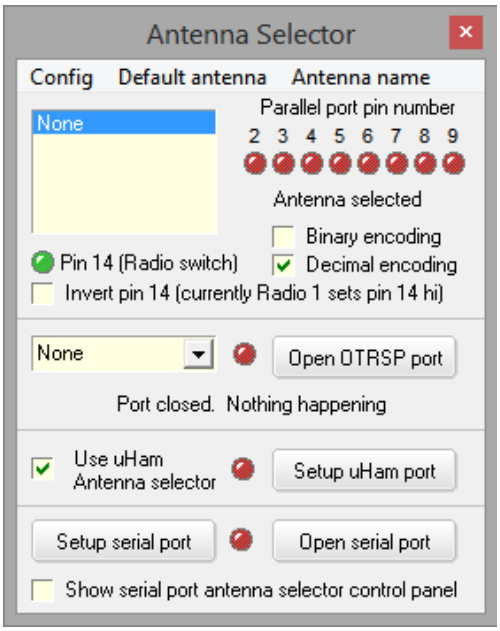
42.1.2 Parallel port BCD antenna selection

The following circuit can select from up to *nine* different antennas using a BCD decoder and hex inverter chips, followed by two-transistor drivers for each antenna relay:



The maximum possible decimal address with a 4-bit BCD value is 15 (1+2+4+8), potentially allowing the selection of any one of 15 antennas[444] using a 4-line BCD to 16-line decimal decoder circuit.

42.1.3 Parallel port antenna switch configuration



First under <**Config**>, set the parallel port's hexadecimal address - typically **&H378** for LPT1 or **&H278** for LPT2[445].

Depending on the type of interface circuit you are using, choose either:

- **Binary encoding:** Logger32 selects antennas 1 through 15 decimal by asserting pins 2, 7, 8 and/or 9 for the corresponding binary number (BCD).
- **Decimal encoding**: Logger32 asserts one of the pins from pin 2 through 9 to enable antenna 1 through 8 directly (pin-wise).

The "LEDs" represent logic probes: they go green when the numbered pins are at logic 1 (+5V), red when at logic 0 (grounded), and clear when unused (as with BCD encoding that only uses 4 of the 8 pins).

[444] Plus all-zeroes or remote-control unit powered off, meaning "Disconnect all antennas".

[445] You may need to open Windows Device Manager from Settings or the Control Panel to determine the address assigned to your parallel port. Parallel port PCI cards and USB devices may have other addresses.

Copyright © 2026 Robert C. Furzer K4CY

A further "LED" represents a typical logic probe, showing the state of **pin 14 (Radio switch)** when two radios are configured in Logger32:

- Green shows the pin is held high (+5V), normally meaning that Radio #1 is selected.
- Red shows the pin is grounded, normally meaning that Radio #2 is selected.
- **Invert pin 14:** inverts the logic state. With invert selected, green shows the pin is held high to select Radio #2, whereas red shows it grounded for Radio #1.

With suitable interfacing, pin 14 can switch cables, microphones *etc*. between two radios. It can also be combined with the antenna lines to switch antennas and linears for whichever radio is active. Read more about pin 14 in the antenna switch and SO2R sections.

42.2 Serial port antenna switching

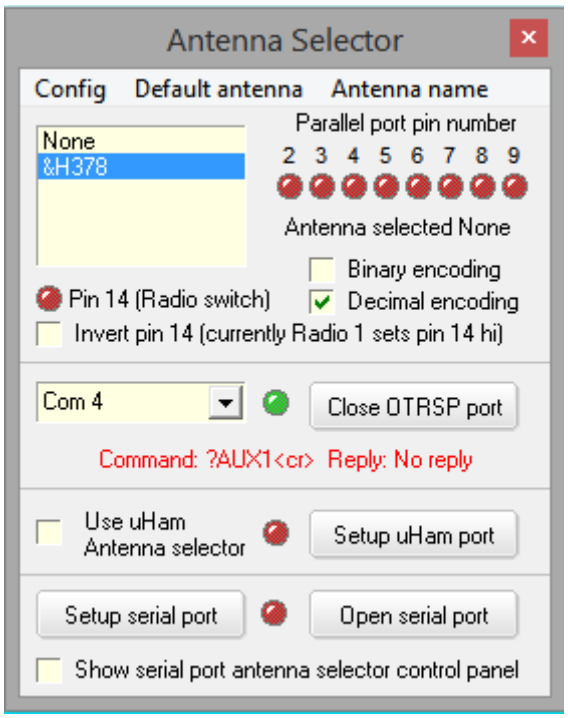
Antennas can be selected by sending appropriate commands through a serial port to an intelligent antenna switch controller. The type of controller device determines the protocol *i.e*. the serial port communications settings and the command structure.

Logger32 supports two predefined protocols, plus a user-configurable option.

42.2.1 OTRSP

The **O**pen **T**wo **R**adio **S**witching **P**rotocol is a standard defining the commands to control an SO2R setup via 9600 baud serial comms.

This shows the configuration to select antennas using the OTRSP protocol on a serial port, COM4 in this example ▶



Having selected the relevant COM port from the drop-down list and clicked <**Open OTRSP port**>, the "LED" in this section turns green showing that OTRSP is in use.

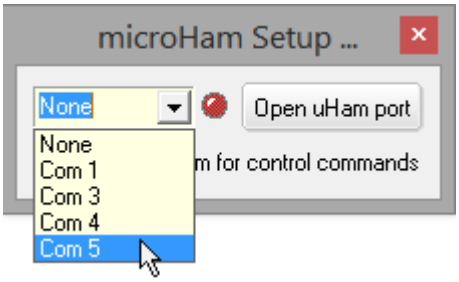
The red text shows OTRSP queries (polls) and commands sent to, and any replies received from, the SO2R device[446].

See the OTRSP and SO2R chapters for more, plus the OTRSP protocol for information about the AUX command used for auxiliary control purposes, such as antenna switching.

[446] My SO2R controller kit remains unboxed and unbuilt at this point, hence there is no red reply shown in this example screenshot. It's a one day project, as in "*One* day I'll get around to building and implementing it, maybe when this User Manual is finally 'done'!"

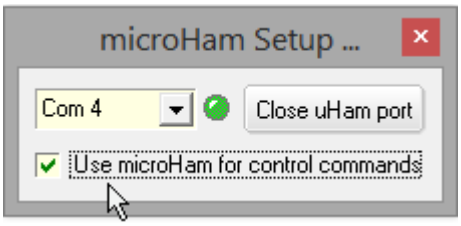
Copyright © 2026 Robert C. Furzer K4CY

42.2.2 microHAM



◀ From **Setup ⇨ Antenna Switch**, click <**Setup uHam port**> to select and then open whichever COM port is connected to your *micro*HAM box.

<**Use microHAM for control commands**> is a switch that leaves the *micro*HAM port open (for other purposes) if you close the *micro*HAM antenna selector port ▶



See the *micro*HAM chapter for more.

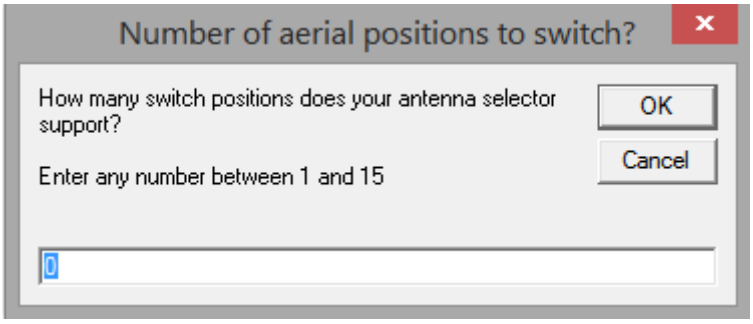
42.2.3 Commanding other intelligent controllers



The bottom section of the Antenna Selector form is used to configure a serial port to pass antenna selection commands [447] to some other type of relay controller, one that supports neither the OTRSP nor *micro*HAM protocols. Such controllers are commonly used for robotics applications.

◀ <**Setup serial port**> opens a typical serial port configurator where you choose the COM port, set the speed and other comms parameters.

At the bottom right, click <**Change #**> to tell Logger32 how many antennas are available to select, then click <**OK**> ▶



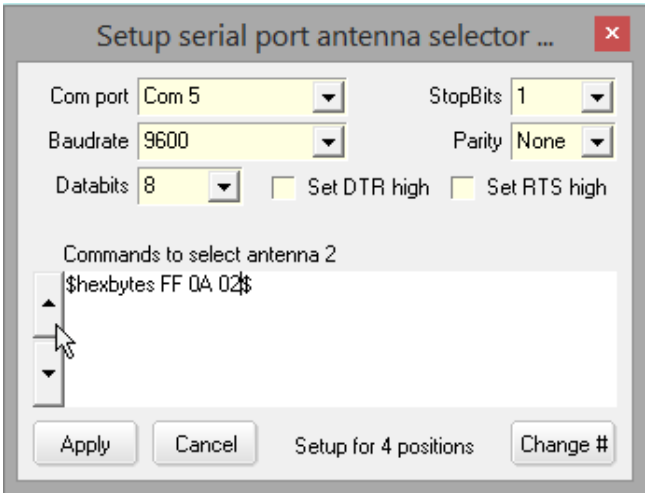
Enter the corresponding antenna-selection commands, starting with the command/s to select no antenna.

[447] Multi-line commands are sent in sequence, for instance a command to open all the aerial relays, following by a command to close the selected one.

Copyright © 2026 Robert C. Furzer K4CY

Click the up or down arrow buttons here to change to the setup panels for each of the antennas you have said are available ▶

Activate the new setup by clicking <**Apply**> ▶



You may need to set the DTR or RTS modem control lines high *e.g.* to feed power to a USB-powered relay controller device.

With the configuration all done, the antenna switch powered-up and connected to the controller which is plugged into the designated PC serial port, Logger32 uses the aerial column in your Bands & Modes table to determine which antenna is to be used for the current VFO frequency, sending the appropriate command to the controller.

42.2.4 An example: the KMTronic USB HOST relay controller

The KMTronic controller contains 8 *local* relays capable of controlling up to 8 *remote* relays – potentially antenna relays sitting in a weatherproof box in your antenna field at the far end of your feeder – either a commercial remote antenna relay unit such as the Ameritron RCS-8 or RCS-10, or a homebrew relay box using surplus Russian vacuum relays, perhaps.

Relay controllers such as the KMTronic are commanded via USB *i.e.* a COM port.

The KMTronic can be configured for decimal (pin-wise) or binary (BCD) control. Both normally-open and normally-closed outputs are available at each position.

These are the commands with the unit in decimal mode:

$**HexBytes FF 0A 00**$ $**HexBytes FF 01 01**$ - Opens all relays[448] then closes #1

$**HexBytes FF 0A 00**$ $**HexBytes FF 02 01**$ - Opens all relays then closes #2 …

… and so on up to …

$**HexBytes FF 0A 00**$ $**HexBytes FF 08 01**$ - Opens all relays then closes #8.

In BCD mode, the commands are of the form:

$**HexBytes FF 0A 00**$ - Clears all binary bits, opening all relays, then

$**HexBytes FF 01 01**$ - Sets binary bit #1 (LSB)

$**HexBytes FF 02 01**$ - Sets binary bit #2 …

… and so on up to …

$**HexBytes FF 08 01**$ - Sets binary bit #8 (MSB)

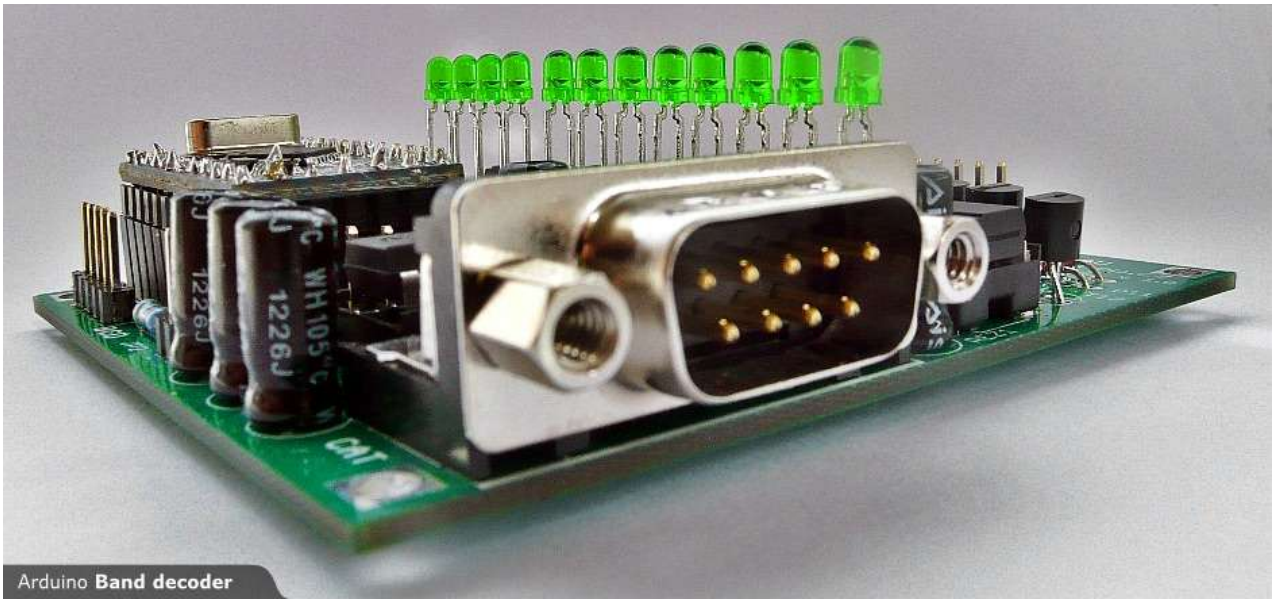
[448] The 'Open all relays' command in each case first resets *all* the relays to a known starting state, reducing the possibility of multiple relays being closed simultaneously.

Copyright © 2026 Robert C. Furzer K4CY

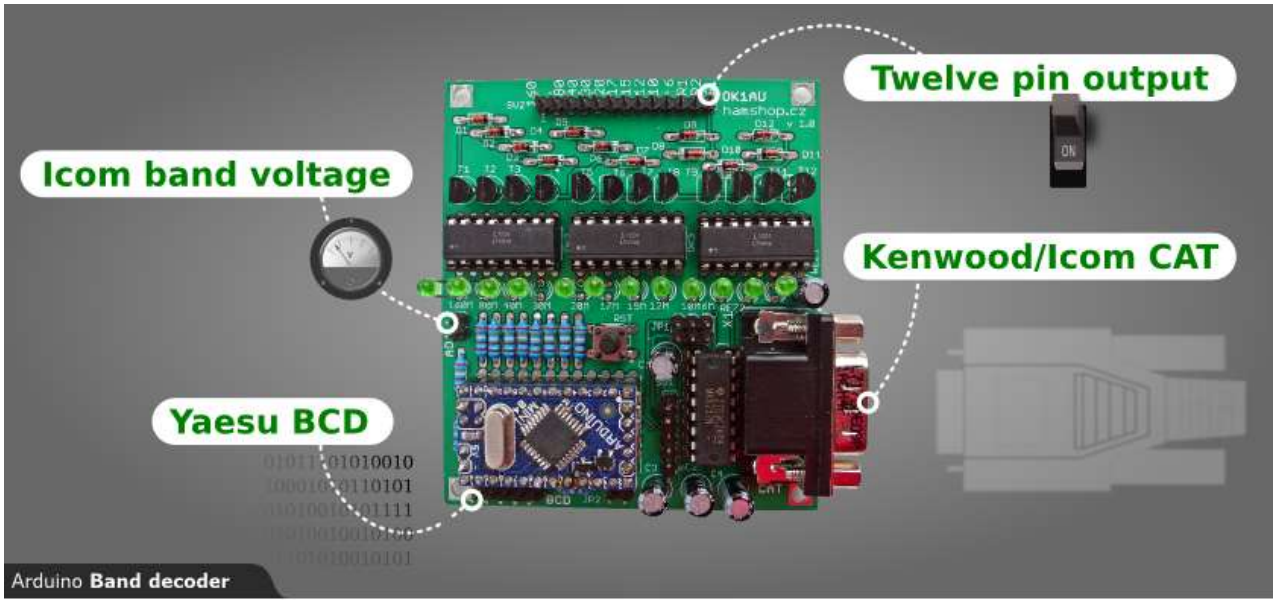
So to close relay #3 requires the BCD code 00000011 *i.e.*:

$**HexBytes FF 0A 00**$ $**HexBytes FF 01 01**$ $**HexBytes FF 02 01**$

42.2.5 Another example: an Arduino based band decoder



OK1AU sells a €31 kit for a cool Arduino-based band decoder that accepts multiple command formats/methods used by various radio manufacturers, to control twelve switches (opto-isolated open collector transistor outputs, each with LED indicators).

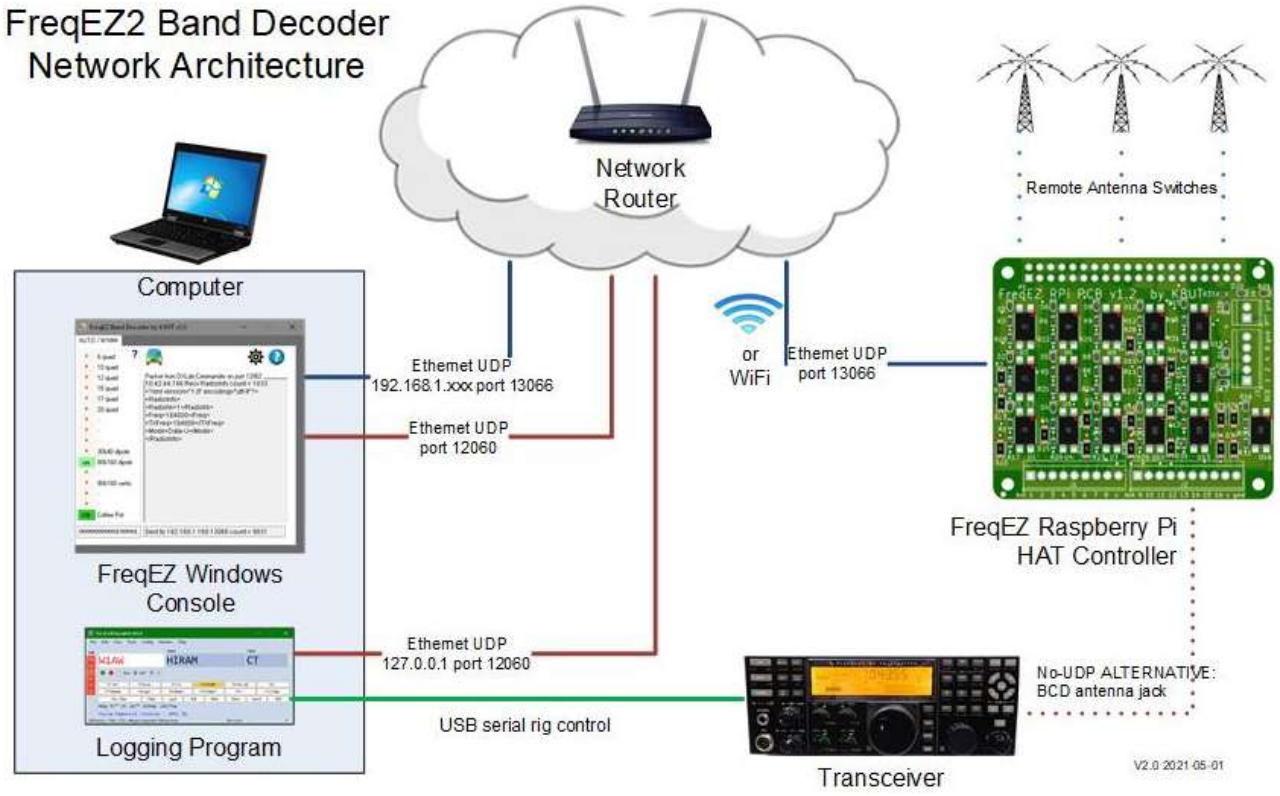


The inputs accept band data directly from the big-brand radios. Logger32 can generate the band data and send it via a COM port to the Kenwood/ICOM CAT port on the board – a conventional 9-way RS232 male connector. Typical USB-RS232 converters should plug straight in.

Hinson tip: If those output pins are not suitable, search your favorite electronics suppliers for microcontroller relay boards to interface the output pins to whatever needs to be switched. There's no shortage of relay boards capable of selecting between 1, 2, 4, 8 or 16 relays, often opto-isolated as well offering still better RF immunity. Make sure the microcontroller and relay power supply is also RF immune though, or expect the unexpected, especially with QRO or high SWR.

Copyright © 2026 Robert C. Furzer K4CY

42.2.6 Yet another example: a Raspberry Pi-based antenna switch



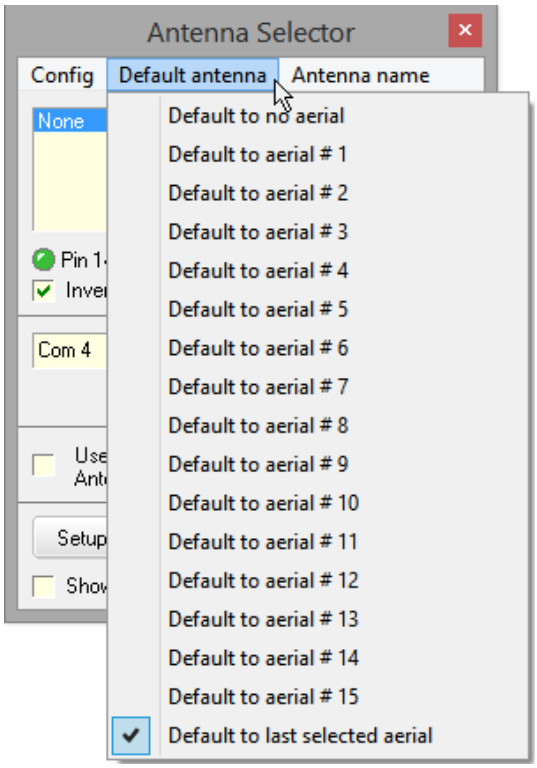
FreqEZ2 by Larry K8UT is an RPi-based band decoder capable of selecting from 16 antennas according to UDP messages broadcast on the station LAN (wired or wireless) by Logger32 or other loggers such as N1MM+.

The microcontroller software is open source, and the hardware is available as a pre-populated circuit board the same size as the RPi with solid-state switches (see the "HAT Controller" on the diagram above), or can be home-constructed, perhaps using encapsulated relay modules like this ▶



Copyright © 2026 Robert C. Furzer K4CY

42.3 Default antenna

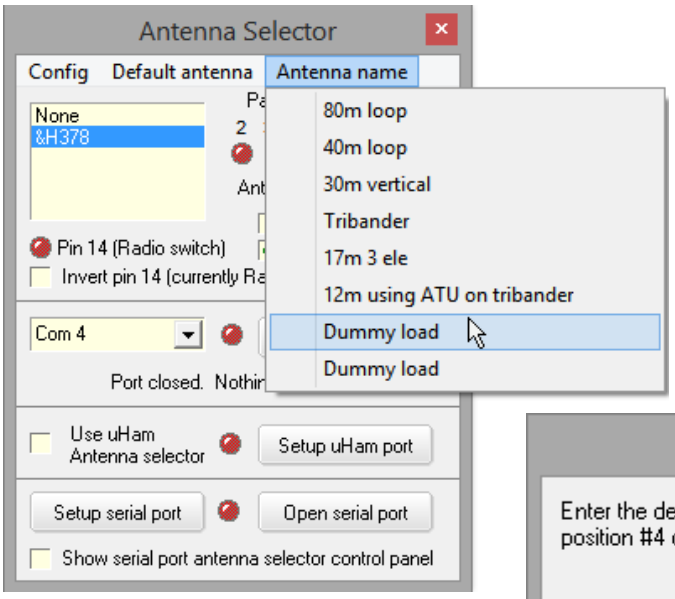


On the Antenna Selector form, click <**Default antenna**> to list the antennas from which to select the default ▶

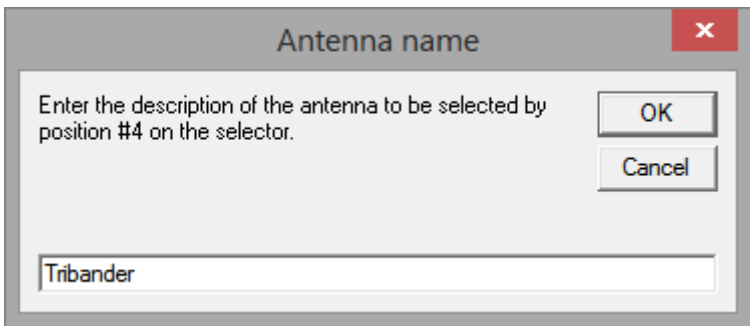
Simply click to tick the desired antenna (*e.g.* a multiband vertical) to be used on any frequency that is undefined, or does not have a specific antenna defined, in your bandplan.

<**Default to last selected aerial**> means that Logger32 will continue using the same antenna when the VFO goes out of a frequency range defined in the bandplan as having a specific antenna. For example, if you were on 20m using a beam, then QSYd just beyond the band edge checking for a QRM source or MARS station, Logger32 would keep the same beam active unless you retuned to a frequency within a defined segment of your bandplan that specified a different antenna. The same 20m beam would remain selected if you QSYd directly to one of the broadcast bands such as 41m, with no defined antenna.

42.4 Antenna names



◀ <**Antenna name**> lists all the antenna positions. Click any one to rename it, ending with <**OK**> ▼



After the Antenna Selector setup has been completed, add/check your antenna numbers in the aerial column of the appropriate bands/segments in your Bands & Modes table.

Copyright © 2026 Robert C. Furzer K4CY

42.5 CAT-controlled radio antenna switching

Some radios, amplifiers and ATUs have built-in antenna switches with several rear-panel antenna connectors. For example, the "ANT1" socket may be connected to an HF triband beam for 20/15/10m, with "ANT2" for a multiband vertical used on the other bands, and "RCV" for a receive-only antenna such as a directional loop or beverage. Typically, such systems *automatically* select the antenna connectors configured for the band in use. Changing bands is sufficient to select the relevant antenna connectors: the radio handles it all.

Some radios accept CAT commands to switch between the antenna connectors. On a K3, for example, the CAT commands are:

- **AN1;** to use the **ANT1** SO239 socket on the rear panel.
- **AN2;** to use the **ANT2** SO239 socket on the rear panel.

You could include these CAT commands in macros linked to buttons on the **R**adio **C**ontrol **P**anel to switch manually *e.g.* to compare signal reports on two antennas.

A few strange radios select the antenna connector automatically if you change bands using the radio's front-panel band switch but *not* if you change bands by CAT commands: for these radios, a workaround can be configured in the Bands & Modes table to accompany band-change CAT commands with the corresponding antenna-change CAT commands where applicable.

42.6 Testing the antenna switch

To manually switch the antennas, either tune the radio to a frequency that does not have an antenna designated in the bandplan, or just turn the radio off so no frequency is displayed.

Now open **Setup** ⇨ **Antenna selector** ⇨ **Default antenna**. Select an antenna so Logger32 commands the antenna switch to select it. On the right side of the Antenna Selector form a series of "LEDs" show which parallel port pin number and associated antenna are selected, correlating antennas with the parallel port hardware pins. A "LED" turns green for the active antenna.

Hinson tip: open this form during operation in order to monitor the antenna switching activity and as an aid to troubleshooting the antenna control interface.

The antenna status can also be monitored by mousing-over the status bar's <**Antenna**> panel to display a tooltip ▼



Re-set your desired default antenna after you have verified that the switch is working properly.

42.7 Antenna switching FAQ

Q. Why is the wrong antenna selected on one band? Other bands are OK.

A. *Carefully* check your Bands & Modes table for the problem band. Does it have an entry for the frequency *and* mode you are using, *plus* the radio if you have more than one? Logger32 needs all that info, otherwise it defaults to … well … the default antenna. The clue is in the name.

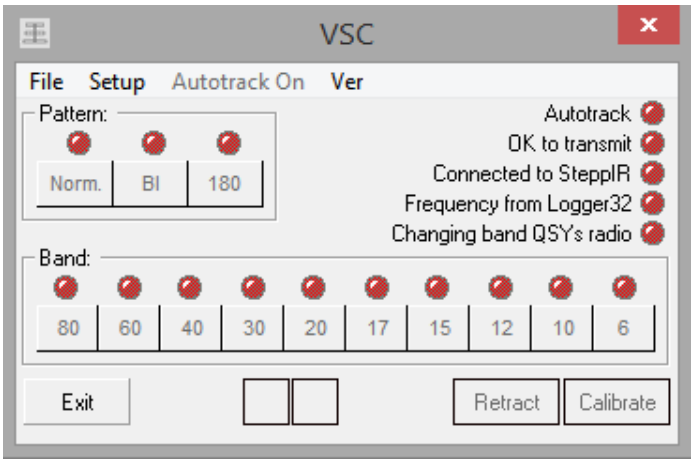
Copyright © 2026 Robert C. Furzer K4CY

43 Virtual SteppIR Controller (VSC)

> "You can't control the wind,
> but you can adjust your sails"
>
> *Thomas S. Monson*



Logger32 interfaces to the SteppIR hardware controller (either the Original transceiver controller or the newer SDA 100 ▲ fitted with the optional SteppIR transceiver interface board), controlling and displaying the SteppIR status on the PC screen like this ▶



Once correctly configured, Logger32 receives frequency data from the transceiver, sending commands to and receiving status from the SteppIR controller.

The VSC acts as a middle-man, with benefits. With it, you can:

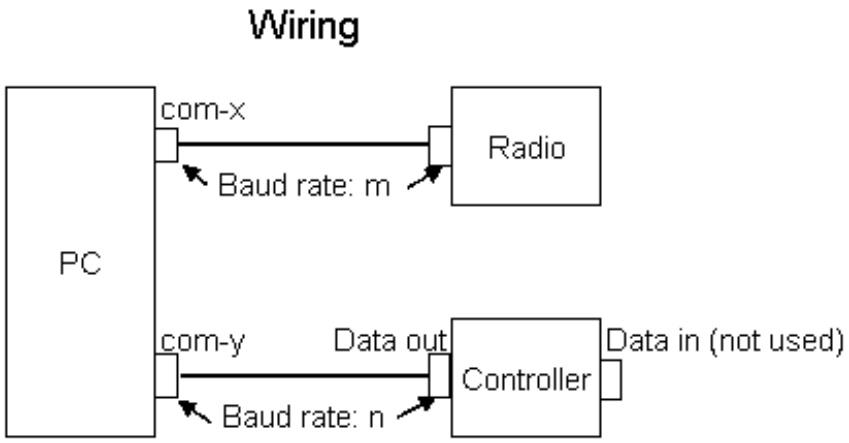
- Change the frequency/band setting of the antenna, automatically tracking the transceiver's VFO.
- Change the primary direction of SteppIR beams (forward, reverse or bidirectional) and change the radiation pattern of SteppIR verticals.
- Retract the elements.
- Calibrate the antenna.
- Move directly to a given band's stored antenna settings by clicking the band button.

The status information and functionality available reflects the type of antenna being controlled.

Copyright © 2026 Robert C. Furzer K4CY

43.1 VSC setup

43.1.1 Hardware



The serial interface to the SteppIR controllers (both types) requires a null-modem cable (cross-connecting pins 2 & 3) between the *Data out* port of the Transceiver Interface and an available serial port on your computer. A straight-through pin-to-pin serial cable will not work.

The PC serial port must be dedicated for VSC use, not shared with another device.

The *Data in* port of the Transceiver interface is not used by the VSC.

The serial interface to the radio is the existing CAT interface to Radio 1 or Radio 2 used by Logger32. An additional serial interface is not required. If there is no CAT connection to your radio, the VSC can still be used although frequency data from the radio will not be sent to the controller. The buttons on the VSC control the antenna based on the frequency and band setup of the VSC.

43.1.2 SteppIR transceiver interface

Setup the SteppIR radio type and baudrate using the transceiver interface setup menu.

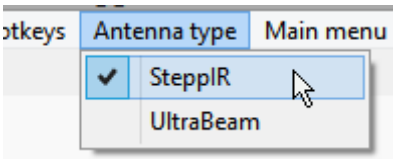
The baudrate must be between 4800 and 19.2k: start at 4800. When you change the baud rate, you must hard-reset the SteppIR interface:

- Turn the SteppIR interface off;
- Remove the DC power cable[449];
- After a few seconds, plug it back in and turn the interface back on.

Set the mode to “General Freq” for the Original controller[450] or “Autotrack” for the SDA 100 controller.

43.1.3 Virtual SteppIR Controller

Logger32 supports SteppIR and Ultrabeam antennas but only one type at a time, so tell Logger32 which type you are using under **Setup ⇨ Antenna type ▶**

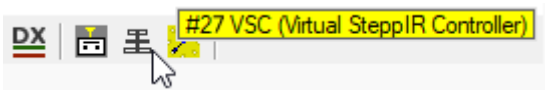


[449] The interface is not completely powered down until you physically disconnect the power cable or turn off the source power supply.

[450] On the Original controller, although the front panel reads “General Freq”, the LCD display shows “Ham Mode”.

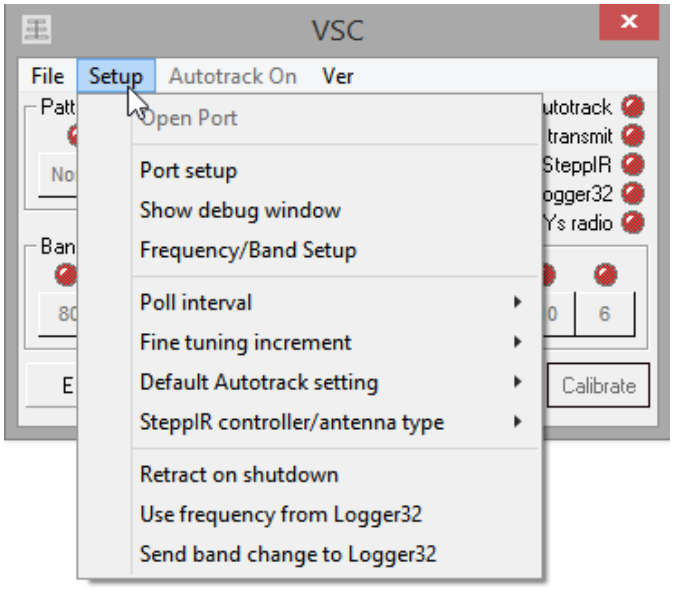
Copyright © 2026 Robert C. Furzer K4CY

Now, open the **V**irtual Steppir **C**ontroller function by clicking the toolbar tribander icon #27 ▶

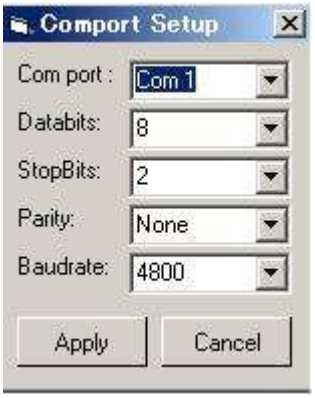


43.1.4 VSC setup

In the VSC window, <**Setup**> lets you ▶



- **Open or close the port**: like a toggle switch, it swaps state.
- **Port setup**: configure the VSC serial (COM) port number, speed and other parameters to match the controller ◀



- **Show debug window**: to watch the serial data flowing to and from the SteppIR controller.

  In this example, the first two lines display a command from the VSC to the controller (Hex data) as well as a plain-language decode of the frequency data (bytes 4-7) ▶



  The next two sets of lines show polls from the VSC to the controller, followed by the replies from the controller in hex plus their meanings.

  Click the buttons to:

  - <**Exit**> the debug function.
  - <**Freeze**> and then <**unFreeze**> (thaw) the window updates.
  - <**Clear**> the window.
  - <**Dump to file**> to save the captured data to disk for further debugging or to report issues and seek assistance.

For details of the commands and replies, study the SteppIR Transceiver Interface Protocol document from www.steppir.com/Manuals.

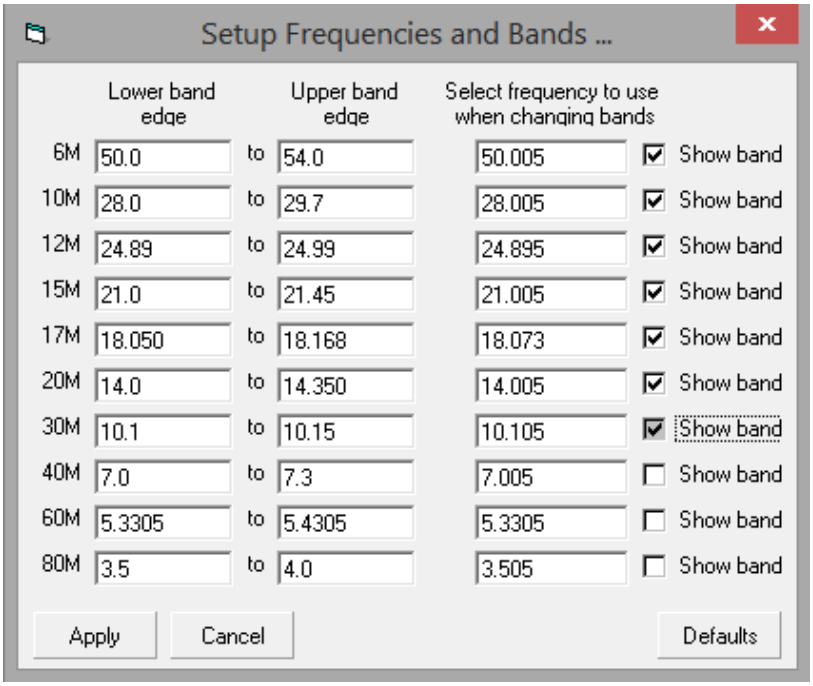
Copyright © 2026 Robert C. Furzer K4CY

- **Frequency/Band setup**: use this to configure the band edges for the VSC. Tuning to frequencies within the designated boundaries turns the respective band "LED" green.

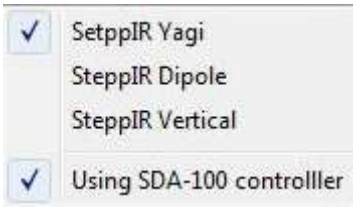
  <**Select frequency to use when changing bands**> sets a starting frequency when you QSY to the band using the bands button on the VSC form.

  Ticks in the <**Show**> boxes display the respective band buttons on the VSC form.

  If you make a mess of things, click <**Defaults**> to discard any customizations and revert to the original default entries.



- **Poll interval:** lets you change how often the VSC communicates with the transceiver interface (*e.g.* to detect a frequency or band change). The default is 1000 ms (once a second).
- **Fine tuning increment**: each click of the up or down buttons on the VSC form can tune 10, 50 or 100 kHz from the current frequency.
- **Default Autotrack setting**: when VSC is opened, this setting determines whether it automatically starts tracking your radio's frequency changes, or doesn't.
- **SteppIR controller/antenna type**: what kind of antenna and controller[451] do you have? ▶
- **Retract on shutdown**: winds-in the copper-beryllium element tapes when the VSC is closed.
- **Use frequency from Logger32**: select this for Logger32 to pass frequency information through to the VSC, routinely[452].



- **Send band change to Logger32**: Logger32 commands the CAT-connected radio to the relevant band whenever a band button is selected in the VSC form.

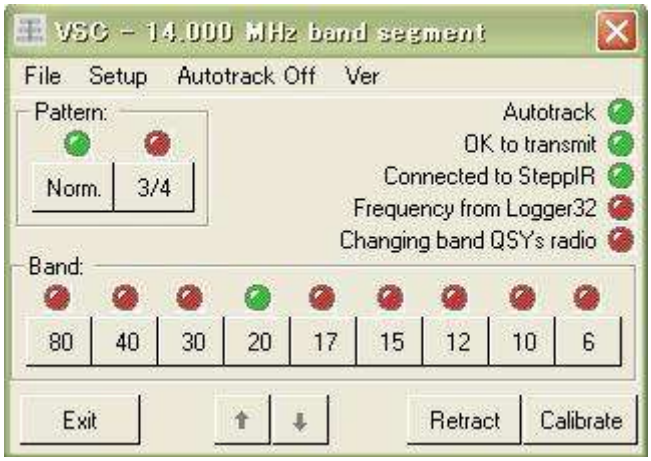
[451] The Original SteppIR controller was *not* an SDA-100.

[452] Don't select this if you would prefer your SteppIR *not* to retune constantly as you tune around a band, avoiding premature wear of the stepper motors, gears and element tapes.

Copyright © 2026 Robert C. Furzer K4CY

43.2 Operating the VSC

The VSC user interface lets you operate and monitor the status of the transceiver interface ▶



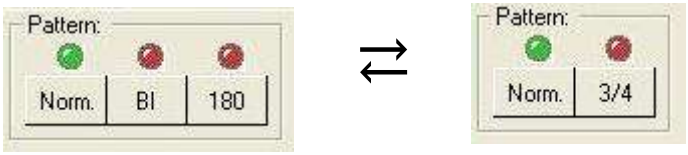
Caption: displays the current band segment to which the antenna is tuned;

Menu bar: to access control functions. From the <**Setup**> menu you can:

- Retract the antenna elements when the VSC is closed;
- Pass through the radio frequency from Logger32; and,
- Send band changes to Logger32.

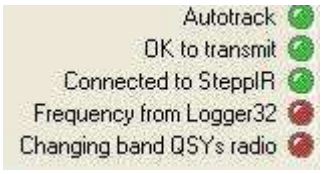
From the VSC menu, you can toggle Autotrack on and off, and check the VSC version in use.

Pattern panel: the configuration depends on the type of SteppIR antenna.

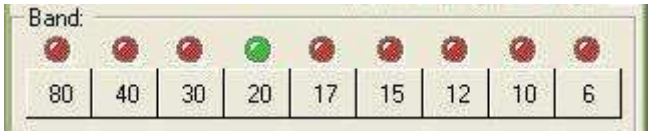


You can select and display the antenna patterns for Yagi and vertical SteppIRs. There is no pattern control or status available for the SteppIR dipole antenna: theoretically in free space, a dipole shows the classic doughnut radiation pattern with no radiation off the ends. In your garden, mounted at a typical amateur height above real ground, you will probably notice the figure-of-8 pattern with limited nulls off the ends.

Status "LEDs" show the status of the VSC-transceiver interface ▶



Band panel: a configurable set of band buttons and "LEDs" to select and display the current status of the antenna. The buttons are enabled/disabled by the <**Show**> check boxes in your bandplan. Click any band button to QSY to that band. If <**Use frequency from Logger32**> is enabled, Logger32 passes new frequencies through to the SteppIR controller as you tune the radio's VFO. The green "LED" denotes to which band the antenna is currently tuned.



Up/Down arrow buttons: increment or decrement the band segment that the antenna is tuned to. The Up and Down arrows are enabled when Autotrack is enabled and <**Use frequency from Logger32**> is disabled (otherwise it overrides the manual tuning).

Retract: retract the antenna element tapes into the reels.

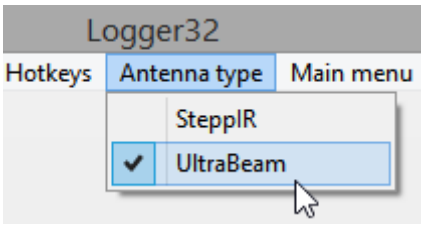
Calibrate: fully retracts all the elements back to a known starting position, then extends them step-by-step to the required lengths.

Copyright © 2026 Robert C. Furzer K4CY

44 Virtual UltraBeam Controller (VUC)

> "Happiness is German engineering, Italian cooking, and Belgian chocolate"
>
> *Patricia Briggs*

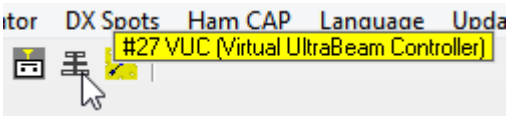
This chapter covers the setup of the VUC. Since the VUC shares much of the functionality of the VSC, it has been implemented with a toggle to select the antenna type.

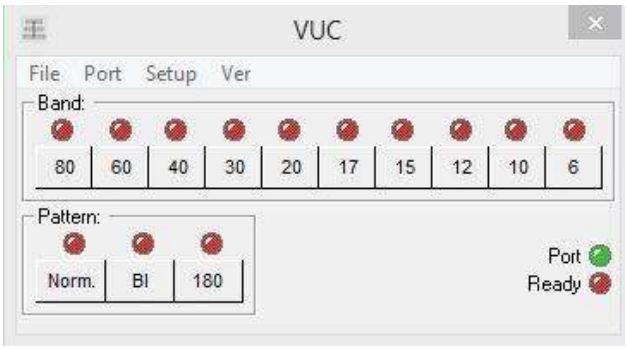


Open **Setup** ⇨ **Antenna type** then click whichever type of antenna you have[453] ▶

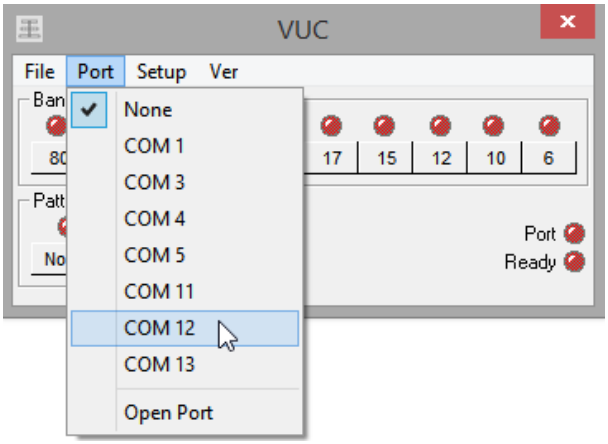
44.1 VUC setup

Having selected UltraBeam, click toolbar beam icon #27 ▶ to open the VUC ▼





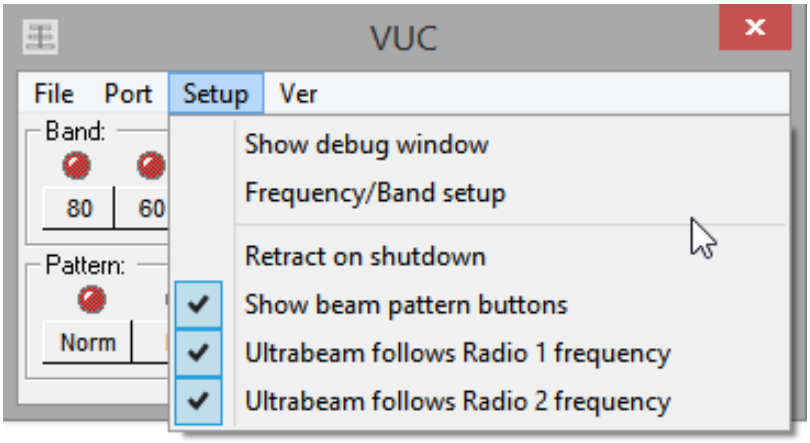
Using <**Port**> on the menu, click to select and then open the applicable serial port to link Logger32 with the VUC ▶



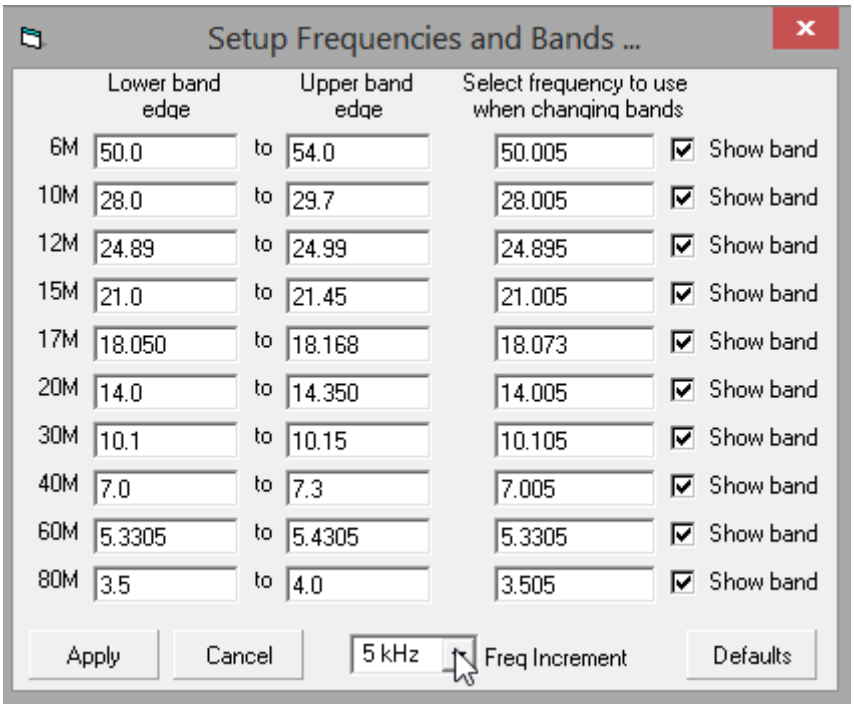
[453] If you have *both*, hopefully you are lucky enough to have two Logger32 systems, one for each …

Copyright © 2026 Robert C. Furzer K4CY

Now setup the VUC ▼



<**Frequency/Band setup**> lets you define the bands ▶



The band edges are obvious enough. The 'frequency to use when changing bands' is a default antenna tuning point for each band – the usual start.

<**Show band**> displays the respective band buttons on the VUC[454].

<**Freq Increment**> sets the minimum change of frequency for the antenna to re-tune[455] ▶

[454] Fantastic as it is, Logger32 cannot magically enable additional bands to be tuned on your antenna beyond its physical design limits. A 40-6m UltraBeam is simply too small to resonate and radiate efficiently on 60 and 80m!

[455] Provided you aren't obsessive about the best possible match and lowest SWR, less wear-and-tear occurs as you tune around a band with a larger step size.

Copyright © 2026 Robert C. Furzer K4CY

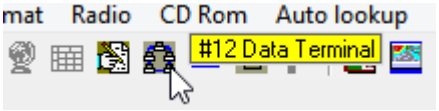
45 PK-232

> "A journey of
> a thousand packets
> begins with a single byte"
>
> *ChatGPT*



The information in this section concerns the AEA PK-232 "PAKRATT" data controller ▲ … and the electrically-identical Heathkit HK-232, built by AEA with a different front panel.

45.1 Data terminal window setup

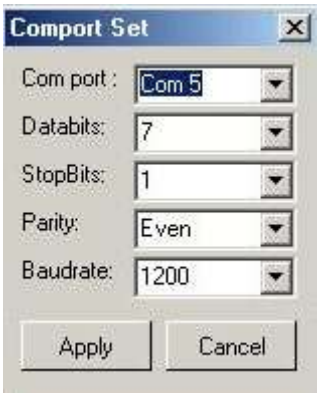


◀ Click the toolbar icon #12 to open the Data Terminal

Then click **Config** ➪ **Port setup** and enter the terminal parameters for your PK232.

If you do not have the PK-232 parameters to hand, try these default settings (your PK232 may be on any COM port, not necessarily COM 5) ▶

These settings should also work if you do not use a battery backup.



Once you have entered the port parameters, click <**Apply**> to close the window and turn on the PK-232. The autobaud routine should work by pressing the main keyboard asterisk <**Shift**+**8**>[456] followed by <**Enter**>.

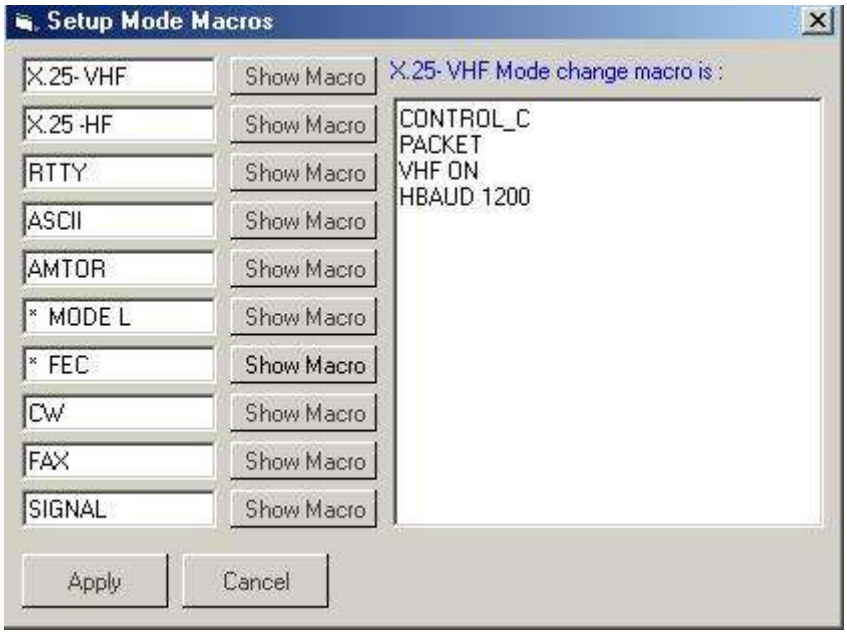
[456] The numeric keypad's * key does not work here. The PK-232 only reacts to the key code emitted by the * key on the main keyboard.

Copyright © 2026 Robert C. Furzer K4CY

45.2 Mode macros

Once you have the PK-232 communicating with Logger32, you can setup the mode macros. In the Data Terminal, select **Config ⇨ Mode macros**.

Enter the modes in your order of preference in the left-hand side of the chart. Click the <**Show Macro**> button and type the desired commands in the right-hand space. Go to the next mode and repeat until you have all the modes covered.



Copy and paste the following text into your *DataTerminal.ini* file to set up the modes as indicated.

---------- Excerpt from *DataTerminal.ini* ----------

[Mode Menu]

Menu #1=X.25- VHF

Menu #2=X.25 -HF

Menu #3=RTTY

Menu #4=ASCII

Menu #5=AMTOR

Menu #6=* MODE L

Menu #7=* FEC

Menu #8=CW

Menu #9=FAX

Menu #10=SIGNAL

[Mode Macros]

Menu Macro #1=CONTROL_C~PACKET~VHF ON~HBAUD 1200

Menu Macro #2=CONTROL_C~PACKET~VHF OFF ~HBAUD 300

Menu Macro #3=CONTROL_C~BAUDOT~RBAUD 45~WIDESHFT OFF

Copyright © 2026 Robert C. Furzer K4CY

Menu Macro #4=CONTROL_C~ASCII

Menu Macro #5=CONTROL_C~AMTOR

Menu Macro #6=CONTROL_C~AMTOR~AL

Menu Macro #7=CONTROL_C~AMTOR~FEC

Menu Macro #8=CONTROL_C~MORSE

Menu Macro #9=CONTROL_C~FAX

Menu Macro #10=CONTROL_C~SIGNAL

The following normal terminal settings are known to work with Logger32:

8BITCONV OFF

ACRDISP 80

AFILTER OFF

ALFDISP ON

AWLEN 7

BBSMSGS OFF

CASEDISP 0 (as is)

DCDCONN OFF

ECHO OFF

ESCAPE OFF

FLOW ON

ILFPACK ON

NUCR OFF

NULF OFF

NULLS 0

PARITY 3 (even)

TBAUD 1200

TRFLOW OFF

TXFLOW OFF

XFLOW ON

Copyright © 2026 Robert C. Furzer K4CY

46 AGW Packet Engine

> "Engineers build the engines that power our dreams"
>
> *Thomas Edison*

If a Telnet connection to the Internet-based DX cluster network is unavailable (*e.g.* due to the lack of an Internet connection), the AGWpe tab in the DX cluster window can connect to a packet cluster node over VHF/UHF amateur radio links using your computer's sound card, a KISS TNC, Baycom modem, DRSI Card, USCC Baycom Card, OE5DXL 9600 G3RUH-compatible modem, YAM 1200/9600 modem *etc*., thanks to the AGW Packet Engine software from SV2AGW.

46.1 AGWpe configuration

Whereas the AGWpe Professional Edition can be set up using a wizard, AGWpe Standard Edition must be configured manually.

The first step is to install the packet engine since Logger32's AGWpe terminal will not work without it! Download and install the AGW Packet Engine (standard or professional editions).

Next, configure AGWpe to work with your TNC, Baycom or YAM modem, DRSI card or computer's sound card.

46.1.1 TNC Configuration

Launch AGWpe from the icon or Windows Start menu.

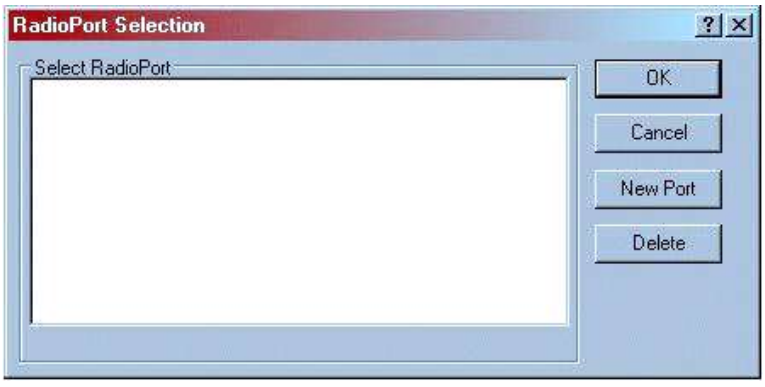


The AGWpe icon appears In the Windows System Tray ▶

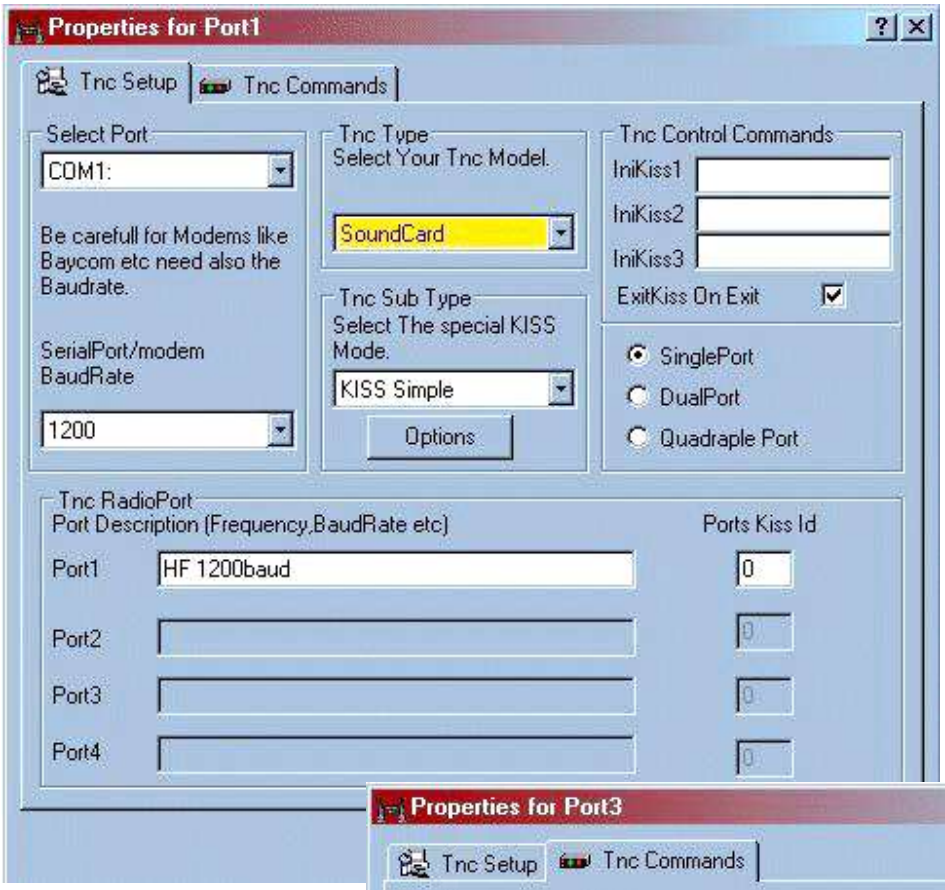


◀ Right-click the icon and click <**Properties**>.

Click <**New Port**> to create and configure a new AGW port for the device you will use ▶



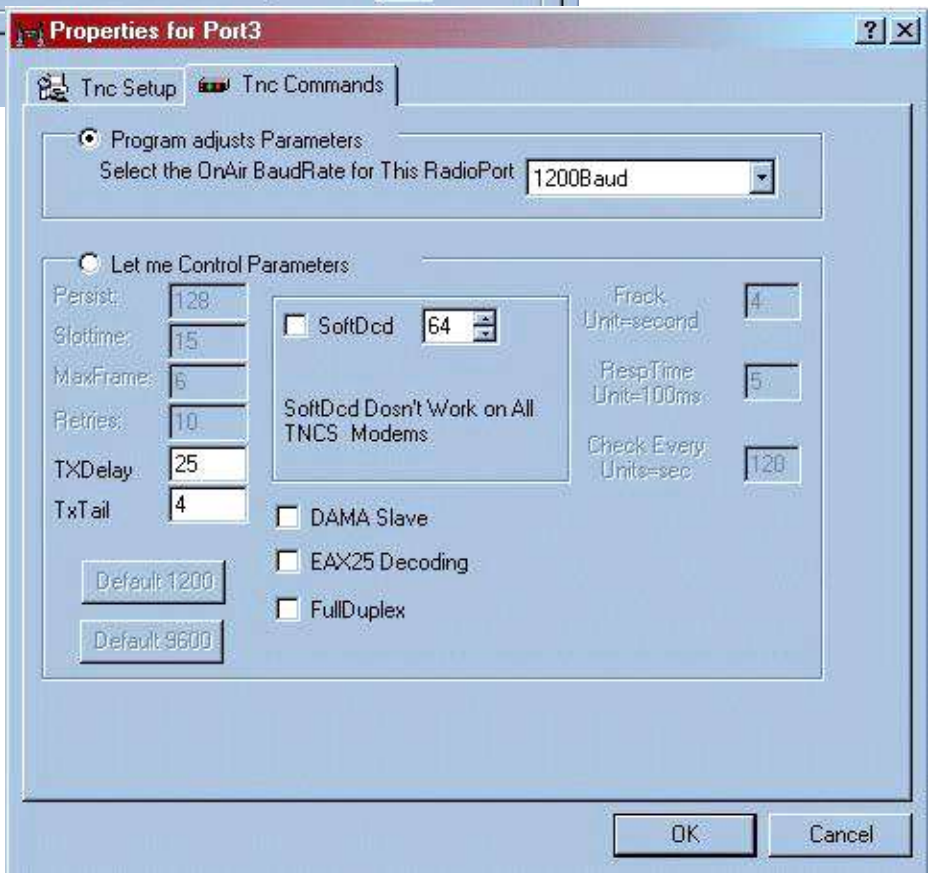
Copyright © 2026 Robert C. Furzer K4CY



◀ Under TNC Type, select your TNC model from a long list that includes the most popular TNCs, packet modems and other possible devices such as a sound card.

Next, configure the port – the COM port number and baud rate. Now, if applicable, you will be able to select the KISS type used by your TNC/Modem.

Next enter a suitable name to identify the Tnc RadioPort.



AGW will automatically select the proper TNC commands according to your on-the-air speed (1200/9600).

If you wish to use your own settings, open the TNC Commands tab ▶ You can then change the TNC's configuration command that affects its on-air performance.

Once you have completed all the above steps, click <OK> to put the configuration into effect.

Now re-start the Packet Engine and a second AGW icon will be shown in the system tray. The TNC icon ▼ tells you the port is working.

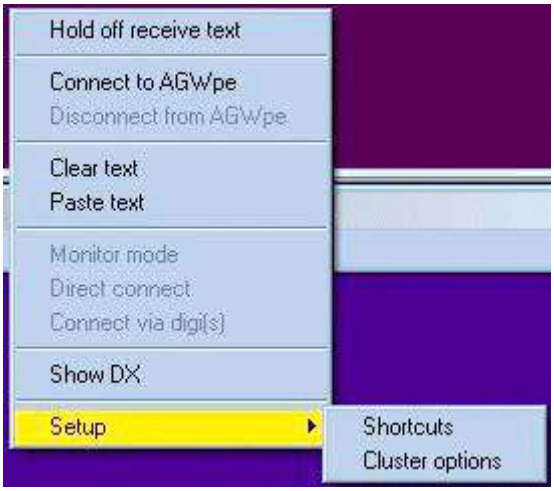


Copyright © 2026 Robert C. Furzer K4CY

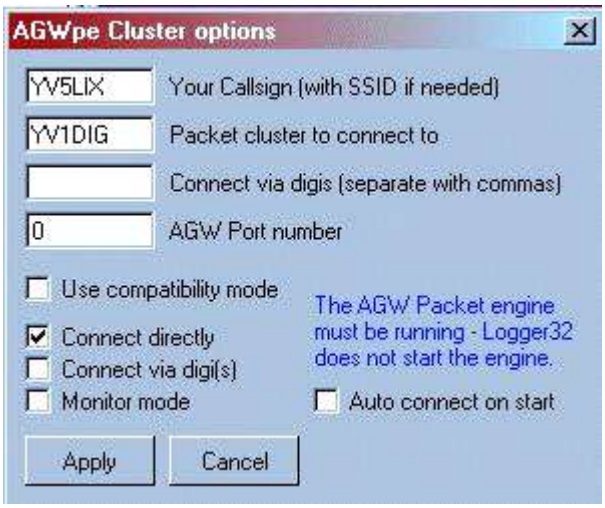
46.1.2 Using the Loger32 AGWPE Terminal

Once you have added the AGW Packet Engine executable program to the Utilities menu using **Tools ⇨ Utility program setup**, you can launch the program from the Utilities menu. Its icon appears on the system tray, near the clock.

In the DX cluster window click to open the AGWpe tab. Right-click the window to open a menu, then click **Setup ⇨ Cluster options** ▶



Then complete the form▼



Enter your callsign, the callsign of the Packet cluster node you wish to connect to, and if you connect via one or more digis (digital repeaters) the relevant callsign/s. Click to select a direct connection, connection via digis or monitor mode (passively monitoring Packet cluster traffic) as you wish. Select <**Auto connect on start**> to connect automatically when Logger32 starts up, provided AGWpe is already running[457].

◀ Finally, click <**Apply**> to save the configuration.

To connect to the cluster and start receiving spots, click <**Connect to AGWpe**>.

If you connect to your cluster via a NODE, in the field "Packet cluster to connect to" enter the call of the NODE. Once connected to the NODE you will be presented with the NODES's command line, where you will issue the command to connect to the cluster.

Note: an AGW Radio Port is not a TNC, for instance, if you are using a dual-port TNC (KAM, PK-232), AGWpe will be configured as 2 Radio Ports and only one TNC icon will be displayed in the window's taskbar (sentry). The DRSI driver supports up to 8 channels (Radio Ports) and again only one icon will be shown in the Windows taskbar at the lower right corner of your screen.

46.1.3 Configuration tips

In the property window are options for configuring each port. Use the property window to change parameters of the AGW ports after they have been created and configured automatically in the Pro version, and for both the initial configuration and subsequent changes in the Standard version.

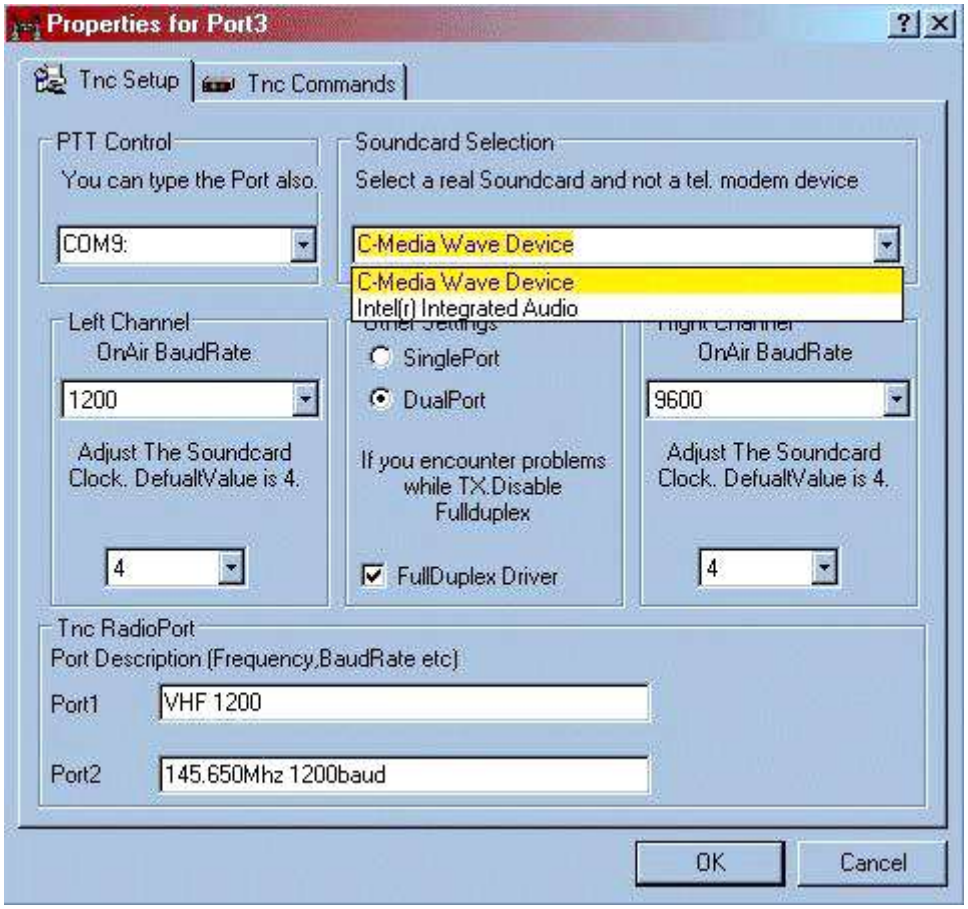
[457] On **Setup ⇨ Setup utilities menu**, click to tick the auto-start box for AGWpe to have it start up automatically when Logger32 starts.

Copyright © 2026 Robert C. Furzer K4CY

Pay attention to the baudrate setting. If you are using an external TNC, enter the serial communication baud rate between your computer and the TNC. If you are using the computer's sound card, enter the on-air baud rate (1200/2400/9600). If you are using a 1200 bps Baycom modem, select 1200.

If your computer has more than one sound card, choose which one you would like to use with AGWpe.

The Standard Edition of the Packet Engine requires you to select and install the proper driver for your device (sound card, Baycom or YAM modem *etc*.) using *Drivers.zip* downloaded from www.raag.org/sv2agw/inst.htm. The Professional Edition handles this for you.

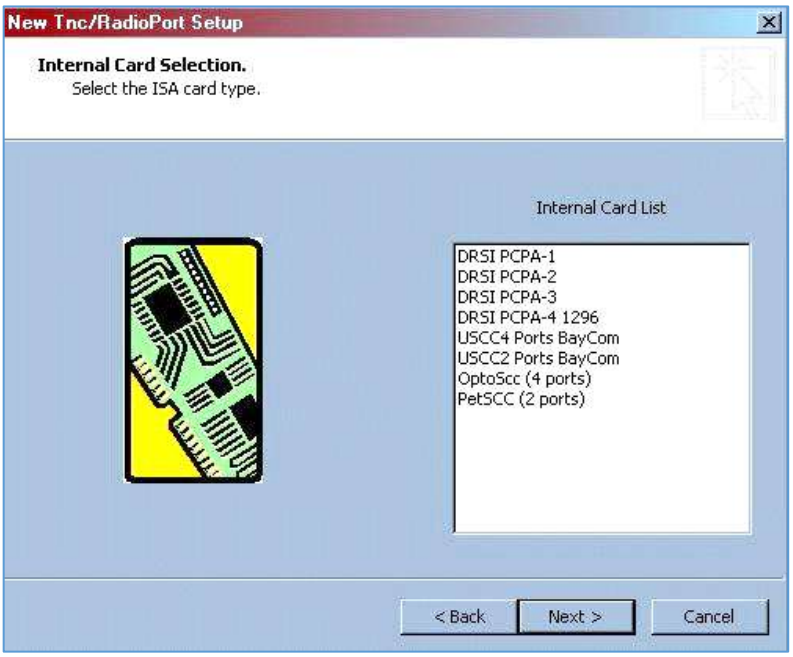


The <**Tnc Commands**> tab allows you to adjust TNC on-air parameters such as TX delay to suit your needs and those of the network you will be operating. In most situations, the default settings are fine.

Sound card operation requires a PTT circuit similar to that used by Logger32's Sound card data window.

Copyright © 2026 Robert C. Furzer K4CY

With a DRSI card, the Professional version makes configuration easy.



1. In the AGW Packet Engine Pro configuration, the wizard should select internal cards.
2. Click to select the DRSI card that you will be using.
3. Enter the base address and IRQ of your DRSI card.
4. Leave the serial port and baud rate as-is.
5. You should now see the dialog with the properties for your DRSI card.
6. Check that the baud rate/s is/are correct for your card, and also confirm the base address and IRQ.
7. If you wish to use hardware clocking for external modems, enter the value 0.
8. In the TNC Radio Port Description, write a friendly name for each Radio Port.

Note regarding 9600 baud operation: not all sound cards operate correctly at 9600 baud. Your radio must also be capable of operating at such speed (which may require internal modifications). Even after performing the required modifications some radios operate very poorly at 9600 baud.

For other modems, refer to the AGW help files.

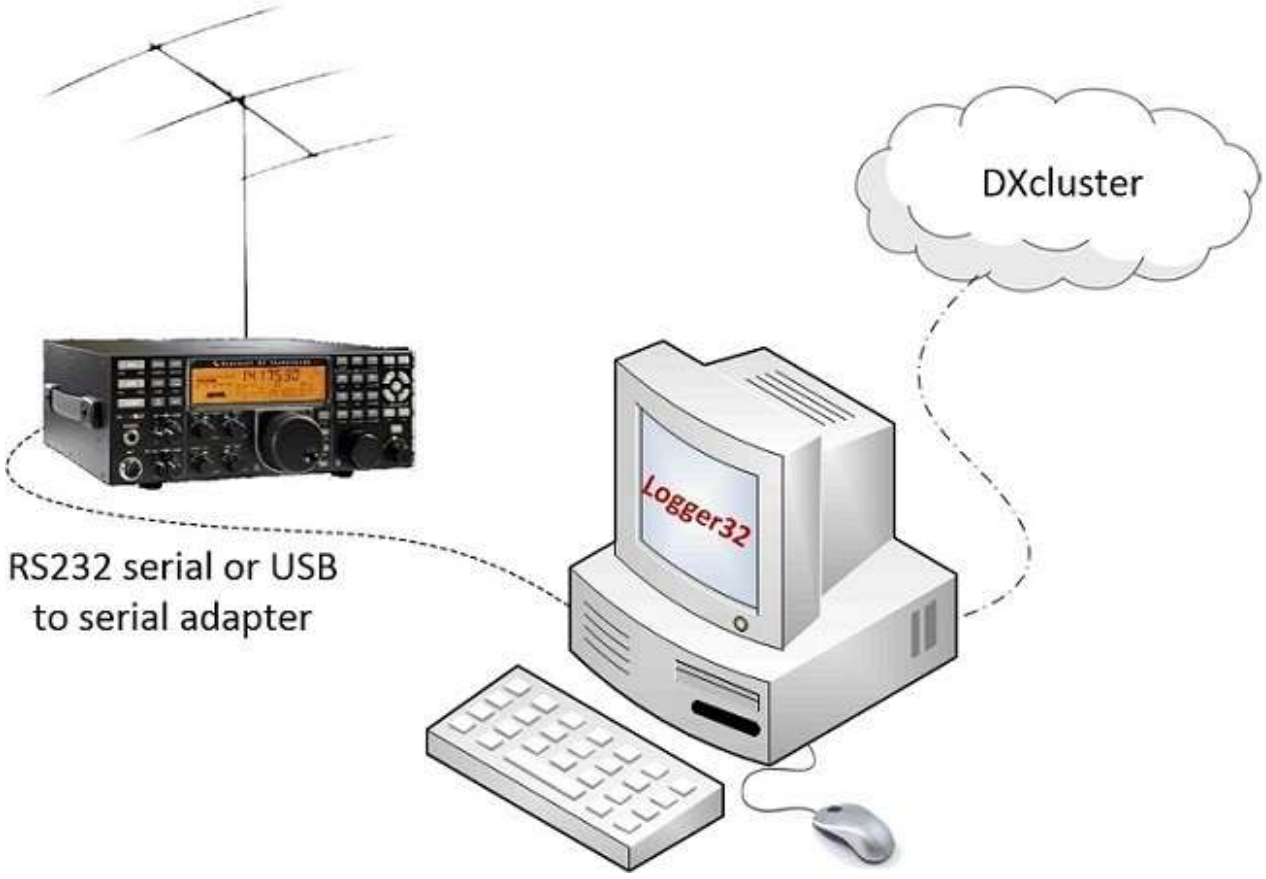
Copyright © 2026 Robert C. Furzer K4CY

47 Elecraft

> "Our design philosophy was clear from the beginning: our radios would offer both high performance and portability"
>
> *Elecraft*

47.1 Elecraft introduction

Logger32 can control Elecraft K2, K3, K4 and KX3 radios through their serial ports (if fitted), for example logging the TX and if applicable RX frequencies automatically from the VFOs, or sending CAT commands to QSY the radio to the frequency and appropriate mode for a DX cluster spot that you have clicked.



Elecraft radios use an extended version of the Kenwood CAT protocol. Logger32 supports all the basics and some of the extended CAT commands directly. You can use the **R**adio **C**ontrol **P**anel's macro buttons for more sophisticated sequences, and send mode-specific CAT commands too.

Copyright © 2026 Robert C. Furzer K4CY

47.1.1 Elecraft CAT setup

Elecraft radios can be CAT-controlled using serial (USB or RS232) connections to a PC, depending on the model and the options fitted.

The serial port speed and other settings need to match at both ends of the cable *i.e*. the CAT port settings in the radio's built-in menu should be used in Logger 32's radio setup form. One little complication is to figure out the *number* of the PC port physically connected to your radio, and (in the case of the K4) which port on the radio is being used for CAT control.

You will find specific setup instructions for particular Elecraft models below, after some general notes on the Elecraft CAT command format and filter/mode settings.

47.1.2 Elecraft CAT commands

CAT commands for Elecraft radios use $**Command** $ strings in Logger32, telling the PC what to send to which radio through its CAT cable. The command syntax is as follows:

- Each CAT command sequence starts with the first of a matching pair of dollar (**$**) symbols enclosing the entire string;
- Next comes the keyword **Command**[458] followed by a **space**;
  - Then two letters specifying the actual **CAT command** to be sent to the radio. This tells the radio which configuration setting or option you are trying to set or change;
  - Then follows any command **parameters**, telling the radio what value to set or change to;
  - A semicolon (**;**) denotes the end of that command;
- Optionally, you can now add *further* CAT commands with their associated parameters in the same format, each one terminated with a semicolon.
  - The radio will work its way sequentially left-to-right through the command sequence (macro) as specified;
- End the entire CAT command string with a final **$** symbol, matching the one at the start.

Omitting or using the wrong characters in a command string can cause unexpected results:

- Logger32 will display an error message if it finds a gross problem with the command string (*e.g*. a typo in the keyword "**command**" or a missing **space** or **dollar**). It will stop processing the macro and will not send commands to the radio;
- Errors in the actual CAT commands or parameters passed through to the radio are less straightforward. The radio will generally ignore invalid commands (*e.g*. if you try to enable a non-existent filter, or a typo in the command itself), but it will slavishly follow valid commands (*e.g*. turning the power or volume up full) even if that was not what you wanted to do!

Hinson tip: while developing and before using Elecraft macros, put the radio into "TEST" mode, then trigger your macros to check what actually happens on the radio. CAT commands for QSYing, splits, filter changes *etc*. should all work as intended but the radio will not transmit, saving your blushes if you made mistakes with the macro coding. It helps to build up longer sequences of CAT commands gradually, adding commands and testing at each stage, since the radio executes sequential commands so quickly that it may not be obvious what has happened.

[458] CAT commands and keywords are case-insensitive *e.g*. "**Command**" and "**command**" both work just fine.

Copyright © 2026 Robert C. Furzer K4CY

47.1.3 Elecraft data modes and filters

Elecraft radios automatically store and recall the last-used data mode settings per band *i.e*.

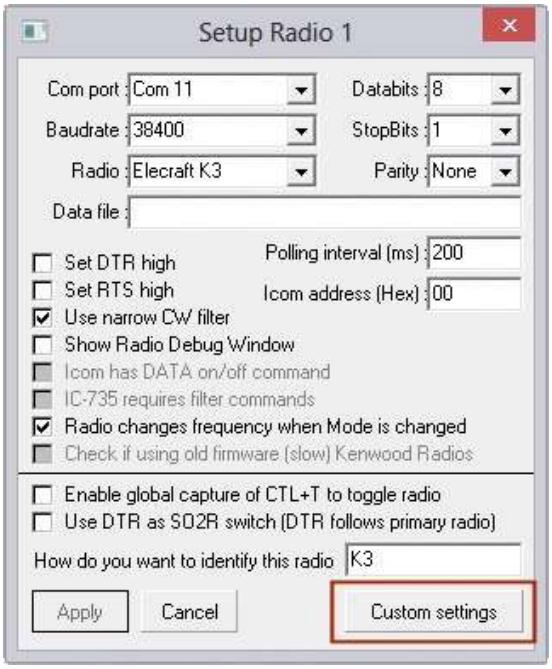
- DATA A for FT#, PSK31, JT65 and other sound card-based audio frequency-shift keying modes;
- AFSK A is a variant optimized for sound card-based RTTY;
- FSK D or PSK D are used for frequency-shift keying on RTTY or PSK, using data or paddle inputs.

When you change to a different band, the radio will automatically change itself to the mode you happened to be using the last time you were on that band … which may not be appropriate.

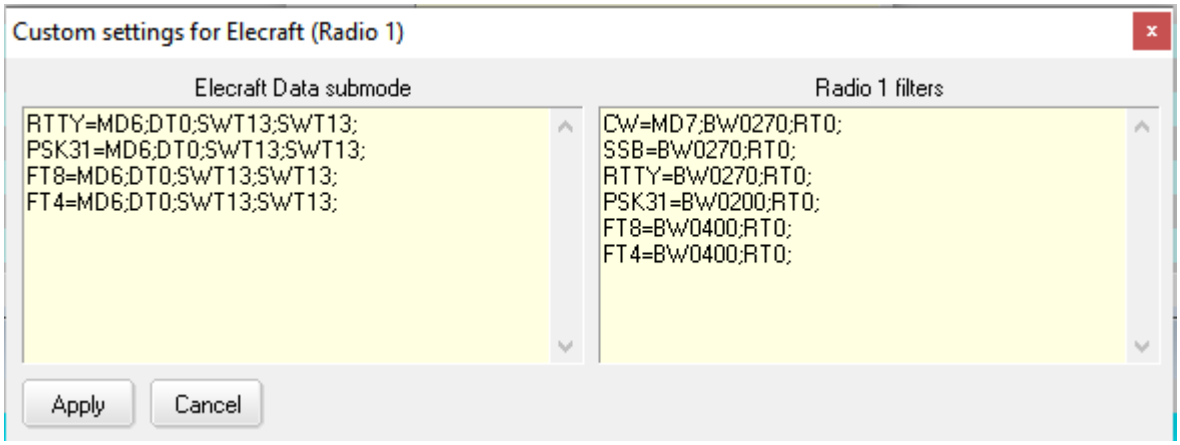
Likewise the radio saves and recalls the filter settings per band. For instance, if you had your filters opened right up while tuning around the SSB sub-band, the same wide filters will be recalled and applied automatically if you later click a CW DX spot for that band.

When you click a DX spot, you may prefer instead to send specific mode- and filter-setting commands to the radio according to the 'operating mode' in Logger32, regardless of what you last used on that band.

To adjust your filter settings, open **Setup ⇨ Radio ⇨ Radio 1|2 configuration** then click the <**Custom settings**> button ▶



Specify your choice of filter settings per mode using the relevant CAT commands (see the Elecraft Programmer's Reference on the applicable model for details) *e.g.* for a K3 ▼



The settings are stored in *C:\Logger32\Logger32.INI* when you click <**Apply**> but only take effect when you next change band and/or mode through Logger32 *e.g.* by clicking a DX spot for a different band or mode, or by entering a frequency into the log entry pane.

Copyright © 2026 Robert C. Furzer K4CY

47.1.4 Elecraft VFO controls and subreceivers

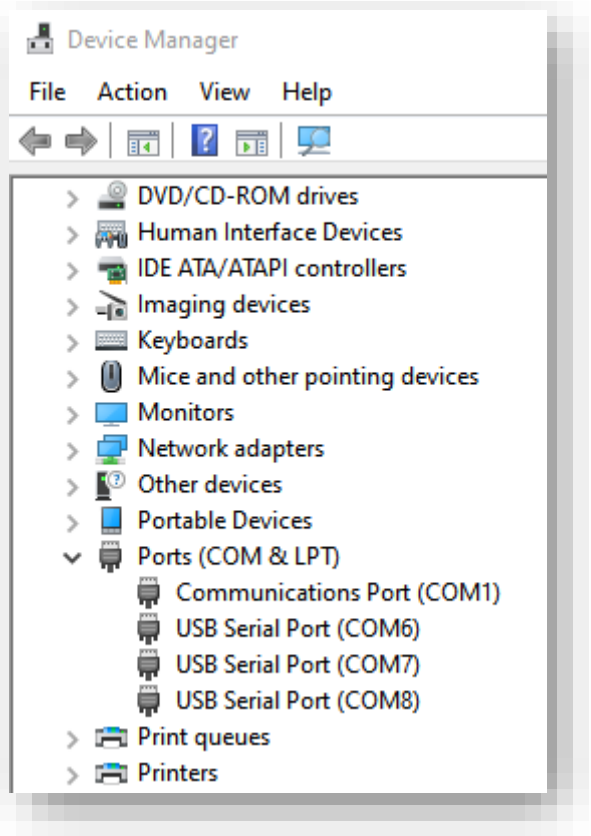
VFO- and receiver-specific CAT commands generally apply to VFO A and the main receiver by default, but may include an optional **<24>**[459] to apply the command to VFO B and the sub-receiver (if fitted).

47.2 Elecraft K2

1. For CAT control of a K2, you first need to build and install the optional KIO2 module in the K2, providing a 9-pin AUX I/O connector on the K2's rear panel. Follow Elecraft's excellent instructions.
2. On the K2, turn PORT ON in the secondary menu. See the Elecraft K2 manual for help with this.
3. Connect the K2 to the PC using a special custom-made serial cable if your PC has a free RS232 port, or to a USB-serial adapter if it only has USB ports. See the KIO2 manual for wiring instructions.

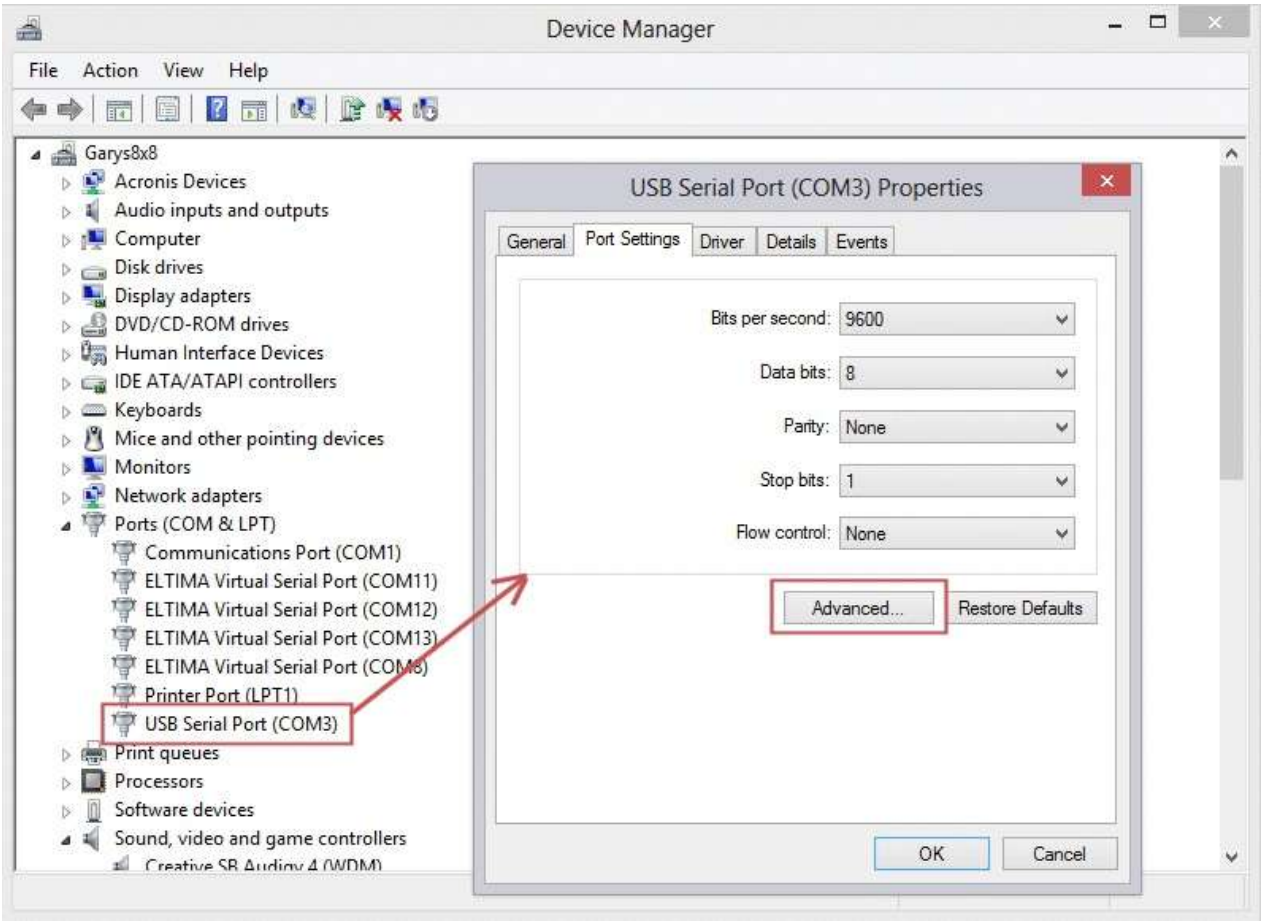
   Note: despite using a DB-9 9-pin socket, **the K2 AUX I/O port is *not* RS232**. The KIO2 manual warns "***CAUTION**: Do not use a pre-assembled cable of any kind (printer cable etc.) between the KIO2 and a computer. Some pins on the KIO2 provide special signals not intended for use with an RS232 interface, and you could damage the K2, KIO2, or the computer if these lines are used incorrectly. You must assemble your own K2-to-computer cable.*"
4. Check the COM port number on the PC. In Windows, open Device Manager using <**Win+X>M** and click the > symbol to expand the "Ports (COM & LPT)" section ▶

   In that example on a PC running Windows 10, I had one true RS232 serial port (COM1) plus three COM ports using USB-serial adapters. To find out which COM port my K2 was using, I simply unplugged the radio's USB-serial adapter cable from the PC to find that COM8 disappeared from the list. When I plugged it back in, COM8 reappeared. Mystery solved!

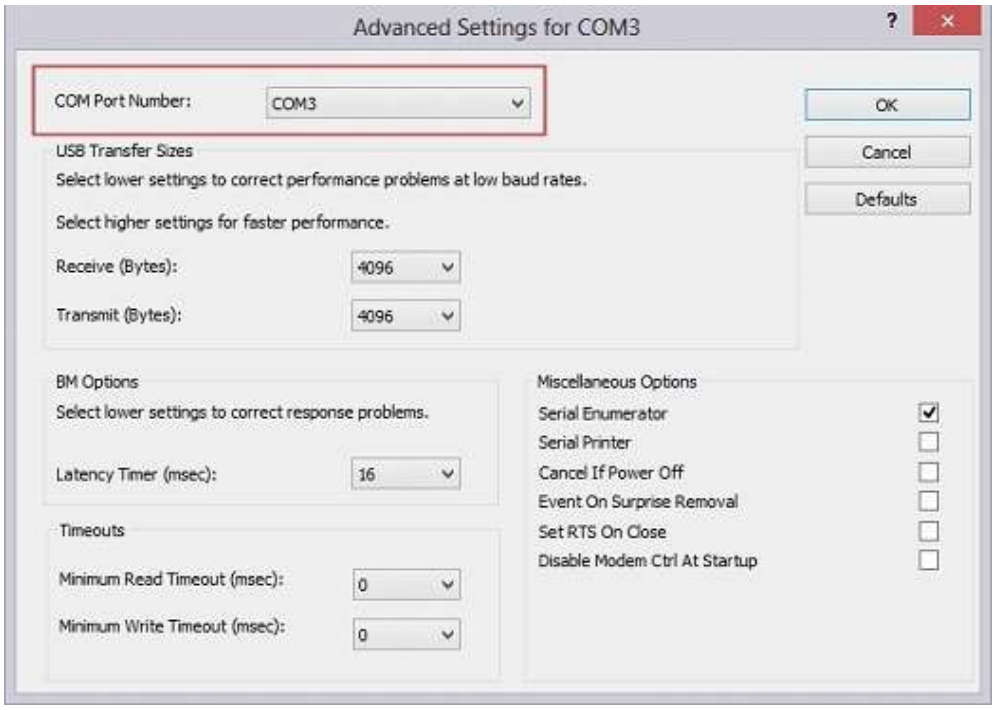


[459] **<24>** is the HEX code for "$". Think of it as "Sub" if that helps. You cannot simply embed a plain "$" in the $**Command** macro string because Logger32 sees the matching pair of dollars and prematurely closes the string without sending the remainder. **Chr(36)** works the same way, 36 being the ASCII code for "$".

Copyright © 2026 Robert C. Furzer K4CY



Now click <**Advanced**> and top left is a pull-down box to select the COM port number. Pick a COM port that is not already "in use".



If all your COM ports are "in use", you either have a lot of serial devices connected, or more likely Windows has reserved the ports for devices that may no longer be connected. You may be able to tell what devices they were reserved for by selecting **View ⇨ Show hidden devices** in Device Manager. You can delete redundant device ports, or even pick the same COM port

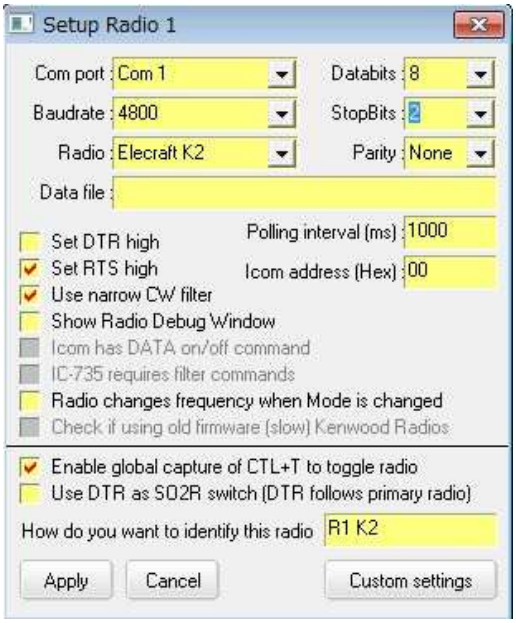
Copyright © 2026 Robert C. Furzer K4CY

number to be used by the radio connection, ignoring the fact that it is “in use”. You may need to experiment with different port numbers.

5. In Logger32, click **Setup ⇨ Radio ⇨ Radio 1|2 configuration**[460]
6. Configure Logger32’s parameters for the K2, then click <**Apply**> ▶

   Pick the appropriate COM port number. The baud rate for the K2 should be 4800 bps with 8 data bits, 2 stop bits and no parity.

   <**Polling interval (ms)**> determines how often Logger32 checks the radio for any changes. A low value such as 250 milliseconds makes Logger32 respond more quickly if you QSY the radio, but loads both the PC and K2 more heavily. A higher value such as 1000 milliseconds is less responsive and less demanding on the equipment. Experiment with this value to find a compromise that works best for you.



7. Provided the K3 is turned on, plugged in and connected to Logger32, the current VFO frequency should be shown at the top of the log entry pane, and the “Radio 1|2” text on the lower status bar should be blue.

[460] Logger32 can connect to and control up to two radios for SO2R operation. Since each radio’s CAT connection is configured independently on different COM ports, they need not be the same make or model of radio.

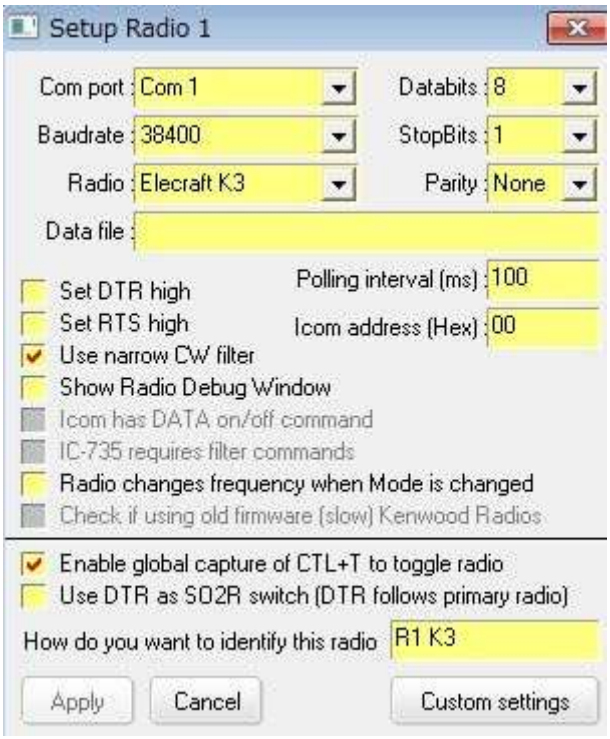
Copyright © 2026 Robert C. Furzer K4CY

47.3 Elecraft K3/K3s

Note: since the K3 and K3s are identical from the CAT perspective, we use “K3” in this manual to refer to both models.

47.3.1 K3 configuration

1. Using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**, check the serial port speed on the K3 under the RS232 entry on the radio’s config menu. The default speed is 38400 baud which suits Logger32 just fine ▶
2. Connect the K3 to the PC using an RS232 serial cable if your PC has a free RS232 port, or a USB-serial converter if it only has USB ports. See the Elecraft K3 Operator Manual for cable wiring instructions.
3. Check the port number on the PC (see K2 setup step 4 for details).
4. Configure Logger32’s parameters for the K3 in the Setup Radio window (see K2 setup step 5), then select the <**Apply**> button. Pick the appropriate COM port number. The baud rate for the K3 should be 38400 baud with 8 data bits, 1 stop bit and no parity.



5. Setup is now complete. Provided the K3 is turned on, plugged in and connected to Logger32, the current VFO frequency should be shown at the top of the log entry pane, and the “Radio 1|2” text on the lower status bar should be blue showing that the CAT port is open.

47.3.2 K3 macros

These macros control VFO A and the main receiver:

- $**Command FA 00007000000;**$ Set the **F**requency of VFO **A** to 7,000,000 Hz (7 MHz)
- $**Command BW0025;**$ Set main receiver IF **B**and**W**idth to 250 Hz
- $**Command BW0270;**$ Set main receiver IF **B**and**W**idth to 2700 Hz
- $**Command NB1;**$ Turn main receiver **N**oise **B**lanker on
- $**Command NB0;**$ Turn main receiver **N**oise **B**lanker off

The following macros control VFO B and also the optional sub-receiver if fitted:

- $**Command FB 00007000000;**$ Set the **F**requency of VFO **B** to 7,000,000 Hz (7 MHz)
- $**Command BW<24>0025;**$ Set sub-receiver IF **B**and**W**idth to 250 Hz
- $**Command BW<24>0270;**$ Set sub-receiver IF **B**and**W**idth to 2700 Hz
- $**Command NB<24>1;**$ Turn sub-receiver **N**oise **B**lanker on
- $**Command NB<24>0;**$ Turn sub-receiver **N**oise **B**lanker off

Read all about the K3 CAT commands in the K3S Programmer’s Reference manual.

Copyright © 2026 Robert C. Furzer K4CY

47.4 Elecraft K4

Refer to the latest Elecraft K4 Operating Manual for setup instructions[461].

The K4 has an Ethernet port (for software updates, remote control and streaming data such as audio received and to be transmitted), 4 USB sockets (3 type A plus 1 type B for CAT and audio I/O), a DE9 true RS232 port (for remote control, to send VFO frequency information to external devices such as antenna controllers, and for FSK data input) *and* a DE15 connector[462] for accessories such as Elecraft amplifiers and antenna switches. Using different ports, the K4 can be controlled simultaneously by multiple clients such as local *and* remote Logger32 systems: they all interact with a server inside the K4 which services the commands and data streams independently.

> *With the Elecraft K4 you have multiple CAT ports. I like to use one for FT8 and one for the radio CAT. That way I can use the RCP at any time to send CAT commands to the radio while operating FT8 or any of the digital modes.*
>
> *Jim K4JAF*

47.4.1 K4 setup

1. Physically connect any one of the USB or RS232 ports on the K4 to a USB or RS232 port on your PC using the appropriate cable.
2. Select the corresponding USB or Com port number in Logger 32's Setup Radio form.
3. Set the port speed and other parameters in Logger32 to match the K4's settings, as defined in the Serial Connection options of the K4 menu. The defaults are similar to the K3.

The K4 Direct Sampling Transceiver community on *groups.io* is a tremendous resource. There you will find ample setup hints.

47.4.2 K4 macros

Refer to the K4 Programmer's Reference manual for the CAT commands.

The K4 extends the K3's CAT functionality with some additional CAT commands (*e.g*. to control its built-in menu or P3-style panadapter) and parameters (*e.g*. for the extra menu options and front panel controls on the K4).

The CAT format remains similar to the K3: generally a couple of letters identifying the CAT command, then any parameters, ending with a semicolon.

Here are some basic K4 macro examples:

- $**Command SW132;**$ Enter TEST mode (as if you tapped front panel key switch #132)
- $**Command MD3;**$ Set the K4 to CW on VFO A
- $**Command MD<24>1;**$ Set the K4 to LSB on VFO B

[461] Check the Elecraft website for the latest/current manuals.

[462] The DE15 *looks* like a VGA connector but it is ***not*** VGA!

Copyright © 2026 Robert C. Furzer K4CY

- $**Command AB0;**$ Transfer the frequency and settings of VFO A to VFO B
- $**Command SW16;**$ Enter TUNE mode.
- $**Command RX;**$ Instantly halt any transmission (switch immediately to RX)
- $**Command KY xxxxxxx**;$ In CW, send the string xxxxxxx (*e.g*. replace with your callsign.

  Note: this uses CAT to send text in CW using the built-in keyer. A CW interface is not needed.
- $**Command PB4;**$ In voice modes, transmit the audio content of voice memory M4.

  Note: pre-record your message (such as your callsign) into the K4, following the instructions in the K4 Operating Manual.
- $**Command MI0;**$ Select the microphone plugged into the front panel
- $**Command MI1;**$ Select the microphone plugged into the rear panel

Here are some more advanced K4 macro examples that can be programmed into buttons on the **R**adio **C**ontrol **P**anel. Multiple CAT commands can be conCATenated on the same line for speed or split across multiple lines for readability. Either way, *every command must end with a semicolon*.

- $**Command RA/;RA<24>/;**$ ATTN A/B toggle.

  Note: in place of RA, use PA for the **P**re**A**mp, NR for **N**oise **R**eduction *etc*.
- $**Command DARC;DARS;DE999;DA0;DAMS1;**$ Make a 10 second off-air recording.

  Note: uses the built-in **D**igital **V**oice **R**ecorder.
- $**Command DAMP1;DE999;DA0;**$ Play back 10 seconds of off-air recorded audio.
- $**Command DAME1;**$ Erase the DVR memory.
- $**Command FA21.035;MD3;FB21.040;MD<24>3;#SPN70000;FC;FA21.020;PA1;PA<24>1;**$ (!)

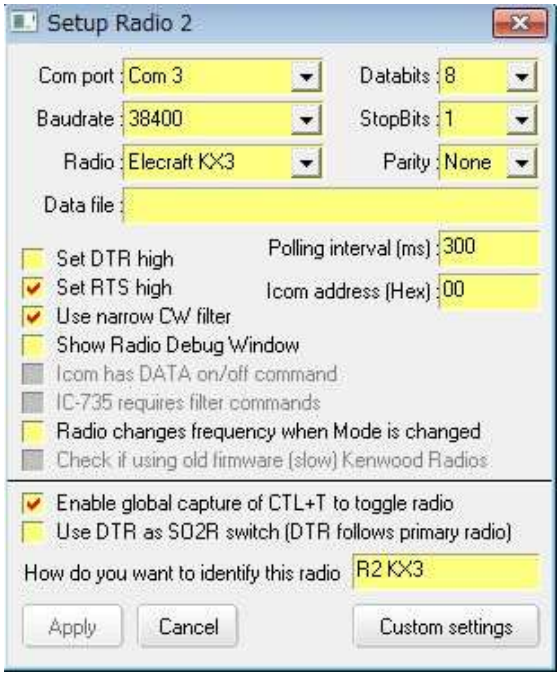
  This complex sequence of CAT commands puts the K4 on 15m CW with the operator's preferred settings. Consider coding a set of these, using color-coded RCP buttons to jump directly to the bands, modes and settings of *your* choice, like this ▼



Copyright © 2026 Robert C. Furzer K4CY

47.5 Elecraft KX3

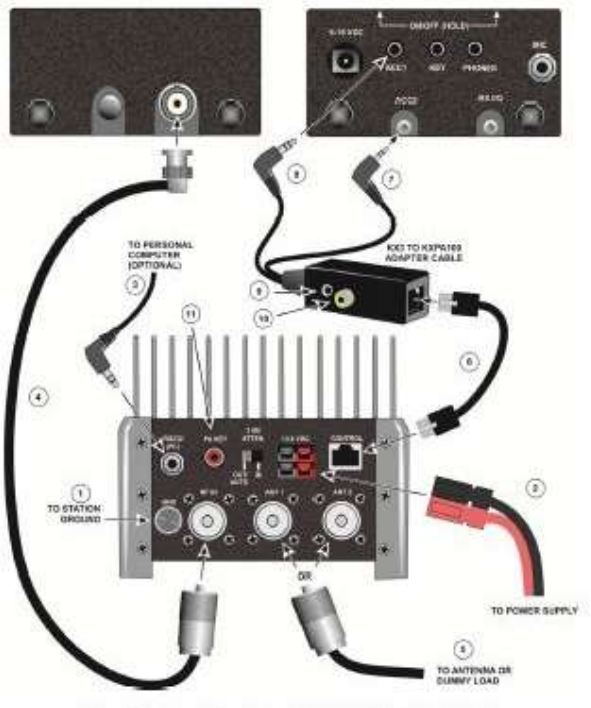
47.5.1 Using the QRP KX3 *without* the KXPA100



1. Connect the PC serial port to the KX3's ACC1 jack e.g. using an Elecraft option cable (KXSER or KXUSB).
2. On the KX3, check that the Com port is set to 38400 baud in the radio's menu.
3. Check the port number on the PC (see K2 setup step 4 for details).
4. In Logger32, configure the radio using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** (see K2 setup step 5) to the same baud rate, 8 data bits, 1 stop bit and no parity ▶
5. Confirm that the KX3 is turned on, plugged in and connected to Logger32 (see K2 setup step 7).

That's it, you're done!

47.5.2 Using the KX3 *with* the KXPA100



1. Connect the PC serial port to the KXPA100's RS232 jack using Elecraft option cable (KXSER or KXUSB). Refer to the KXPA100 Owners' manual.
2. Connect the KXPA100 to the KX3 using a KXPACBL KXPA100 integrated cable.
3. On the KX3 set the following parameters in the menu:
   - RS232 38400 (other speeds may not work)
   - PA enabled
   - TUN PWR NOR
4. On the PC, check the COM port number (see K2 setup step 4).
5. In Logger32, configure the radio using the Configure radio menu (see K2 setup step 5) to the same 38400 baud rate, 8 data bits, 1 stop bit and no parity.
6. Confirm that the KX3 & KPA100 are turned on, plugged in and connected to Logger32 (see K2 setup step 7).
7. Check that it works. If Logger32 fails to connect to the radio, try closing Logger32 and turning off both the KX3 and the KXPA100. Turn the KX3 & KXPA100 back on and wait until the KXPA100's ATU mode LED is lit *before* starting Logger32.

Copyright © 2026 Robert C. Furzer K4CY

48 ICOM

> "Icom Inc.'s roots are in designing, engineering, and manufacturing highly advanced, compact solid-state radio equipment for use in the amateur (ham) radio industry"
>
> *ICOM*

48.1 ICOM hex addresses

These are the default *hexadecimal addresses* directing CAT commands to most ICOM radio models. They are all two-digit hexadecimal numbers, where *0* is the number zero *not* the letter Oh.

IC-1271 *24*
IC-1275 *18*
IC-127A/E *18*
IC-271 *20*
IC-275A/E/H *10*
IC-375A *12*
IC-471 *22*
IC-475A/E/H *14*
IC-575A/H *16*
IC-7000 *70*
IC-703 *68*
IC-705 *A4*
IC-706 *48*
IC-706MKII *4E*
IC-706MKIIG *58*
IC-707 *3E*
IC-718 *5E*
IC-7200 *76*

IC-725 *28*
IC-726 *30*
IC-728 *38*
IC-729 *3A*
IC-7300 *94**
IC-735 *04*
IC-736 *40*
IC-737/A *3C*
IC-738 *44*
IC-7400 *66*
IC-7410 *80*
IC-746 *56*
IC-746 PRO *66*
IC-751 *1C*
IC-755 *0A*
IC-756 *50*
IC-756PRO *5C*
IC-756PROII *64*

IC-756PROIII *6E*
IC-7600 *7A**
IC-761 *1E*
IC-7610 *98*
IC-765 *2C*
IC-7700 *74**
IC-7760 *B2*
IC-775 *46*
IC-78 *62*
IC-7800 *6A**
IC-781 *26*
IC-7851 *8E*
IC-820/H *42*
IC-821 *4C*
IC-910 *60*
IC-970 *2E*
IC-9700 *A2*
IC-970A/E/H *2E*

IC-R10 *52*
IC-R20 *6C*
IC-R7000 *08*
IC-R71 *1A*
IC-R7100 *34*
IC-R72 *32*
IC-R75 *5A*
IC-R8500 *4A*
IC-R9000 *2A*
IC-R9500 *72*
IC-RX7 *78*
TEN-TEC Paragon II *04* with N4PY chip (uses the IC-735 protocol) More here

These 4 ICOMs will QSY VFO B by **shift-clicking spots.*

Copyright © 2026 Robert C. Furzer K4CY

48.2 ICOM interfacing

Recent ICOMs (*e.g*. IC-7xxx) have a built-in USB interface for both CAT and audio signals *but* installing the ICOM USB driver is recommended or required if the USB to UART bridge controller shows an error in the Windows device manager, despite using the correct ICOM driver.



Older ICOMs need a CT-17 CI-V interface or equivalent ▶ to convert levels between the PC and radio.

48.3 ICOM filter setup

A chart taken from the CI-V specification shows the relationship between control numbers and filter widths ▶

Control	Filter width
01 Width 1	Wide
02 Width 2	Narrow
03 Width 3	Very narrow

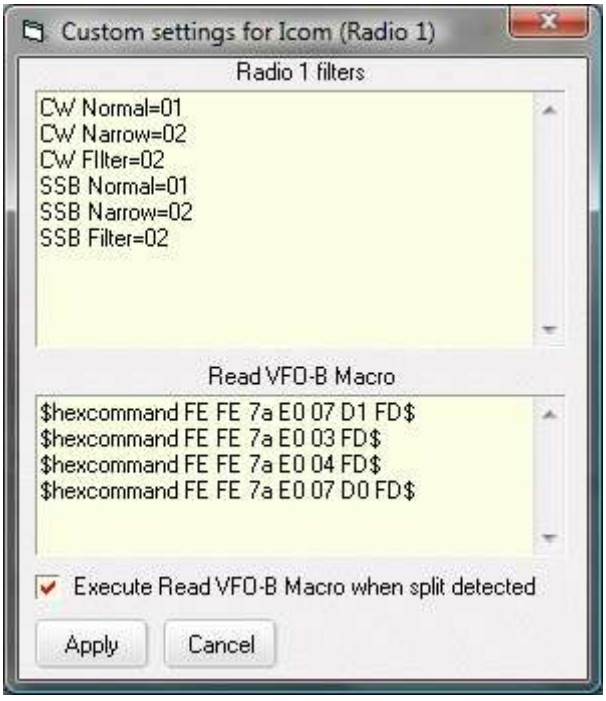
Enter the desired data in the Radio 1 filters list box and click <**Apply**>:

- CW Normal=**xx**
- CW Narrow=**xx**
- SSB Filter=**xx**

… where **xx** is the filter command number 01, 02 or 03. [Non-ICOM radios typically select default filters for each operating mode.]

Logger32 defaults to filter 01 for CW normal and filter 02 for CW narrow. If the defaults are unsuitable (*i.e*. you want to use filters other than the defaults), the following section(s) need to be added to the filter setup.

Here is the setup for an IC-7600 ▶



If you have more than one CW filter, consult your manual and experiment with the filter numbers. The SSB filter lines are only needed if your radio supports more than one SSB filter setting.

Copyright © 2026 Robert C. Furzer K4CY

48.3.1 Setting a VFO frequency

The following example commands use the hex address 4E (the default for an IC-706MKII).

To use LSB on 80 meters, define the following in a macro button and click to execute the macro:

- $**HexCommand FE FE 4E E0 06 00 FD**$ Set LSB without changing the filter
- $**QSY3681.50**$ QSY to 3681.50 kHz.

Likewise to use USB on 20m:

- $**QSY14169.50**$ QSY to 14169.50 kHz.
- $**HexCommand FE FE 4E E0 06 01 01 FD**$ Set USB with a wide filter

The order of the QSY and HexCommands is not important.

48.3.2 Some mode and filter selection commands

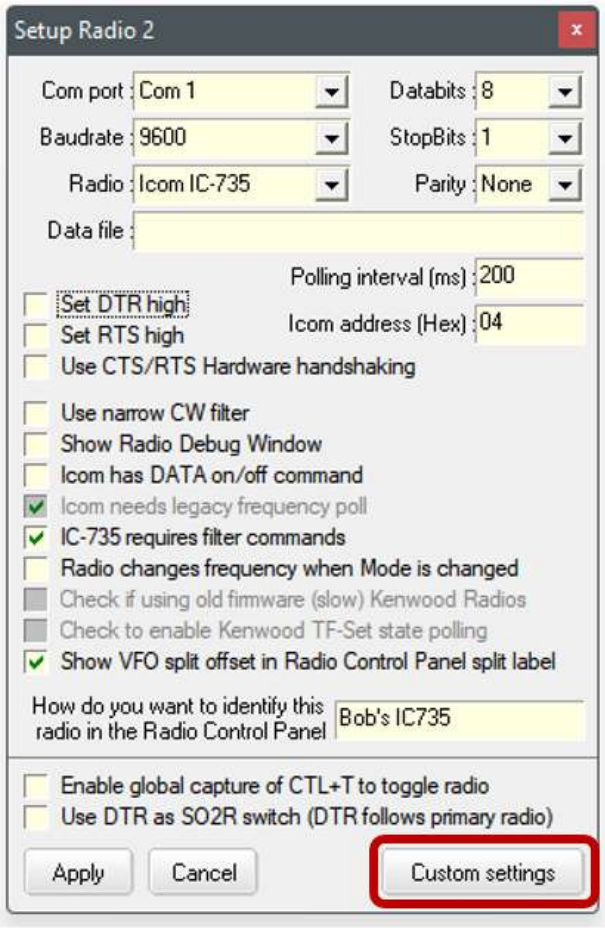
- $**HexCommand FE FE 4E E0 06 01 01 FD**$ USB wide
- $**HexCommand FE FE 4E E0 06 01 02 FD**$ USB narrow
- $**HexCommand FE FE 4E E0 06 00 01 FD**$ LSB wide
- $**HexCommand FE FE 4E E0 06 00 02 FD**$ LSB narrow
- $**HexCommand FE FE 4E E0 06 04 FD**$ FSK wide
- $**HexCommand FE FE 4E E0 06 04 02 FD**$ FSK narrow

48.4 ICOM VFO B display

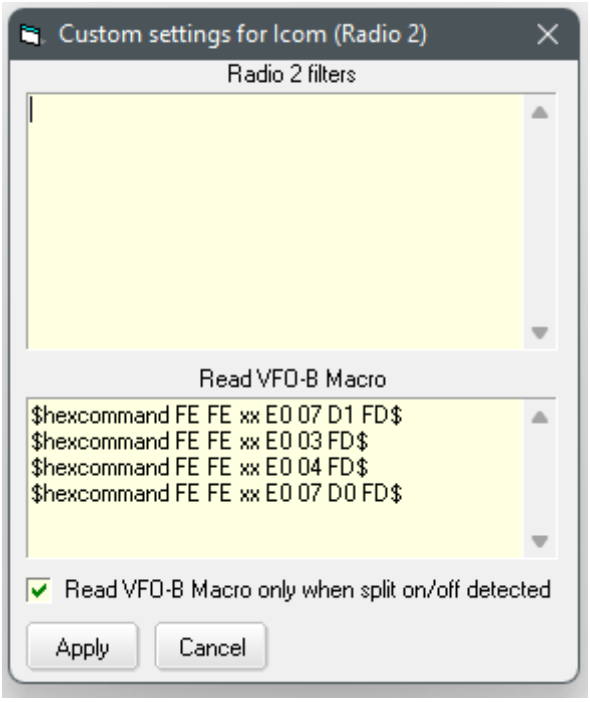
Since some ICOM radios do not have a CI-V CAT command to interrogate VFO B status, a special Logger32 macro is available to display and allow logging of the VFO B frequency and mode.

The macro is $**IcomVFOB**$.

To personalize the macro, in the **Setup ⇨ Radio ⇨ Radio 1 [**or **2] configuration**, click the <**Custom settings**> button ▶



Copyright © 2026 Robert C. Furzer K4CY



◀ To compensate for the deficient ICOM CAT protocol, a macro sending the four commands shown here can read the frequency and mode of VFO-B.

These four commands should work with most late model ICOM radios. In sequence, they:

- Switch to VFO B;
- Poll (grab) the frequency;
- Poll (grab) the mode;
- Switch back to VFO A.

xx is the ICOM's CI-V address. Replace **xx** in each command with the correct Hex address for *your* ICOM radio model.

Verify the commands for your ICOM by referring to the dusty radio instruction manual: you remember, the one you cast aside after excitedly unpacking your shiny new radio.

To activate this function, tick <**Execute Read VFO-B Macro when split detected**> and your reasonably modern ICOM radio will duly pass its VFO B frequency and mode to Logger32 whenever you put the radio in split.

Configure a button on the **R**adio **C**ontrol **P**anel with the $**IcomVFOB**$ macro to poll your ICOM for its VFO B settings at will.

For automatic operation, include the $**IcomVFOB**$ macro in the split macros in the RCP. See Setup DX spot macros for specific details on configuring and using the macro.

48.5 ICOM clock setting

Assuming your PC has a reasonably accurate clock (ideally being synchronized to an atomic reference clock), you can send CAT commands to *some* ICOM radios[463] through the **R**adio **C**ontrol **P**anel to reset their built-in clocks to match your PC's clock using these macros:

- $**SetIcomClockUTC**$ Calculates and sends the *UTC* time from the PC to the radio
- $**SetIcomDateUTC**$ Calculates and sends the *UTC* date from the PC to the radio
- $**SetIcomUTCOffset**$ Sets the hours' offset between local/clock time and UTC
- $**SetIcomClock**$ Sends the *local/clock* PC time to the radio
- $**SetIcomDate**$ Sends the *local/clock* PC date to the radio

[463] We're not sure, yet, which ICOM radios accept CAT commands to set the clock: by all means check your radio manual and try out the RCP macros, then tell us how you got on through the Logger32 reflector.

Copyright © 2026 Robert C. Furzer K4CY

48.6 ICOM CW and SSB bandwidth settings

If you have CW and SSB narrow filters installed, you can tell Logger32 which filters to select when you click DX spots, depending on their mode (as determined by Logger32 dutifully looking up the spot frequency in your Bands & Modes table).

In the radio port setup, select <**Use Narrow Filters for CW**> and edit *C:\Logger32\Logger32.INI*:

1. Close Logger32.
2. Make a copy of *C:\Logger32\Logger32.INI* and put it somewhere safe just in case something goes wrong.
3. Open *Logger32.INI* in a plain ASCII editor such as MS Notepad or WordPad, or TED Notepad.
4. Scroll down until you find a heading [Radio Port] or [Radio Port 2]. There should be 8 or so lines of text looking something like this:

   [Radio Port]

   Automatic Open=TRUE

   Comport=COM5,9600,n,8,2

   Radio Type= ICOM (not IC-735)

   Poll Interval=50

   ICOM Address=58

   Set DTR High=0

   Set RTS High=1

   CW Filter=1

5. The software defaults to filter 01 for CW normal and filter 02 for CW narrow. To use filters other than the defaults, add this section:

   [ICOM Filter 1|2] … where the number corresponds to *either* Radio 1 *or* 2

   CW Normal=xx … where xx equals the filter number 01, 02 or 03

   CW Narrow=xx … where xx equals the filter number 01, 02 or 03

   SSB Filter=xx … where xx equals the filter number 01, 02 or 03 (only for those ICOMs that support more than one SSB filter setting).

   The following chart, from the CI-V specification, shows the relationship of the numbers to filter width:

   Control Data IF Width

   01 Width 1 (Wide)

   02 Width 2 (Narrow)

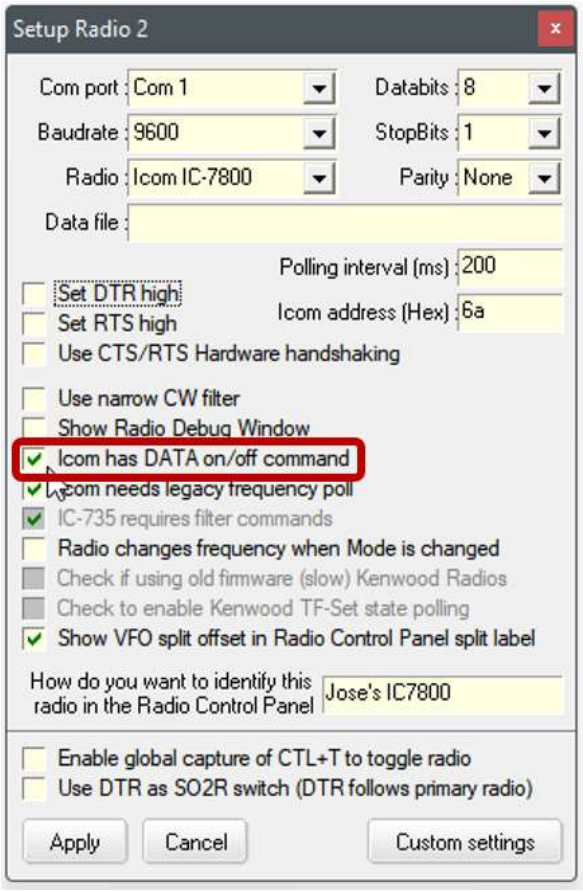
   03 Width 3 (Narrower)

   If you have more than one CW filter, consult your radio manual and experiment with the filter numbers.

Copyright © 2026 Robert C. Furzer K4CY

48.7 ICOM radio "DATA" modes

Some ICOM radios (recent ones, presumably) have a DATA setting. In DATA mode, the radio selects the AUX connector on the rear of the radio for audio input and output, muting the microphone and setting a fixed audio output level.



If your model of ICOM supports DATA, Logger32 can generate the CI-V CAT commands if you select <**Icom has DATA on/off command**> using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

In the DTR/RTS setting, you may need to select either <**Set DTR high**> or <**Set RTS high**>, depending on your radio and CAT interface.

48.8 ICOM radio settings for Logger32

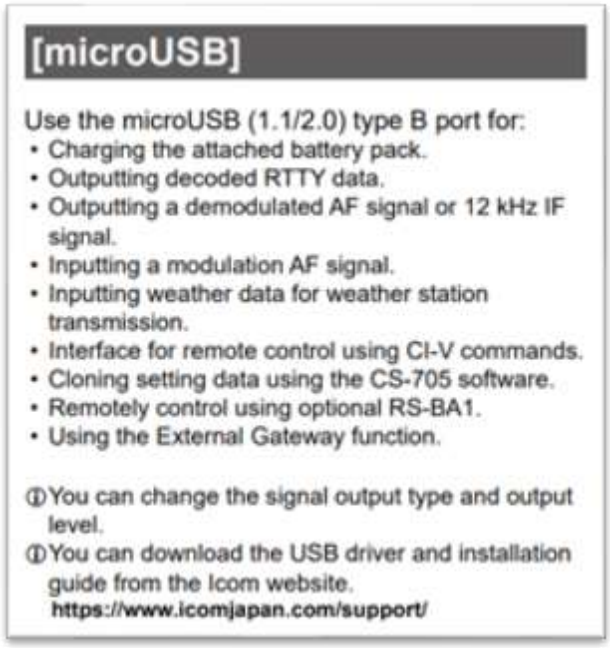
You need to set several important items in the ICOM radio menu to communicate with Logger32.

- **CI-V address**: the default CI-V addresses for several ICOM radios are listed above. Although you may be able to change the CI-V hexadecimal address in the radio menu, it is recommended to stick with the default. **You *must* use the actual CI-V hex address of your ICOM in the Setup radio form** in order for Logger32 to communicate with the radio via CAT.
- **CI-V Transceive:** *except* for the IC-735, ICOMs that support CI-V Transceive should have it *disabled* in the radio menu to reduce data collisions caused by unsolicited messages from the radio when you change the VFO frequency. However, if your linear amplifier needs CI-V control from the radio, the CI-V Transceive function should remain *enabled* despite the possibility of data collisions. You'll just have to live with it the occasional delay as data gets re-sent.
- **CI-V [USB] echo back:** this *must* be **on**, otherwise Logger32 cannot communicate bidirectionally with the radio for CAT. You may be able to QSY the VFO from Logger32, but if you turn the radio's VFO knob or hit the band change buttons on the front panel, Logger32 may not know anything has changed.
- **Radio model:** in **Setup ⇨ Radio**, Logger 32 generates the appropriate CAT commands for those ICOM models listed. If your radio is not listed and is a few years old, try the generic <**ICOM**> first. Recent ICOMs typically work with <**IC-7800**>. You may need to experiment with other radio models and some CAT commands may be unavailable or inoperable. So long as the most important commands (such as reading and setting the frequency and mode) are working, you will at least have partial CAT control of your ICOM.

Copyright © 2026 Robert C. Furzer K4CY

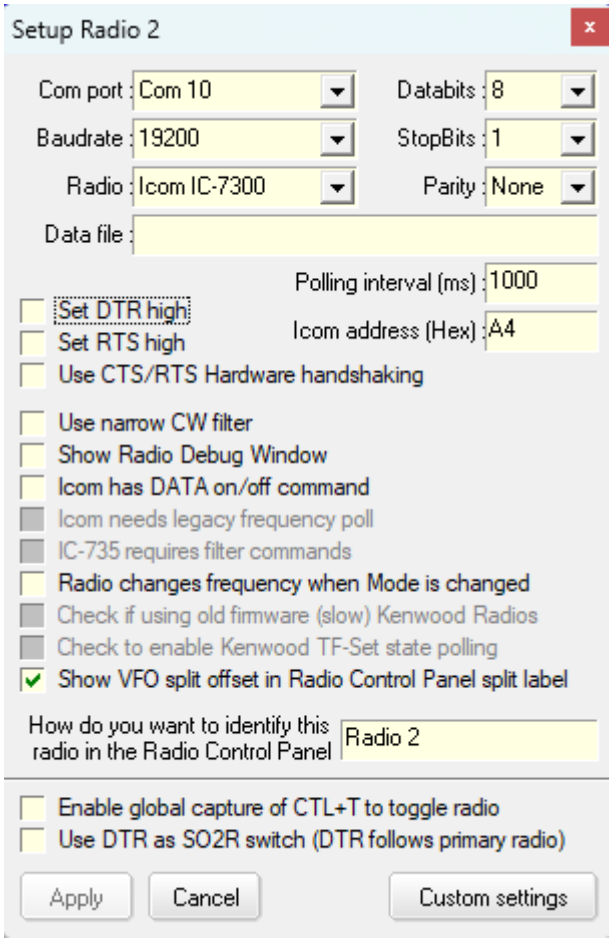
48.9 ICOM IC-705

According to the basic IC-705 manual, the radio's CAT connection requires a cable with a **microUSB type B** connector at the radio end, and you'll need the USB driver from ICOM Japan ▶



The type of USB connector at the *other* end of the CAT cable should match a free USB port on your computer/device ▼



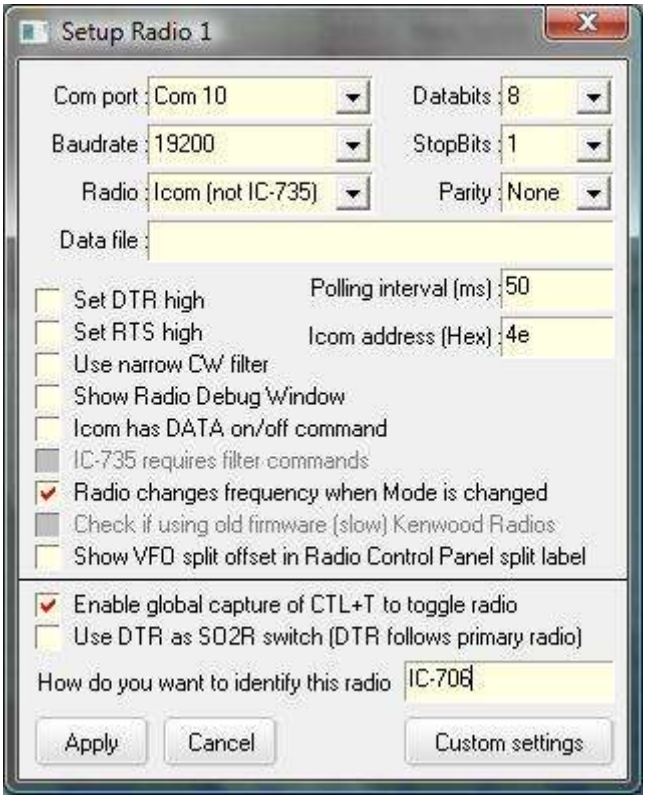


◀ Panos SV1GRN reports that this radio setup works with his IC-705.

Hinson tip: try reducing the polling interval for quicker, more responsive frequency and mode updates, Panos. 1,000 ms (1 second) looks slow to me.

Copyright © 2026 Robert C. Furzer K4CY

48.10 ICOM IC-706 Mk II



Some of the IC-706 MkII radio factory default settings should be changed.

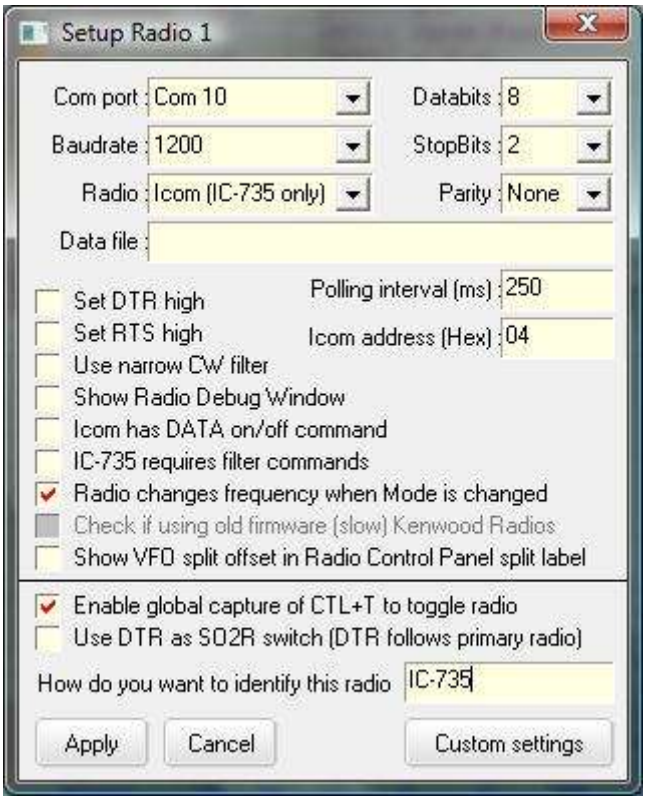
- **CI-V Hex address** = 4EH (menu item № 25) Leave as-is.
- **CI-V Baud rate** = Auto (menu item № 26) Change to 19200.
- **CI-V Transceive** = On. (menu item № 27) Change to OFF.

Use **Setup ⇨ Radio ⇨ Radio 1|2 configuration** to configure the CAT connection. The COM port should be set to suit that which is actually being used ▶

It has been found that <**StopBits**> can be either 1 or 2 with little difference to performance.

Commercial interface circuits typically draw power from the computer RTS or DTR lines. Tell Logger32 to set the DTR and/or RTS lines high by clicking to select (tick) the setup options. If you do so, you will not be able to share the same COM port for CW operation. However, if your interface draws power from the DTR line, you may still be able to share the RTS line for PTT operation of the Sound card data window.

48.11 ICOM IC-735



The IC-735 radio comes with the following factory default settings:

- **Hex address** = 04
- **Baud rate** = 1200
- **Transceive function** = Enabled.

Whereas other ICOM transceivers communicate using 5 byte parcels of data, the IC-735 alone uses just 4 bytes. This is why the IC-735 in a category by itself when setting up the CI-V communications with a computer. Logger32 gives you the choice of "ICOM (IC-735 only)" or "ICOM (not IC-735)."

Use **Setup ⇨ Radio ⇨ Radio 1|2 configuration** to configure the CAT ▶

Although the default settings work fine with Logger32, it is suggested that you use 2 stop bits. It seems to work even better.

Copyright © 2026 Robert C. Furzer K4CY

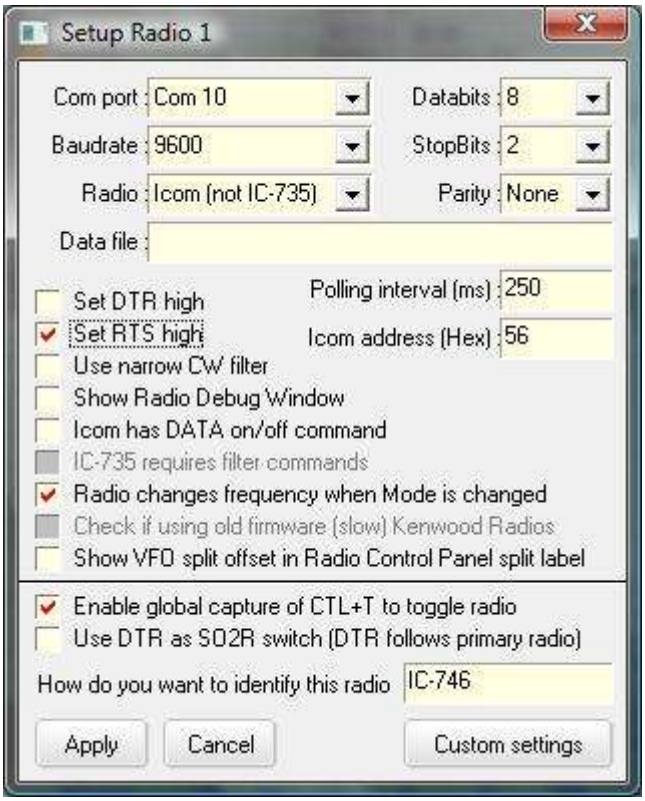
If desired for faster performance, the baud rate can be increased to 9600 by moving jumper “D4” to position “D5” on J-22 inside the radio under the PA unit. To gain access you need to remove the top cover and the 4 screws attaching the PA unit. You can set aside the PA unit by disconnecting the power cable to the PA unit.

In contrast to other ICOM radios, the IC-735 will not communicate properly with Logger32 if you disable the **Transceive** function. It must be left ON which is the factory default.

Do not set the polling Interval below 250 ms as the IC-735 cannot respond fast enough.

Commercial interface circuits typically draw power from the computer RTS or DTR lines. Tell Logger32 to set the DTR and/or RTS lines high by clicking to select (tick) the setup options. If you do so, you will not be able to share the same COM port for CW operation. However, if your interface draws power from the DTR line, you may still be able to share the RTS line for PTT operation of the Sound card data window.

48.12 ICOM IC-746 with OPC-662 interface



Using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** set the COM port and baudrate to match the radio’s CAT connection ▶

Set the radio to <**Icom (not IC-735)**>.

Enable <**Set RTS high**> to power the OPC 662 interface - part of the ICOM RS-746 Package.

<**Icom address (Hex)**> default for the IC-746 is 56.

The default values of 9600 baud and 250ms polling are starting points. If your radio and computer can handle faster rates, you may prefer the highest rate that provides reliable operation with your specific hardware. Baud rates make little difference but reducing the value of the polling speed (increasing the rate of radio polling) will provide a faster and smoother display of all band, frequency and mode changes.

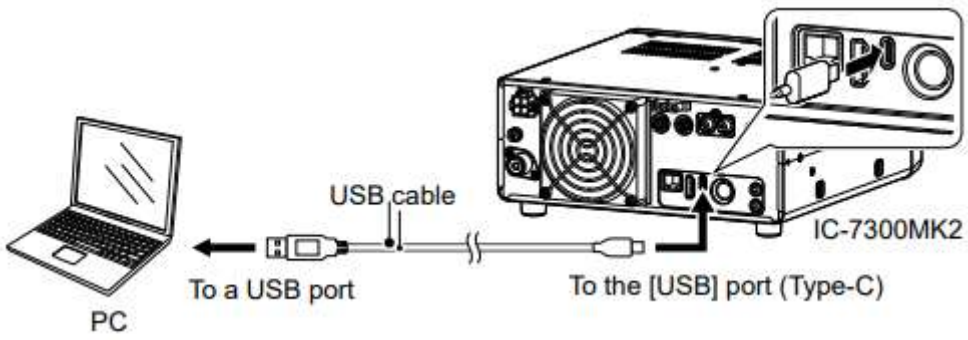
Disable CI-V Transceive to reduce data collisions caused by unsolicited messages from the radio every time you change the VFO frequency.

If you have CW and SSB narrow filters installed, you can tell Logger32 which filter to use by editing *C:\Logger32\Logger32.INI* to automatically set the desired bandpass filter for DX spots according to the mode.

48.13 ICOM IC-7300

The IC-7300 has both CI-V and USB interfaces for CAT. A built-in audio CODEC is available through the USB interface. **This section of the Logger32 manual explains how to use the USB interface.**

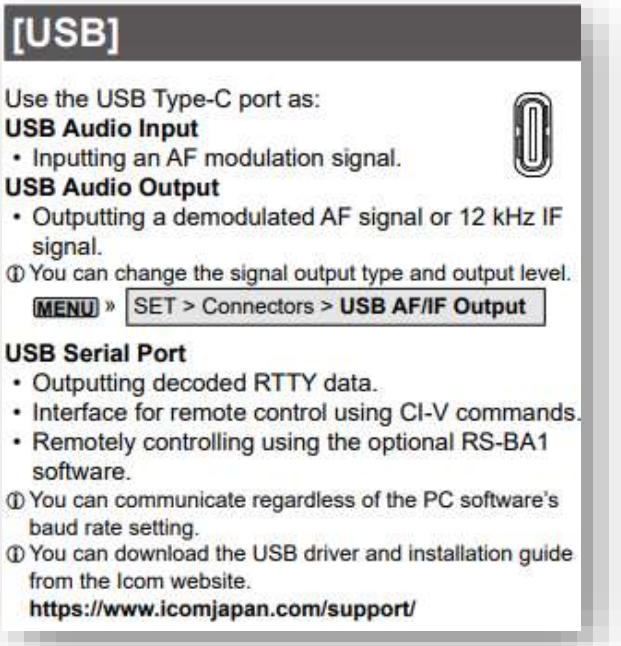
Copyright © 2026 Robert C. Furzer K4CY



Hinson tip: Dave N8DC says that, for RTTY, PSK and other AFSK modes with the IC-7300, you need a CAT cable connected to the CI-V remote jack on the rear panel *in addition to* the USB cable used for the audio.

Start by installing the correct driver and connecting a USB cable between a PC USB socket and the IC-7300 USB socket, referring to the ICOM IC-7300 manual ▶

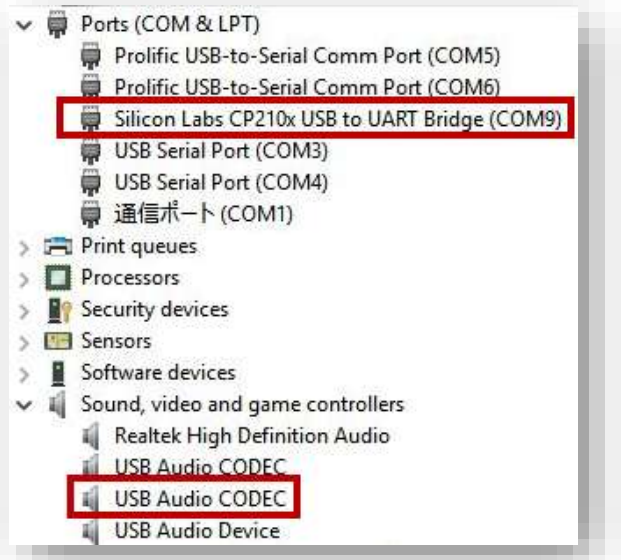




▲ To find out the IC-7300's USB port number, temporarily unplug the USB cable and, in Windows **Control Panel ⇨ Device Manager** look at the **Ports (COM & LPT)** and **Sound, video and game controllers** sections showing all devices *except* for the unplugged IC-7300.

Now plug in the IC-7300 USB cable.

The port and sound device for the IC-7300 appear ▶



Copyright © 2026 Robert C. Furzer K4CY

In that example, the Silicon Labs CP210x USB to UART Bridge (COM9) provides the CAT connection, and one of the two USB Audio CODECs is the audio interface [464].

Setup IC-7300 menu items: through the radio menu, select **SET** ⇨ **Connectors** and configure the following connector-related settings first:

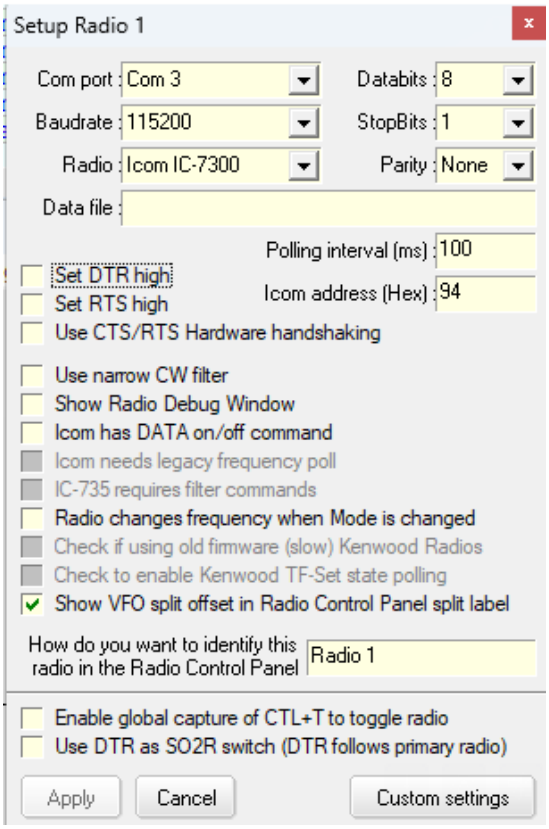
- **ACC/USB Output Select**: AF
- **ACC/USB AF Output Level**: 20%
- **ACC/USB AF SQL**: OFF
- **ACC/USB AF Beep/Speech Output**: OFF
- **USB MOD level**: 50%
- **USB KEYING**: (CW) USB (A) RTS
- **DATA OFF MOD**: MIC
- **DATA MOD**: USB

Now select **SET** ⇨ **Connectors** ⇨ **CI-V** and set these:

- **CI-V address**: 94h
- **CI-V Transceive**: OFF*
- **CI-V USB->Remote Transceive address**: 94
- **CI-V Output (for ANT)**: OFF*
- **CI-V USB Port**: Unlink from [Remote]
- **CI-V USB Baud Rate**: 115200
- **CI-V USB Echo Back**: ON – this is necessary for the CAT connection to operate correctly in both directions, passing data to *and* from the radio.

* *Unless* you have other devices on the shack LAN such as amplifiers or antenna switches that need the frequency information from the radio.

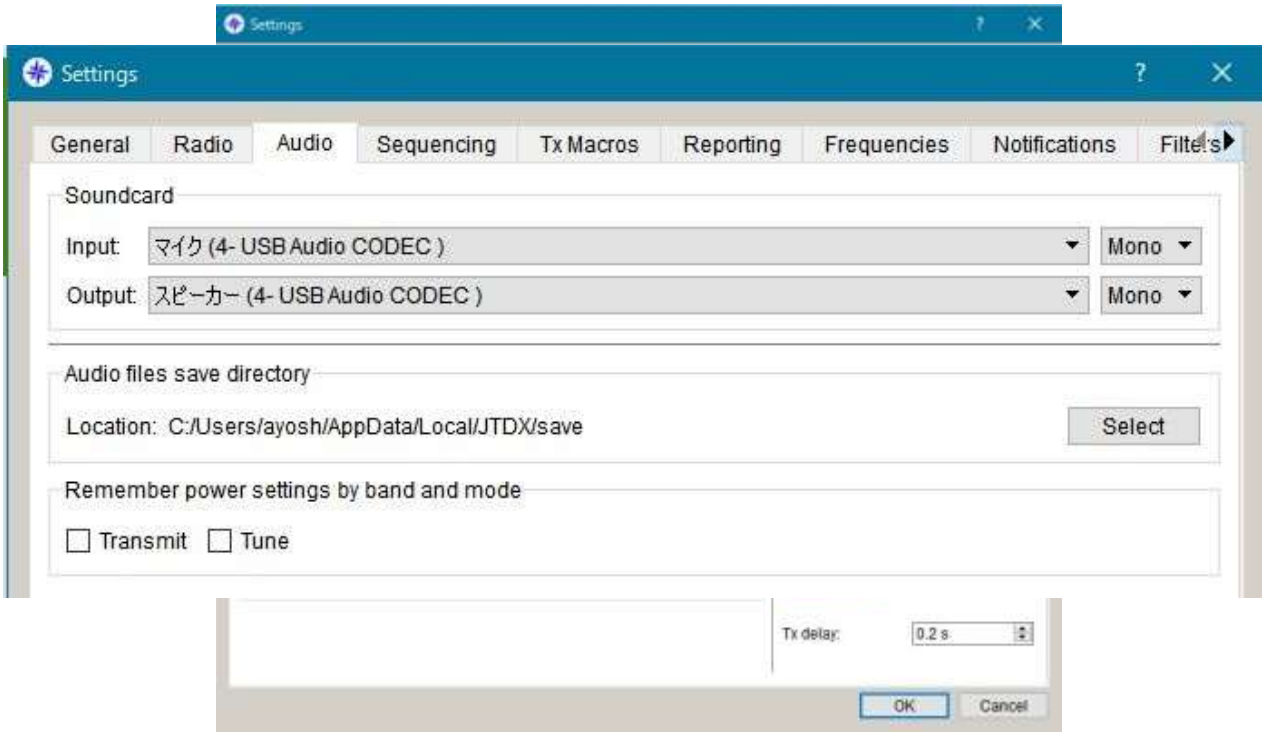
In Logger32, open **Setup** ⇨ **Radio** ⇨ **Radio 1|2 setup** and use these settings but with *your* COM port number and identification for the RCP ▶



[464] Without additional info, we don't know which is which … but at least now we know the CODEC is active!

Copyright © 2026 Robert C. Furzer K4CY

For FT# operation using the IC-7300, JTDX works flawlessly using the radio's built-in CODEC for the audio and the CI-V control cable for CAT. Carefully configure each item, especially the radio type (which governs the CAT commands used), the COM port number and baudrate ▼



48.13.1 IC-7300 CAT commands

See the IC-7300 CI-V reference guide for a comprehensive, detailed list. This is just a small selection, a few common CAT commands:

- $**HexCommand FE FE 94 E0 0F 01 FD**$ Split on
- $**HexCommand FE FE 94 E0 0F 00 FD**$ Split off
- $**HexCommand FE FE 94 E0 07 00 FD**$ VFO A
- $**HexCommand FE FE 94 E0 07 01 FD**$ VFO B
- $**HexCommand FE FE 94 E0 07 A0 FD**$ A=B
- $**HexCommand FE FE 94 E0 07 A0 FD**$ A/B
- $**HexCommand FE FE 94 E0 06 01 FD**$ USB
- $**HexCommand FE FE 94 E0 06 00 FD**$ LSB
- $**HexCommand FE FE 94 E0 06 02 FD**$ AM
- $**HexCommand FE FE 94 E0 06 04 FD**$ RTTY
- $**HexCommand FE FE 94 E0 06 03 FD**$ CW
- $**HexCommand FE FE 94 E0 06 05 FD**$ FM

- $**HexCommand FE FE 94 E0 06 01 FD**$ USB with next line below
- $**HexCommand FE FE 94 E0 1A 06 01 FD**$ USB D

- $**HexCommand FE FE 94 E0 16 41 01 FD**$ VFO A notch filter on
- $**HexCommand FE FE 94 E0 16 41 00 FD**$ VFO A notch filter off

Copyright © 2026 Robert C. Furzer K4CY

- $**HexCommand FE FE 94 E0 16 22 01 FD**$ **N**oise **B**lanker on
- $**HexCommand FE FE 94 E0 16 22 00 FD**$ **N**oise **B**lanker off
- $**HexCommand FE FE 94 E0 16 40 01 FD**$ Noise **R**eduction on
- $**HexCommand FE FE 94 E0 16 40 00 FD**$ **N**oise **R**eduction off

- $**HexCommand FE FE 94 E0 16 45 01 FD**$ Monitor on
- $**HexCommand FE FE 94 E0 16 45 00 FD**$ Monitor off

- $**HexCommand FE FE 94 E0 14 01 00 00 FD**$ Volume level 0 (mute)
- $**HexCommand FE FE 94 E0 14 01 01 40 FD**$ Volume level 40
- $**HexCommand FE FE 94 E0 14 01 01 60 FD**$ Volume level 60

- $**HexCommand FE FE 94 E0 14 0A 00 52 FD**$ Power output 20 watts
- $**HexCommand FE FE 94 E0 14 0A 01 30 FD**$ Power output 50 watts
- $**HexCommand FE FE 94 E0 14 0A 02 55 FD**$ Power output 100 watts

- $**HexCommand FE FE FE 94 E0 18 01 FD**$ Switch on the radio
- $**HexCommand FE FE FE 94 E0 18 00 FD**$ Switch off the radio

48.13.2 Example IC-7300 macros

- $**QSY 14195**$ $**HexCommand FE FE 7A E0 06 01 FD**$ QSY to 14195 kHz
- $**QSY 3930**$ $**HexCommand FE FE 94 E0 06 00 FD**$ QSY to 3930 kHz

Right-click the righthand side of the RCP, then enter these commands:

Top window: macro to clear split

- $**HexCommand FE FE 94 E0 0F 00 FD**$
- $**HexCommand FE FE 94 E0 07 D0 FD**$
- $**HexCommand FE FE 94 E0 0F 00 FD**$

Center window: macro to apply split information from a DX spot

- $**HexCommand FE FE 94 E0 07 A0 FD**$ B=A
- $**HexCommand FE FE 94 E0 07 00 FD**$ VFO A
- $**HexCommand FE FE 94 E0 0F 01 FD**$ Split
- $**HexCommand FE FE 94 E0 05 #DXSpotSplit# FD**$ Use split info from a DX spot
- $**HexCommand FE FE 94 E0 07 B0 FD**$ Swap A/B

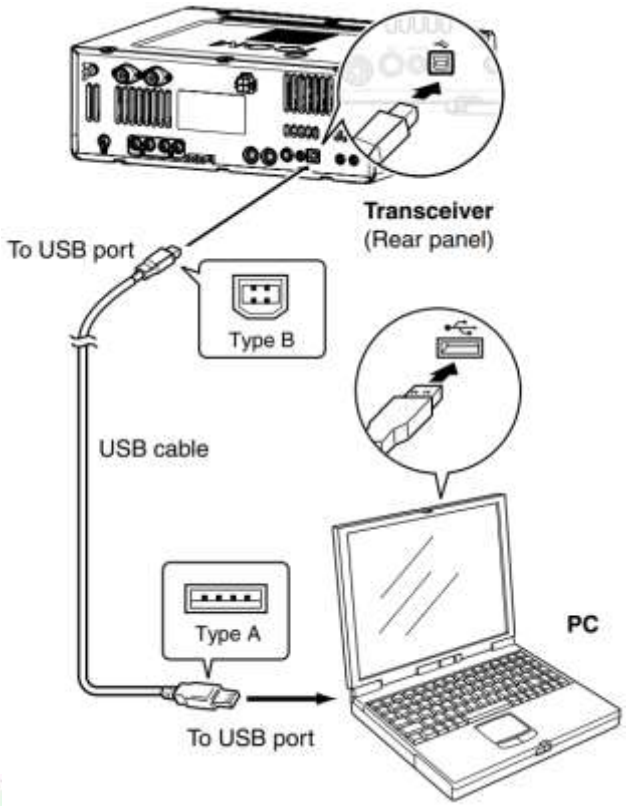
Bottom window

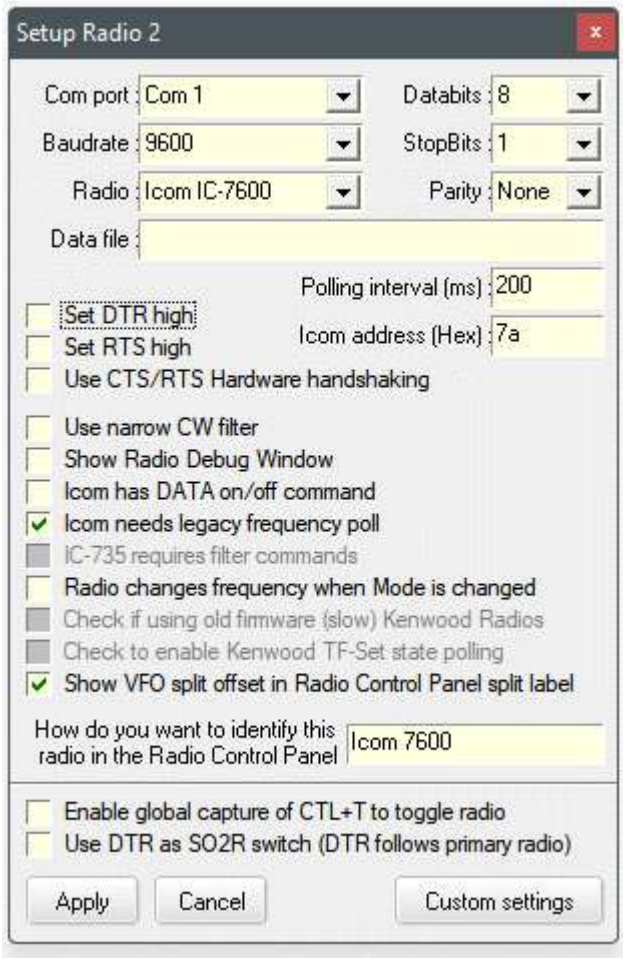
Leave blank

Copyright © 2026 Robert C. Furzer K4CY

48.14 ICOM IC-7600

1. Download and install ICOM's USB driver onto your PC from here, following the USB driver installation guide.
2. Connect your PC to the 7600 *rear* panel USB port using a USB type A to type B cable ▶
3. Turn on the radio. The radio's USB port should spring to life, and it should now appear in Windows **Control Panel ⇨ Device Manager** as a "CP210x USB to UART bridge controller (COM [n])" where [n] is the COM port number it is using.
4. Check the radio's menu settings for the USB port *e.g.* CI-V baudrate 9600, USB serial function CI-V ... [465]
5. In Logger32, configure the CAT settings to match the radio using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▼**





Be sure to set the correct radio type, baudrate, Icom address (Hex) **7a**, and the COM port number from step 3. Give it a suitable name to identify it in the **R**adio **C**ontrol **P**anel and elsewhere.

6. Check that Logger32 has CAT control of the radio *e.g.* "Radio 1|2" on the lower status line should now be blue. The VFO frequency should appear on the log entry pane. If you right-click the blue "Radio 1|2" box and click to open the radio debug window, you should see CAT commands and responses flowing in each direction between PC and radio. Turn the radio VFO knob or pick a new band on the radio, and the frequency in the log entry pane should show the new value.

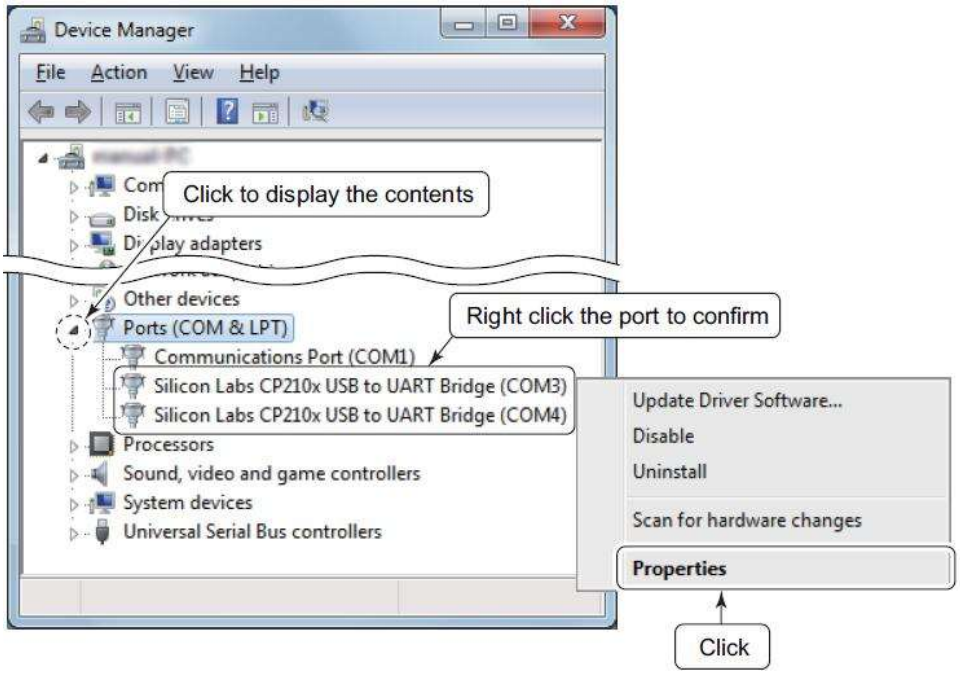
7. Make QSOs, log them, click interesting DX spots, have fun, experiment!

[465] I don't have an IC-7600. These instructions are based on my understanding of the ICOM documentation.

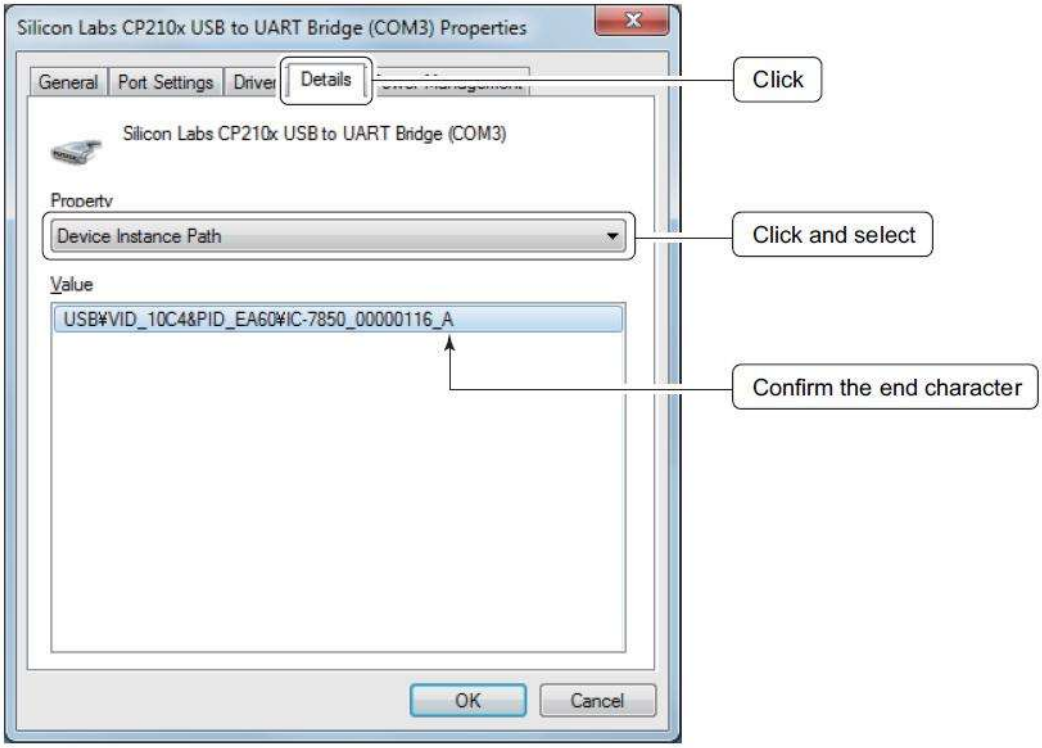
Copyright © 2026 Robert C. Furzer K4CY

48.15 ICOM IC-7610

Install the drivers and connect the radio. Of the *pair* of newly created com ports, determine which is which using Windows **Control Panel ⇨ Device Manager ▼**



Click to open the <**Details**> tab ▼



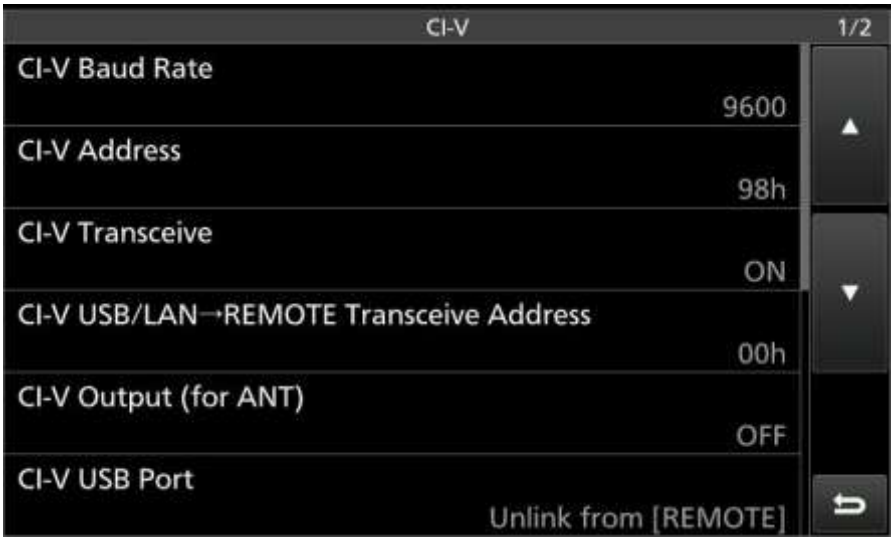
Click <**Property**> and select <**Device Instance Path**> from the drop-down list.

Check the information displayed in "Value."

In my set up the **A** port (COM3 in the above example) is the CAT port, while the **B** port (COM4 above) is used in the radio's USB Send/Keying menu for Digital Modes and CW keying, PTT *etc*.

Copyright © 2026 Robert C. Furzer K4CY

My radio and Logger32 CAT settings ▼



CI-V USB Echo Back *must* be ON for Logger32 to track frequency/band changes, and a fixed CI-V USB Baud Rate (not auto) makes it simple to match Logger32's serial port speed setting ▼



My radio menu for USB Send/Keying ▼

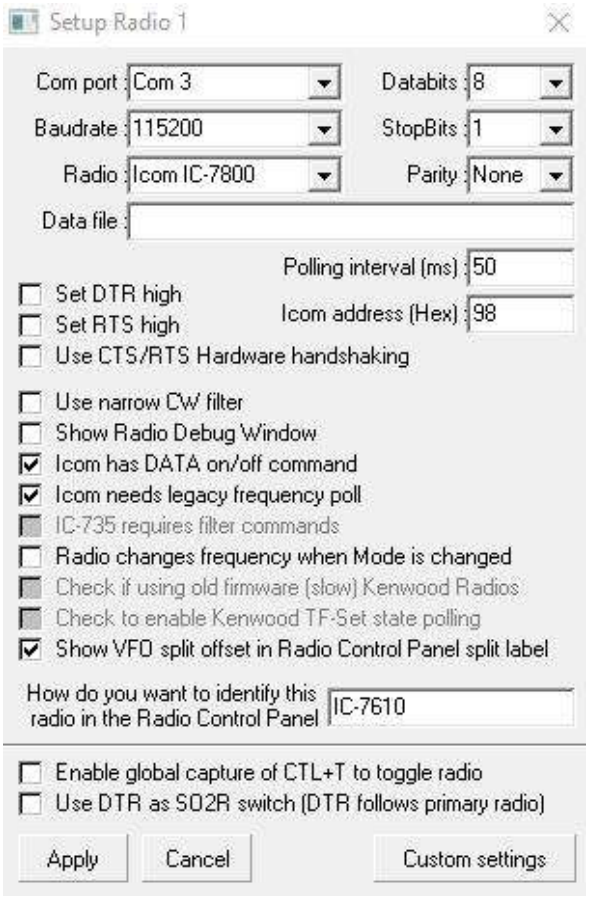


Copyright © 2026 Robert C. Furzer K4CY

48.15.1 Setup IC-7610 in Logger32

When connected to the radio, Windows Device Manager shows a *pair* of serial ports (Silicon Labs CP210x USB-to-UART bridge).

The lower-numbered COM port (which identifies as the "A" port) is used for CAT commands, using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**



The higher-numbered COM port (the "B" port) is used for digital mode/CW software for PTT, CW and FSK keying.

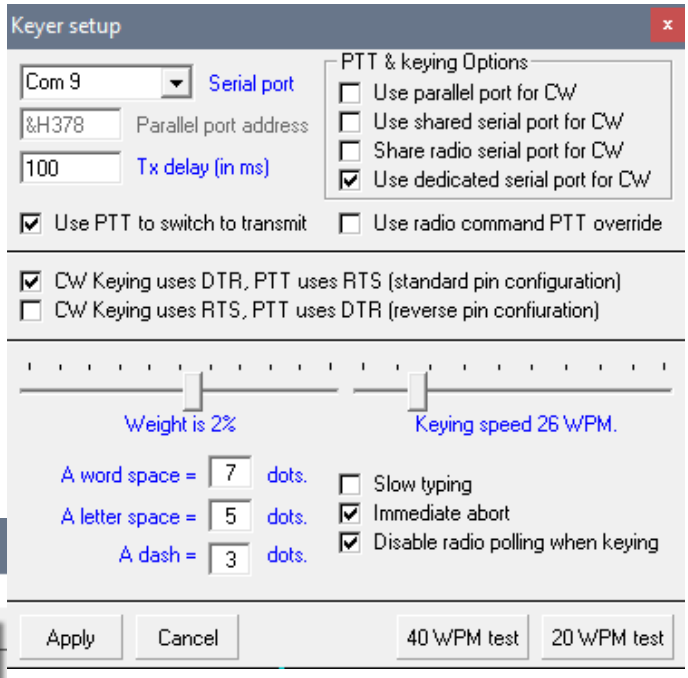
Using the radio's USB SEND/KEYING menu:

- USB SEND USB1(B)RTS
- USB Keying (CW) USB1(B)DTR
- USB keying (RTTY) USB1(B)DTR

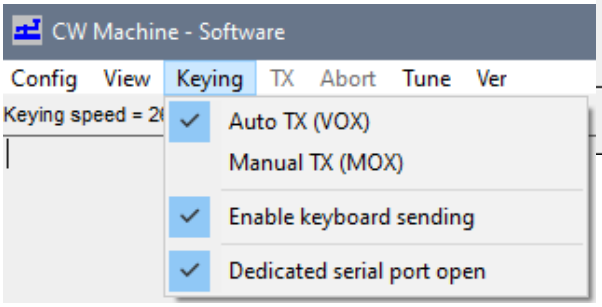


◀ In the CW Machine <**Config**> menu, click to tick **Software** keying.

Then under **Config ⇨ Keyer setup ▼**



Finally, in the CW Machine <**Keying**> menu ▼



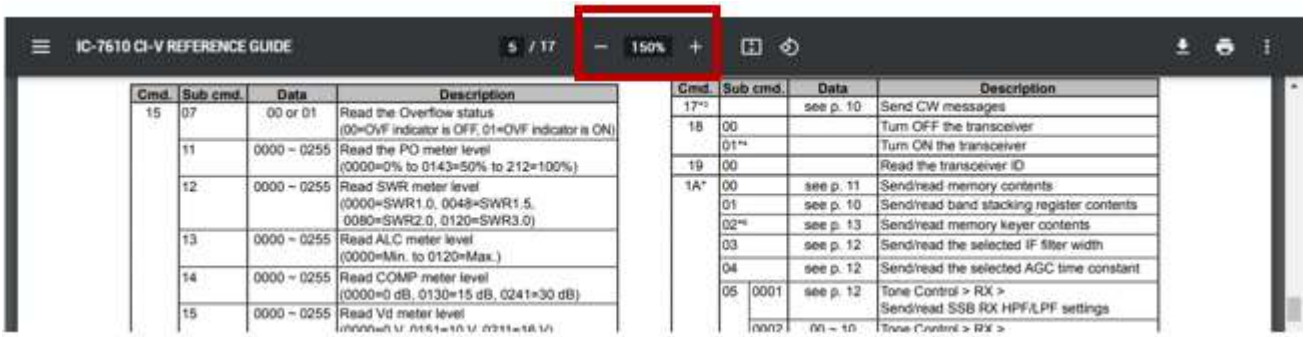
Copyright © 2026 Robert C. Furzer K4CY

Hinson tip: John G7SSE reports that configuring his IC-7610 as an IC-7300 in Logger32 lets Logger32 control the rig's DVK. Other functions operate normally.



◀ If you find the split indicator is always flashing on the log entry pane and stealing the focus, enable <**Read VFO-B Macro only when split on/off detected**> on the form that opens when you click the <**Custom settings**> button at the bottom of the radio configuration form.

Hinson tip: the IC-7610 CI-V Reference Guide describes the rig's CAT commands in fine detail … and fine print. Click the Acrobat reader's *plus* button to magnify the document so the page fills the full width of your screen ▼



Experiment with the commands and parameters through the Radio Control Panel.

Copyright © 2026 Robert C. Furzer K4CY

48.16 ICOM IC-9700

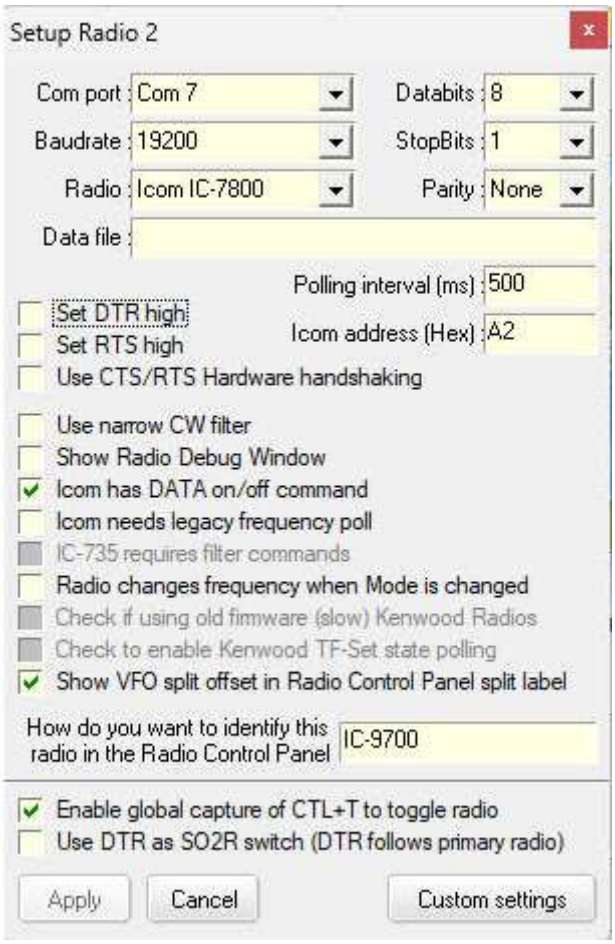
Although Logger32 does not directly support the *entire* IC-9700 CAT instruction set with all its fancy satellite and other functions, the **IC-7800** radio type provides the basic CAT commands at least.

First, download and install the USB driver from the ICOM website, then connect the radio to your PC using a USB A/B lead.

Next, using the radio's built-in menu, select the following settings:

- CI-V Baud 19200
- CI-V Address A2h
- CI-V Transceive OFF
- USB/LAN 00h
- CI-V USB Port Unlink from (Remote)
- CI-V DATA Baud rate 19200
- **CI-V Echo Back ON**
- CI-V DATA Baud Rate 19200
- CI-V DATA Echo Back OFF

Finally, in Logger32, using **Setup ⇨ Radio ⇨ Radio 1|2 setup**, these port settings reportedly do the trick ▶



Hinson tip: if you are keen to try out the extended CAT commands available on the IC-9700, you can compose and send additional CAT command sequences from Logger32's **Radio Control Panel**. Study the IC-9700 CI-V Reference Guide for details of the CAT commands that work with this model.

Copyright © 2026 Robert C. Furzer K4CY

49 Kenwood

> "People are moved by the human voice. So we've developed the fine radio communication technology. Voices continue to shape the world, and KENWOOD continues to shape the future of communication."
>
> *Kenwood*

49.1 Supported Kenwood radios

TS-50	TS-480	TS-840	TS-940	TS-2000
TS-60	TS-570	TS-870	TS-950	Others: use "Kenwood" (generic)
TS-440	TS-590	TS-890	TS-990S	

49.2 Kenwood CAT connections and commands

49.2.1 CAT connectors

There are several[466] types of CAT connections for Kenwood amateur radios:

- **Type A**: with the IC-10 kit installed in radio plus an external IF-232C interface box (or equivalent[467]). A 6-pin DIN cable is required between the radio and the IF-232C, and a standard RS232 cable is connected between the IF-232C interface box and the PC's serial port. Example radios: TS-440, TS-850, TS-940.
- **Type B**: the radio has an RS232 serial port. A standard RS232 cable is connected between the radio's RS232 port and the PC's serial port. Example radios: TS-480, TS-570.

[466] There may also be USB-only models in the range, plus other kind of port for specific applications or customers. Kenwood is a major commercial radio manufacturer. Amateur equipment is only part of their extensive range.

[467] The MAX-232 level-converter chip worked nicely in *my* homebrewed TS850 CAT interface.

Copyright © 2026 Robert C. Furzer K4CY

- **Type C**: the radio has both RS232 *and* USB ports - use either of them. The correct USB driver must be installed. Example radios: TS-590S, TS-590SG, TS-990S (which *also* has an Ethernet LAN port for TCP/IP comms).

For Type A and Type B radios, you would need a working USB-RS232 serial adapter if your PC only has USB ports. Pick an adapter with the FTDI chipset that is certified to work with Windows 10 or 11, if possible, as it is more likely (but still not guaranteed!) to work than one with potentially fake Prolific chips.

49.2.2 Kenwood CAT command format

The CAT commands to set various Kenwood [468] functions are composed of: two letters + parameters + a semicolon (;) at the end, without any embedded spaces. For example, this command sets VFO A to 7000.000 kHz (7 MHz exactly):

FA00007000000;

where **FA** is the command to set the VFO A frequency, the numbers are the frequency parameter, and the semicolon terminates the command.

Some of the most useful commands for radio control in connection with Logger32 are filter selection and slope tuning or filter bandwidth settings. Naturally, there are many combinations, so you are left to experiment with the settings and figure out your own preferences …

49.2.3 Basic command examples

Request **IF;**

Reply **Ifxxx;** where **xxx** is 35 bytes giving the radio's frequency, mode *etc*.

Request **FA;**

Reply **FAxxx;** where **xxx** is 11 bytes giving the radio's VFO A frequency.

Request **MD;**

Reply **MDx;** where **x** is 1 byte for the radio's mode: 1=LSB, 2=USB, 3=CW *etc*.

49.2.4 Codes for filter selection for the TS-850

The Kenwood TS-850 command for this function is **FLaaabbb** where **FL** is the filter command, and **aaa** and **bbb** are the codes required for the 8.83 kHz and 455 kHz IF filters, respectively:

- 000 = No select
- 002 = FM wide
- 003 = FM Narrow
- 005 = AM
- 007 = SSB (2.7 kHz)

[468] The Kenwood protocol is also used by Elecraft and recent Yaesu radios. All CAT-enabled Kenwoods use the same CAT command *protocol* although the available *commands* vary a little across the model range. For details, check the CAT command programmer's reference manual for your specific model of radio.

Copyright © 2026 Robert C. Furzer K4CY

- 009 = CW (500 Hz)
- 010 = CW narrow (270 Hz)

So, the Logger32 $**Command**$ macro to select both 2.7 kHz filters would be:

$**Command FL007007;**$

For 500 Hz + 500 Hz the macro would be:

$**Command FL009009;**$

Macros for other filter combinations follow the same pattern.

49.2.5 Codes for filter passband (slope tuning) on the TS-850 and TS-570

The Kenwood TS-850 command for this function is **SH*aa*;** assuming you are using USB, where SH is the command for **S**lope **H**igh, and ***aa*** is the code for the tuning position, in the range 00 to 20.

This is only really effective if the 500 or 270 Hz filters are selected.

The following codes set the nominal filter center frequency:

- 00 = 1450 Hz
- 01 = 1300 Hz
- 02 = 1100 Hz
- 03 = 950 Hz
- 04 = 800 Hz
- 05 = 600 Hz.

So to set a passband centered on 950 Hz, use:

$**Command SH03;**$

On the TS-570:

$**Command SH19;SL19;**$

sets a 200 Hz bandpass centered on 1 kHz, and:

$**Command SH00;SL00;**$

sets the default (open) bandwidth.

49.2.6 VFO/memory channel selection for all Kenwoods

The CAT commands to set the VFO or to select a memory channel are FRx (for receive) and FTx (for transmit). **FRx** is the receive VFO selection and **FTx** is the transmit VFO selection, where **x** is:

- **0** for VFO A.
- **1** for VFO B.
- **2** for memory.

They can be conCATenated as $**Command FRx;FTx;**$ *with* their semicolons. So, to set receive on VFO A and transmit on VFO B (the conventional split arrangement), use:

$**Command FR1;FT2;**$

Copyright © 2026 Robert C. Furzer K4CY

Be careful when using this command in MMTTY and MMVARI. The right-click function to move the receive/transmit frequency to the preferred audio frequency on the waterfall works *only* with VFO A. $**Command FR0;**$ at the start of any $**Align**$ macro selects VFO A.

49.2.7 Specific memory channel selection

The Kenwood TS-850 CAT command for memory channel selection is MC_aa, where MC is the command and aa is the channel number. A space or underscore is required between MC and the parameter. The command only takes effect if the memory channel has first been activated *i.e*. you must set **FRx** or **FTx** to 2, so that there is something for the selected memory channel to do.

Here is the command to select memory channel 05 for transmit and receive:

$**Command FT2;FR2;MC_05;**$

49.2.8 TF-SET control

You can use $**TF-Set**$ and $**MouseTF-Set**$ macros in the **R**adio **C**ontrol **P**anel, CW Machine and Sound card data window for some Kenwood radios. If the $**MouseTF-Set**$ macro is defined to a function key, it works just like the TF-SET button on the radio's front panel: press and hold to listen on your transmit VFO frequency. To make this work, Logger32 constantly polls the radio to determine the TF key status.

49.2.9 Combining codes in a macro

Multiple radio commands (*e.g.* for filter selection and slope tuning) can be combined to perform a complex procedure with one compound command. Here is an example for a TS-850 to select a 500 Hz filter for the 8.83 kHz IF with a passband centered on 950 Hz:

$**Command FL007009;SH03;**$

This macro selects the 270 Hz filter for the 8.83 kHz IF with a passband centered on 950 Hz:

$**Command FL010007;SH03;**$

49.2.10 Cross-mode operating

Some amateurs prefer to receive in CW mode and to transmit in USB mode. This may be impracticable if you are pressing buttons on the radio but you can implement it easily using "Transmit" and "Receive" macro buttons to send the correct CAT commands instead of using the transmit/receive panel in MMVARI or MMTTY.

There is some preparation of your Kenwood radio for the following two commands:

- Set the menu item № 24 to 1000 on your radio.
- Set RIT to +1.00 kHz.
- Set the preset audio frequencies in Logger32 to 1000.

To receive in CW mode:

$**Command MD3;RD1;FW0005;**$

That macro does the following three things:

- Sets the radio to CW mode (**MD3**).
- Turns the RIT on (**RD1**).

Copyright © 2026 Robert C. Furzer K4CY

- Sets the filter to 50 Hz (**FW0005**).

Now follow or precede this sequence with a $**Receive**$ macro and you have defined the 'receive in CW mode' command.

Now for transmit in USB mode:

$**Command MD2;RT0;FW0030;IS+3400;**$

That macro:

- Sets the radio to USB mode (**MD2**).
- Turns the RIT off (**RT0**).
- Sets the bandpass filter low-side to 300 Hz (**FW0030**).
- Sets the bandpass filter high-side to 3400 Hz (**IS+3400**).

Add the $**transmit**$ macro to ground the radio's PTT.

Feel free to select other mode and filter combinations by adjusting the parameters to your preference.

Hinson tip: configuring separate **R**adio **C**ontrol **P**anel macro buttons makes it easier to experiment with different settings and refine the macros. I typically reserve a few buttons for macro development and testing, with a distinctive button color.

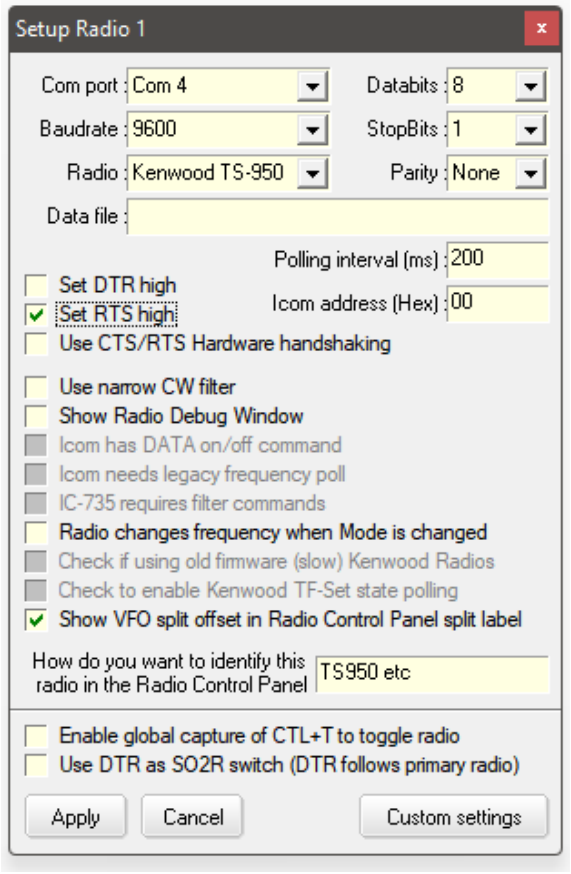
49.2.11 Kenwood clock setting

Assuming your PC has a reasonably accurate clock (ideally being synchronized to an atomic reference clock), you can send CAT commands to *some* Kenwood radios through the **R**adio **C**ontrol **P**anel to reset their built-in clocks to match your PC's clock using these macros:

- $**SetKenwoodClock**$ - sets the radio's clock to local time.
- $**SetKenwoodClockUTC**$ - sets the radio's clock to UTC.

Copyright © 2026 Robert C. Furzer K4CY

49.3 Kenwood TS-50, TS-60, TS-840, TS-940, TS-950SD and TS-950DX



It may be necessary to select <**Set RTS high**> in your configuration and you may need to hard-wire the computer's RTS to the 'modem' side's CTS and vice-versa (one end's **R**equest **T**o **S**end signal triggers the other end's **C**lear **T**o **S**end, and *vice versa*).

There are two alternatives:

Use a three-wire interface (TxD, RxD and Gnd) with appropriate level converters, and loop the CTS and RTS pins inside the radio plug …

- or -

… use a full interface with level converters for TxD, RxD, CTS, RTS and Gnd, such as the Kenwood IF-232, and click to select <**Set RTS high**> in Logger32 using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

49.4 Kenwood TS-440

The Kenwood TS-440S(/AT) was designed to allow for computer control by using a Kenwood IC-10 chip set, installed within the TS-440, and a Kenwood IF-232C external level converter box installed between the TS-440 and the RS232 port of a computer.

The IC-10 accessory kit consists of two ICs that are physically installed within the TS-440. You can either obtain an IC-10 kit from the dealer or buy inexpensive equivalent ICs (IC-54 is a uPD-8251-AC also known as 8251A; IC-55 is a CMOS TC-4040-BP also known as TC4040BP) and install them following the instructions in the TS-440 manual.

With the chips installed, the TS-440 has an asynchronous, ASCII, 4800BPS, 1-Start 8-Data 2-Stop bit TTL level (0 to +5V) interface at the ACC-1 6-pin DIN connector on the rear panel.

The radio uses CTS/RTS control: the IC-10 interface will only transmit data when it sees a TTL high signal (+5 volts) on the CTS (**C**lear **T**o **S**end) pin (pin 4 in the DIN connector). The interface sets the RTS (**R**equest **T**o **S**end) pin (pin 5 in the DIN connector) high when the radio is able to accept data.

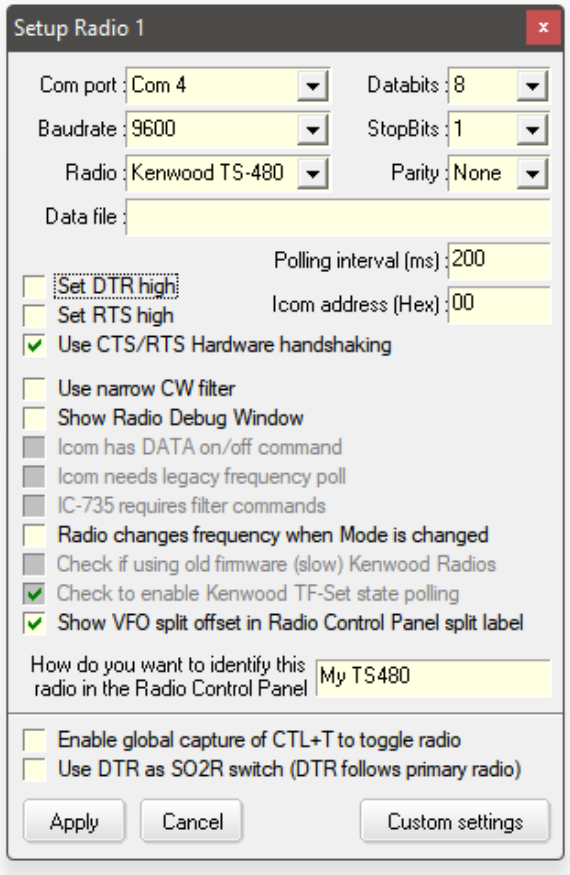
The Kenwood IF-232C external level converter box comprises opto-isolators and level converters to interface the TS-440's ACC-1 DIN connection's TTL voltages to the computer's serial port RS232C voltages.

The Kenwood IF-232C only senses the signal levels from the computer RS232 connector and the radio's TTL DIN connector. The IF-232C is powered from an external +13.8 volt power supply and does not use the signal lines for internal power. This voltage is filtered and is used to generate a regulated +5 volts for internal use and to provide the TTL signals (0.0v and 5.0v) back to the radio. The filtered 13.8 volt line is used to generate +11.5 and -11.5 volts for the true RS232 signal levels for the computer interface.

Copyright © 2026 Robert C. Furzer K4CY

Because the IC-10 interface requires a TTL high (+5V) at its CTS pin (coming from the IF-232C) before it will transmit data to your computer you must <**Set RTS high**> in Logger32's CAT configuration. Also, your RS232 cable must connect the computer's RTS to the radio's CTS and *vice versa* (each side's RTS raises the other's CTS, meaning "Go ahead, I'm waiting for the data").

49.5 Kenwood TS-480



◀ Configure the CAT connection using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**

Choose whichever Com port you have connected to the radio for CAT, and whatever baudrate is configured on the radio.

<**Use CTS/RTS Hardware handshaking**> should be selected for the TS-480.

Feel free to experiment with different polling intervals to find one where the log entry pane frequency display keeps up with VFO knob spins, and the CAT connection is stable and reliable.

For sound card digital modes, assert PTT on pin 3 of the rear panel DATA connection to mute the front panel microphone input and enable the DATA audio input.

49.6 Kenwood TS-570 and TS-870

The procedure for interfacing the Kenwood TS-570, TS-870, and TS-2000 transceivers to Logger32 is very simple, and almost identical for all three radios. The COM connector on the rear panel of the transceiver allows you to directly connect your computer using a standard RS232 cable: no level-converter interface is needed. You can still use all transceiver controls while under computer control.

The RS232 cable must be a standard "straight through" cable terminated with a 9-pin female connector at the radio end and either a 9 or 25-pin female connector to suit your PC serial port. Most USB-serial adapters use 9-pin connectors.

49.6.1 Setup the TS-570/870

To use computer control with the radio, set up the Communication Parameters section of the radio's built-in menu:

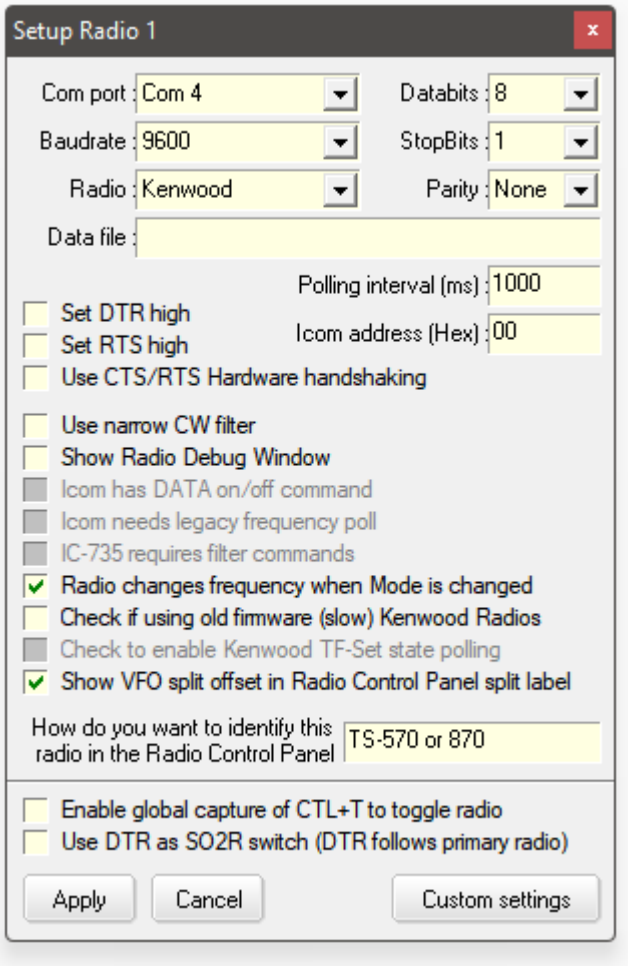
Copyright © 2026 Robert C. Furzer K4CY

- **TS-570**: from menu item № 35 (Computer Interface), select one of the eight options. The recommended option is the Kenwood default of 96-1 *i.e*. 9600 bps with 1 stop bit.
- **TS-870**: from menu item № 56 (COM.RATE), select one of the eight options. The recommended option is the Kenwood default of 96-1 *i.e*. 9600 bps with 1 stop bit.

After selecting the desired menu item, you *may* need to turn the radio off then back on for the new parameters to take effect.

49.6.2 Configure the TS-570/870 in Logger32

The TS-570 and TS-870 operate with no parity and 8 data bits. Setting up Logger32 is simple using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**



From the drop-down lists, select the following:

- **Com port**: select whichever PC serial port is connected to the radio. Logger32 only lists the available/active ports: if you can't see the correct port listed, make sure the radio is physically connected and on.
- **Baudrate:** choose the same baud rate as configured in the radio's menu.
- **Radio**: choose *Kenwood*.
- **Databits**: 8.
- **StopBits**: the number of stop bits as configured on the radio – generally 1.
- **Parity**: none.

Accept all the other defaults and click <**OK**>.

Note: The suggested values of 9600 baud for the COM port and 1000 ms (1 second) for the polling interval are just a starting point. If your radio and computer can handle faster baud rates, select the highest baud rate that works *reliably* for you. Increasing the baud rate makes little if any different in practice, but reducing the polling interval towards 150 or maybe 100 ms (polling the radio more often) gives a faster and smoother display of band, frequency and mode changes. If your PC or radio can't quite keep up, the CAT connection will become unreliable and may fail completely. If you have high stray RF levels in the shack due to a mismatched antenna, inadequately shielded cables and poor RF earthing, the CAT may fail when you transmit.

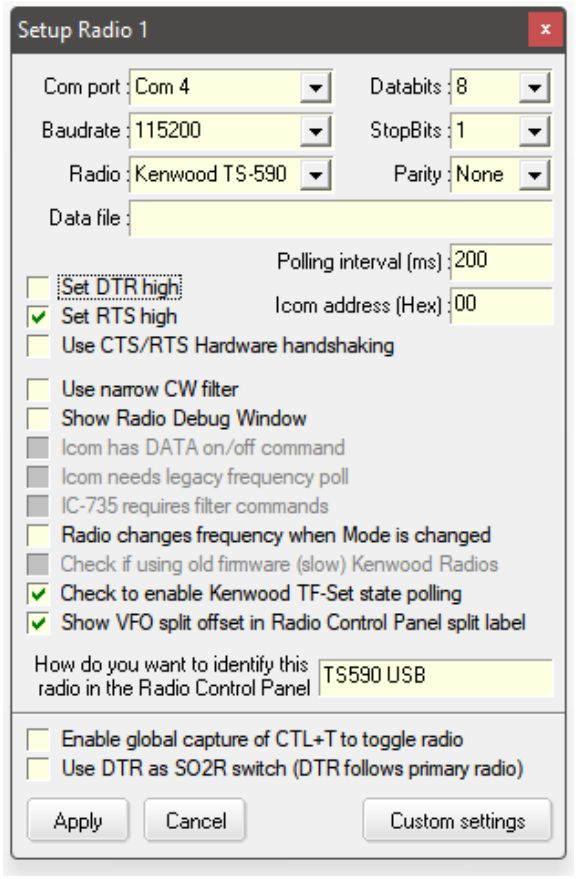
49.6.3 Codes for filter passband (slope tuning) on the TS-870

On the TS-870, the macro $**Command FW0100;IS+1400**$ sets the low edge of the passband to 1000 Hz, the high edge to 1400 Hz, giving a 400 Hz filter passband centered at 1200 Hz.

Similarly, the macro $**Command FW0030;IS+3400**$ sets the low edge to 300 Hz and high edge to 3400 Hz, giving the normal 3100 Hz bandwidth for everyday SSB.

Copyright © 2026 Robert C. Furzer K4CY

49.7 Kenwood TS-590



This is the configuration suggested for the TS-590SG using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

There are two options to connect a PC to the TS-590 for CAT control: an RS232 COM port or a USB port. To use the USB port, you need to install the driver properly on your PC. See the TS-590SG instruction manual.

Baudrate is set separately for each port in the TS-590 built-in menu: the defaults are 115200 baud (for USB) and 9600 baud (for RS232), but the speeds can be changed via radio menu items № 68 or 67 respectively. You may need <**Set DTR high**> and/or <**Set RTS high**> depending on your radio and CAT interface. If you intend to use the $**TF-Set**$ and/or $**MouseTF-Set**$ macros, select <**Check to enable Kenwood TF-Set state polling**>.

Hinson tip: if you mistakenly choose "Kenwood" rather than "Kenwood TS-590" as the radio type, Logger32 will be unable to access VFO B, causing a great gnashing of teeth.



- Setup in Radio Control Panel: these are sample macros configured in RCP Main.

  Macro for UP 1:

  $**Command FT0;FB#Split+1#;FT1;**$

  $**ClearCallsignOnQSYOff**$

  $**ClearQSYMarker**$

  Macro for bandwidth 200 Hz:

  $**Command FW0200;**$

  Macro for TF-SET:

  $**Command TS1;EX06600001;FL2;**$

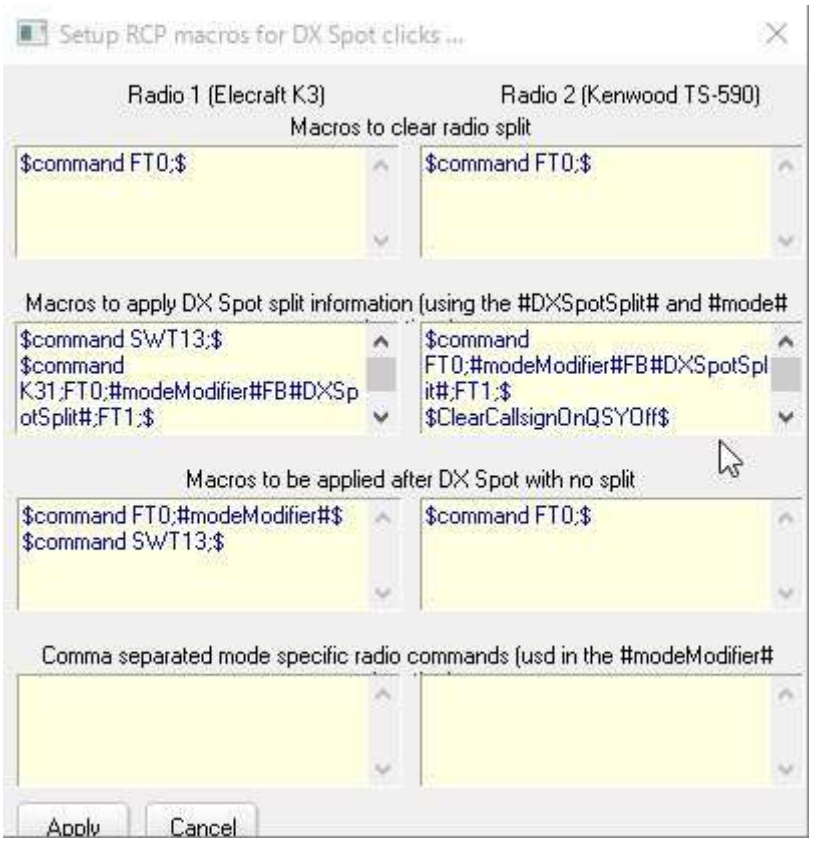
  Macro for TF-SET off:

  $**Command TS0;EX06600000;FL1;**$

  $**Command FB;**$

Copyright © 2026 Robert C. Furzer K4CY

- Setup DX spot macros. Here is an example RCP macro:



Here are the details of those macros to apply DX spot split as the form above does not show them fully:

$**Command FT0;#ModeModifier#FB#DXSpotSplit#;FT1;**$

$**ClearCallsignOnQSYOff**$

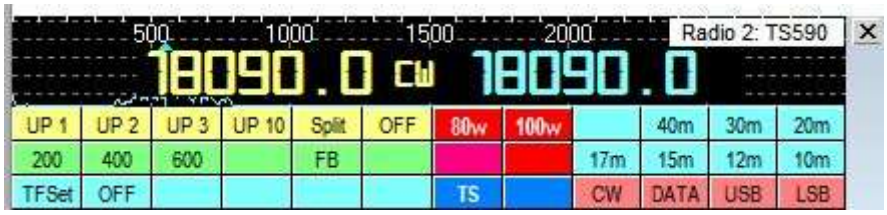
$**ClearQSYMarker**$

- Frequency and mode displayed in **Radio Control Panel** main: in this example VFO A is set to 18085 kHz CW and VFO B is set to 18090 kHz CW. <**Show VFO A and VFO B side by side**> is enabled.
  - Simplex mode with VFO A in use:



    VFO B mode is not displayed. This is normal.

  - Simplex mode with VFO B in use:



Copyright © 2026 Robert C. Furzer K4CY

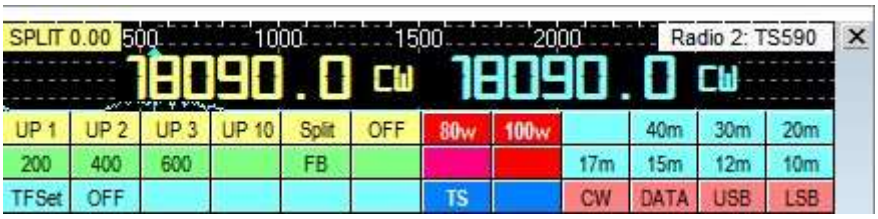
VFO B frequency/mode is displayed on the left and VFO B frequency on the right.

- Split mode with VFO A for RX and VFO B for TX:



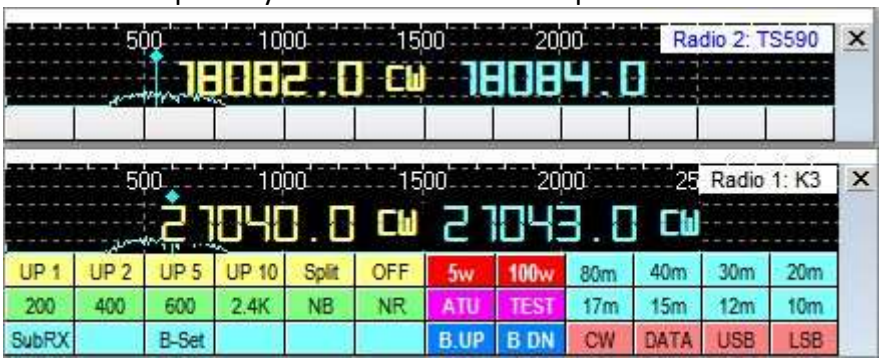
VFO B frequency/mode is displayed on the left and VFO B frequency on the right.

- Split mode with VFO A in use. Hold TF-SET button to swap VFO A and B:



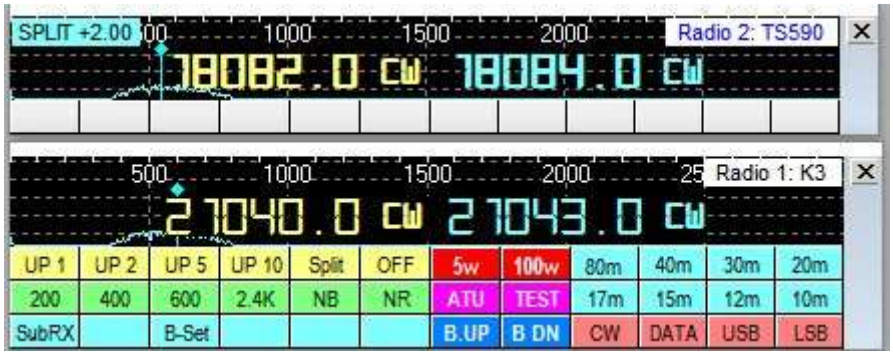
VFO B frequency/mode is displayed both left and right.

- **Frequency and mode displayed in Radio Control Panel SO2R**: in this example, my K3 is Radio 1 and TS-590SG is Radio 2.
  - Here Radio 1 is the primary radio. Radio 2 is in simplex mode ▼



VFO B mode is not displayed in Radio Control Panel SO2R. This is normal.

  - Here Radio 1 is the primary radio while Radio 2 is in split mode.

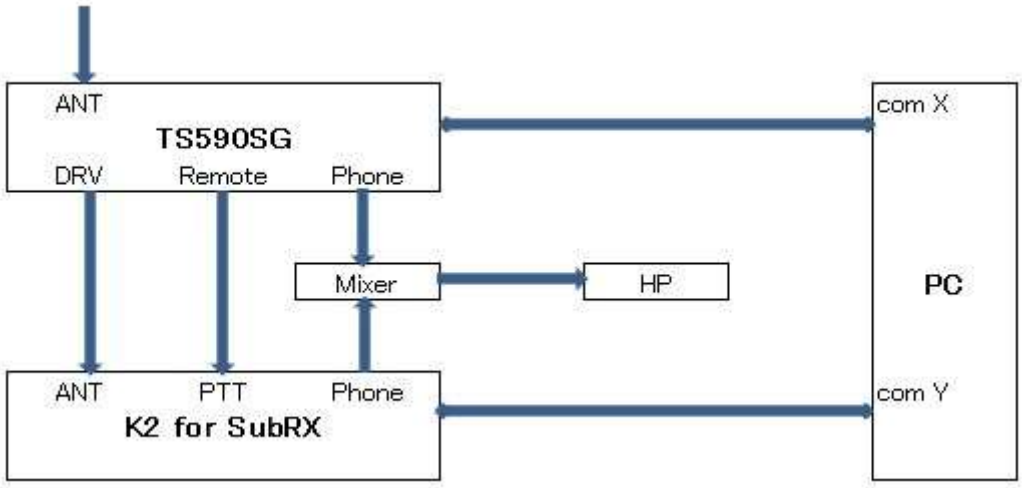


Copyright © 2026 Robert C. Furzer K4CY

49.7.1 How to add an *external* sub-receiver

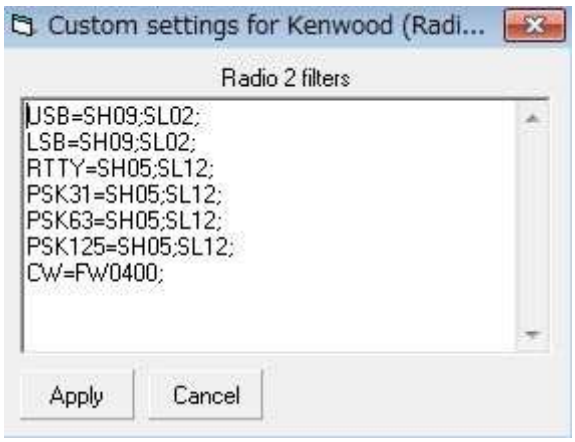
The TS-590 and TS-480 radios lack built-in sub-receivers, a disadvantage of these otherwise great transceivers. You may add an *external* receiver, for example using an Elecraft K2 to listen to the DX station while tuning around in his pileup on the TS-590SG to find a clear frequency on which to call him. Both radios should operate in simplex (not split).

- **Using an echo port**:
- **Using K2 as Radio 2**: when you click a DX spot or turn the TS-590SG VFO knob, Logger32 must



command the K2 to same frequency as the TS-590SG. This command is configured in one of the Radio Control Panel buttons.

49.7.2 Kenwood TS-590 filter setup



Kenwood radios do not have a CAT command to set IF bandwidth except in CW and FSK mode.

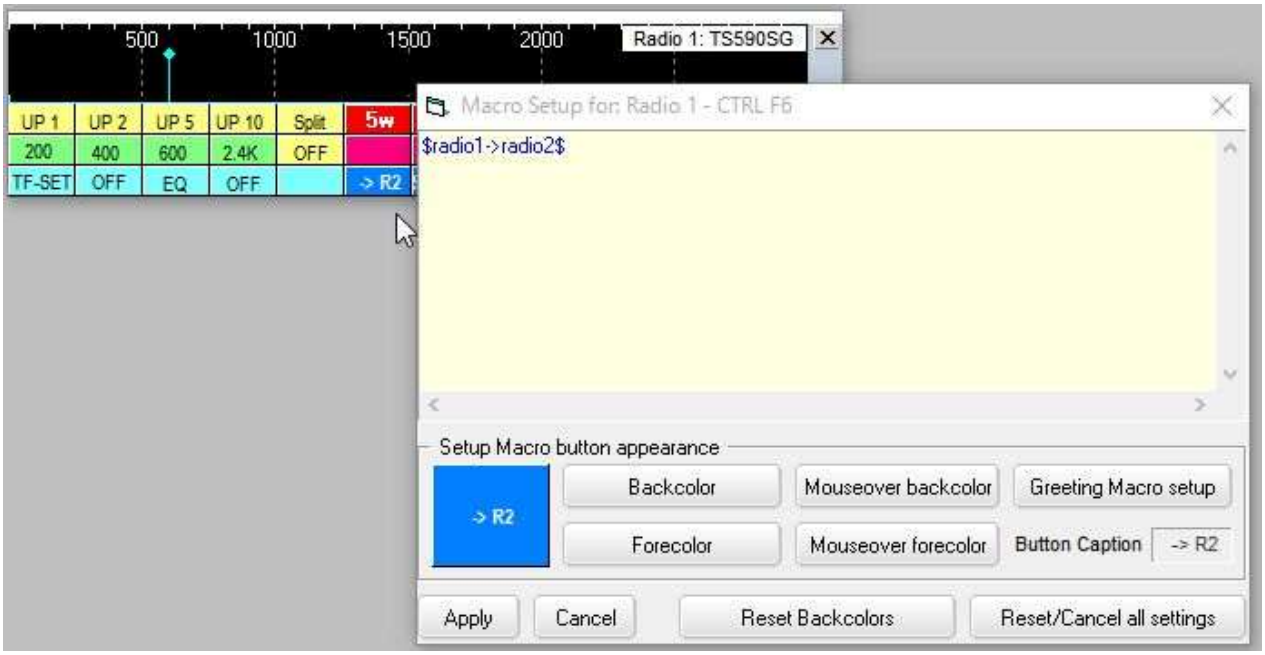
Use SH (**S**et **H**igh-cut frequency) and SL (**S**et Low-cut frequency) instead using the custom settings button under **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

49.7.3 Logger32 setup

- **TS-590SG setup**: TS-590SG is configured as Radio 1.
- **K2 setup**: K2 is configured as Radio 2. See the Elecraft K2 K3 KX3 section for details.
- Radio Control Panel setup: when you click a DX spot with a split comment, both radios should not turn on split mode. For this purpose <**Enable Set Split Macro when clicking a DX spot**> should *not* be ticked.

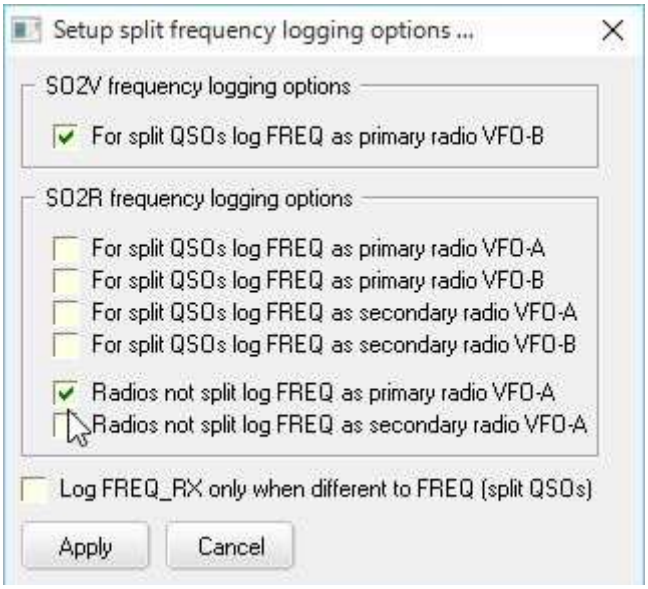
Copyright © 2026 Robert C. Furzer K4CY

Configure the following macro to tune the K2 to the same frequency and mode as the TS-590SG.



Logging frequencies are written depending on checked option in the Logging frequency setup table located in the **R**adio **C**ontrol **P**anel.

In this example primary radio VFO A frequency is written as FREQ and secondary radio VFO A frequency is written as FREQ_RX in the logbook ▶



49.7.4 Using SDR as panadapter or second receiver for TS-590SG

It is possible to add SDR support to the TS-590SG, allowing other apps also to access and control the radio. *[The following instructions do not cover the download/installation of any of the tools, only the process for configuring them to work together on the PC - not Linux or Mac: sorry!]*

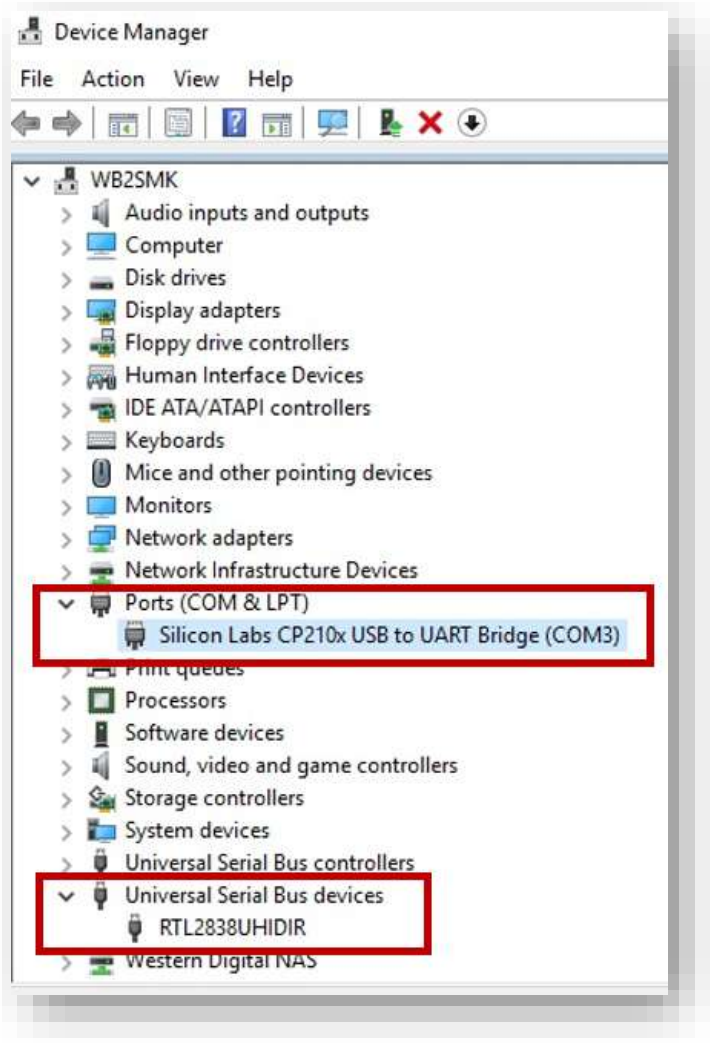
- **My setup**: I currently have the following items set up in my shack:
  - Kenwood TS-590SG
  - Logger32, naturally
  - HDSDR

Copyright © 2026 Robert C. Furzer K4CY

- JT-65 HF HB9HQX Edition
- WSJT-X
- Nooelec dongle (RTL2832) + Nooelec Up-converter
- VSPE (**V**irtual **S**erial **P**ort **E**mulator software)
- Omni-Radio
- Windows 10 PC

- **Display spectrum via HDSDR**: this lets you:
  - See the full ham (or other!) band to see where the activity is or isn't.
  - Identify file-ups or open areas for calling CQ
  - Look for particular types of activity *e.g.* JT65, SSB, CW, pileups.
- **Link frequency and mode**: the system connects the various elements to share the frequency and mode. This allows the SDR to provide a panoramic display of the spectrum space in use and links the radio to the SDR as well as JT-65, FT-8 and Logger32 so that frequency and mode track between them all.
- **Settings**: various elements in this system require their own unique settings. These settings were derived based on many resources found on-line supplemented with my own experimentation. While it is possible that other combinations will work even better, this set has at least worked for me.

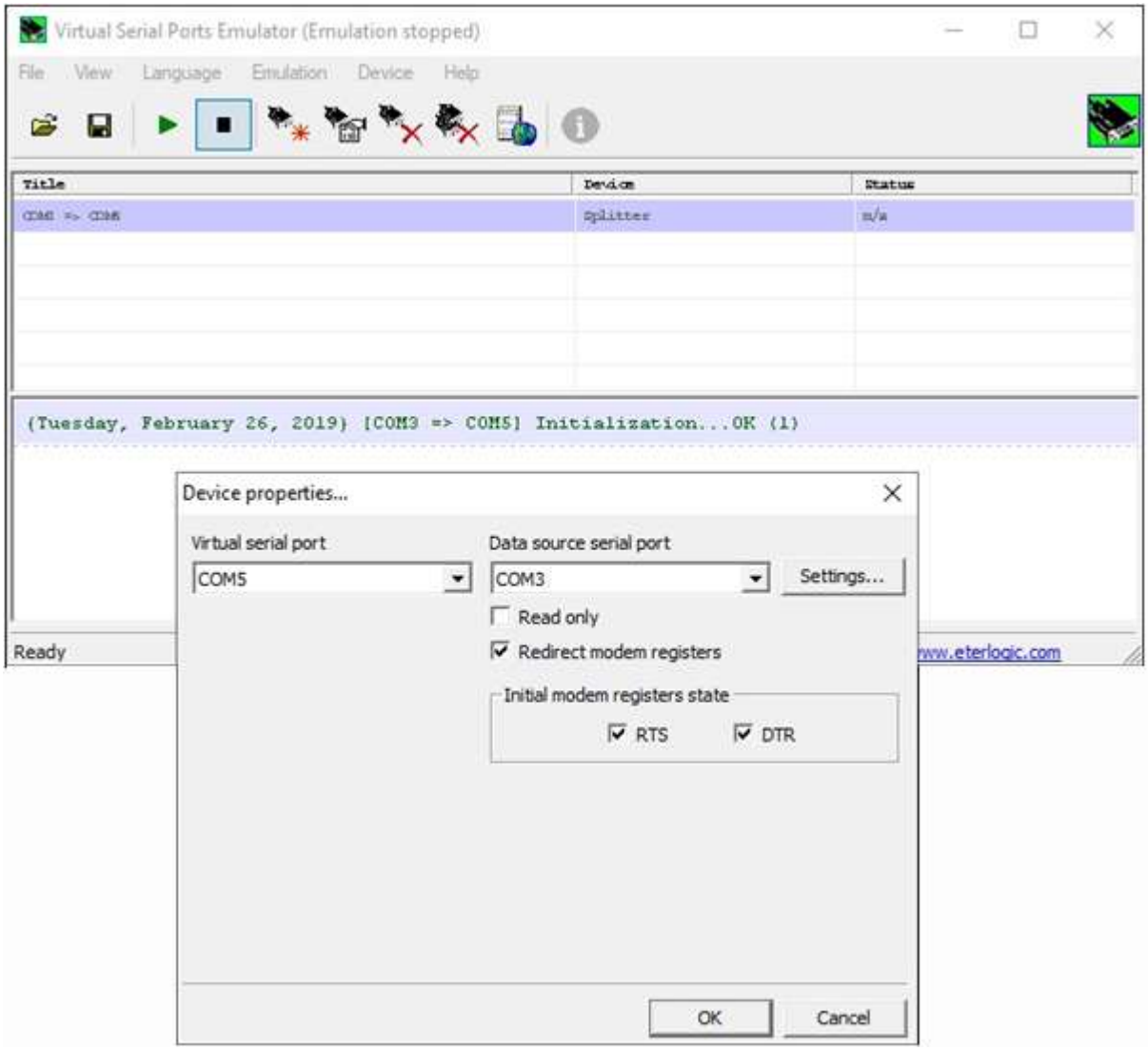
  With the radio and SDR dongle plugged in, Windows Device Manager shows the two items. In this example, COM3 is the Silicon Labs chip in the TS-590SG radio, while the RTL2838UHIDIR is the Nooelec SDR dongle ▶



- **Radio**: all of this is easy to do because the TS-590SG supports an antenna output to drive a second receiver, which in this case is the SDR. On the radio, you must change Settings Menu 85 to set the DRV function to antenna. On other radios you would probably have to tap into the IF for a panoramic display, or use a fast T/R switch to feed the antenna to both the radio and the SDR in receive but physically disconnect/isolate the antenna from the SDR while the radio is transmitting.

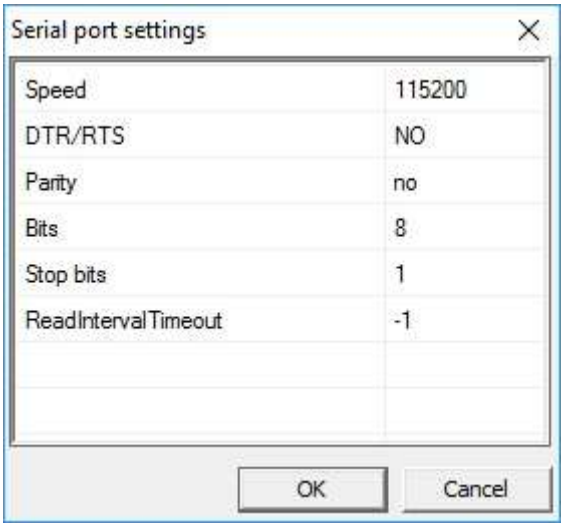
Copyright © 2026 Robert C. Furzer K4CY

- **VSPE**: this package provides virtual serial ports and is currently a key component in the configuration. When starting up, it will nag you to buy the 64-bit license but you do not need it: the 32-bit (free) mode works fine so just click <**No**> to the purchase offer, then <**OK**> for the next window. Create a "Splitter" to create a link to virtual port COM5 from physical port COM3. You may need to use other ports based on what's available in your PC.



I found that I needed RTS/DTR to start in the "on" condition.

Next, click <**Settings**> and set it up according to your radio's communications settings ▶

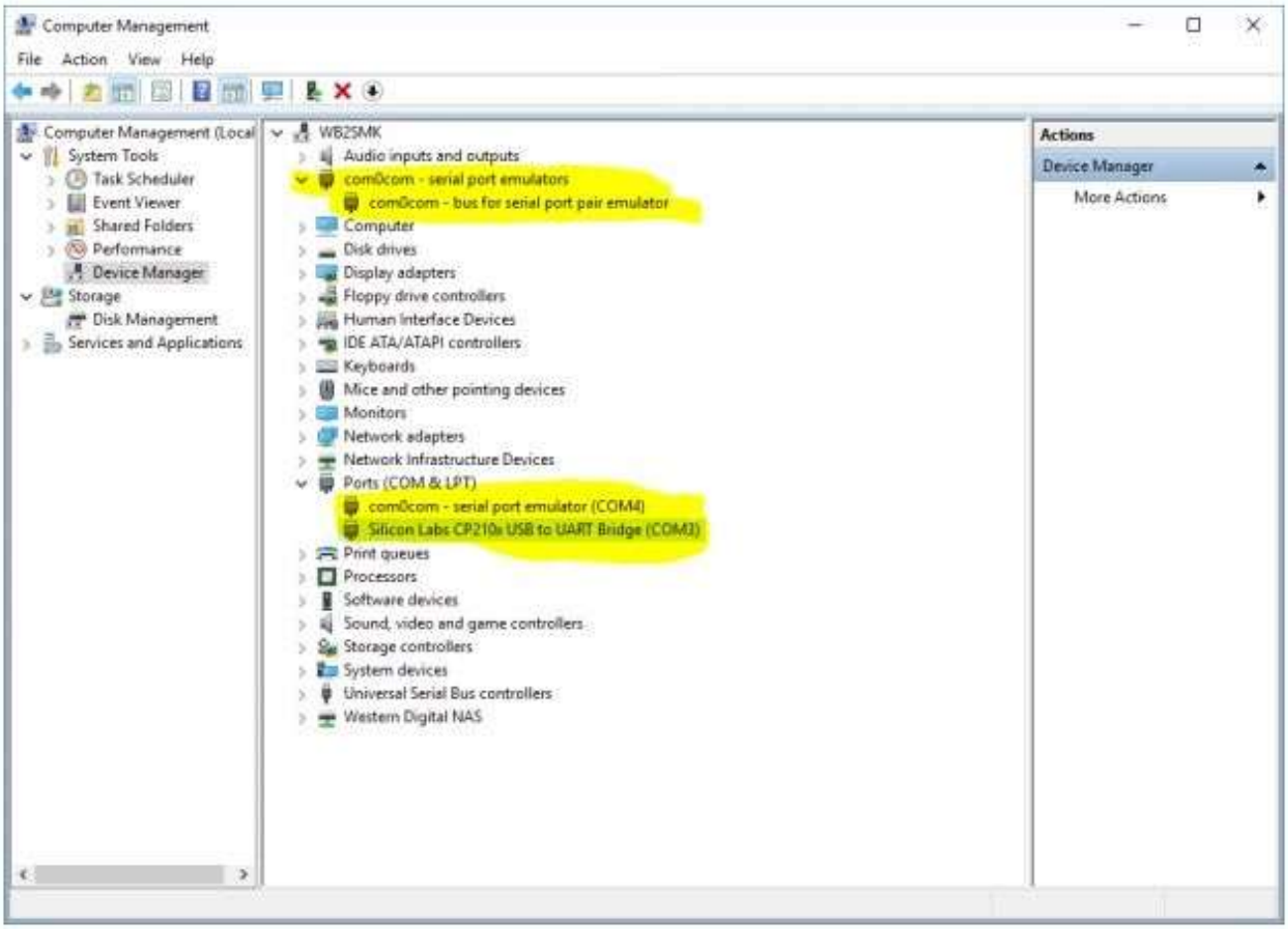


The settings ***must not*** include DRT/RTS. I did not find any other combination of settings that worked!

Once configured, be sure to "Run" the configuration using the green arrow button.

Copyright © 2026 Robert C. Furzer K4CY

The virtual port COM5 does *not* show up in the Windows Device Manager port listing. Looking at the Hardware Manager reveals ▼



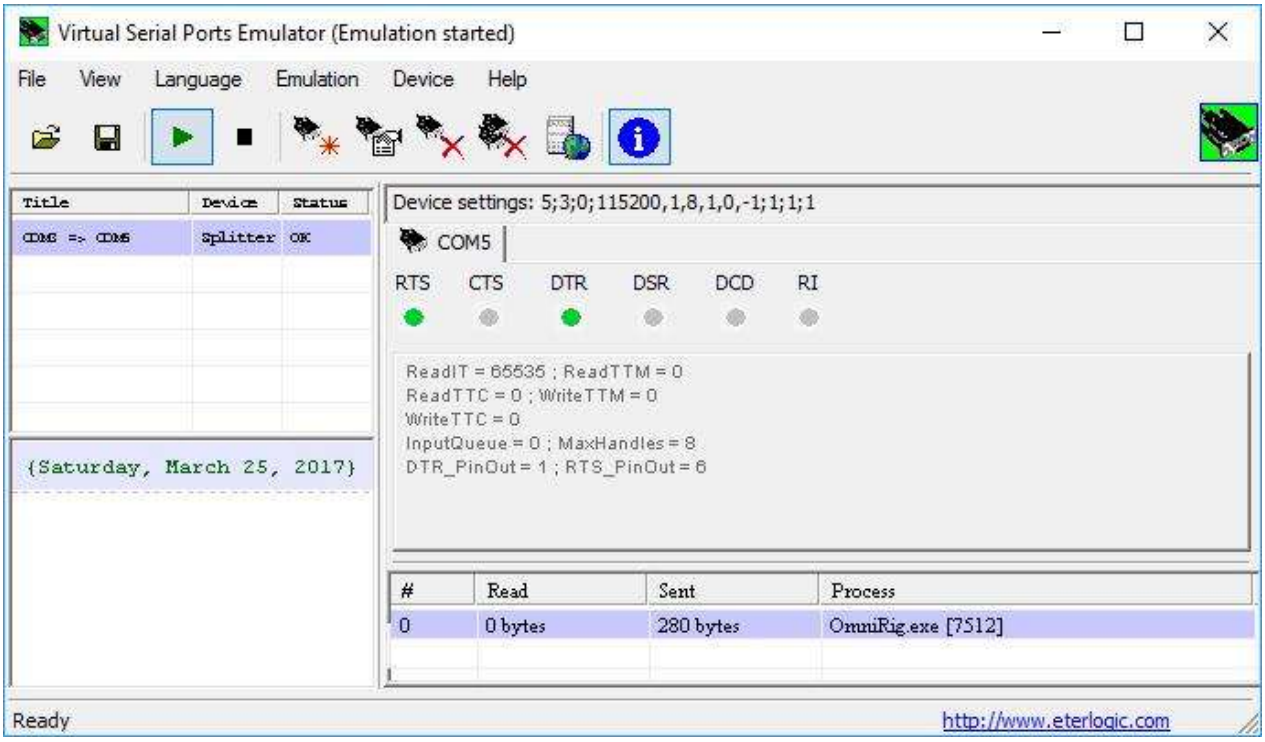
Notes:

- There is an entry called com0com. This does NOT appear when VSPE is not running, nor when it is running with nothing activated. If *two* of these show up, try deleting one. If that fails to get things working, reboot with your fingers crossed …
- The radio shows up as COM3 in the Ports section. This should always be there at the same COM port number and must match the VSPE configuration.
- A com0com emulated port shows up in the Ports section.

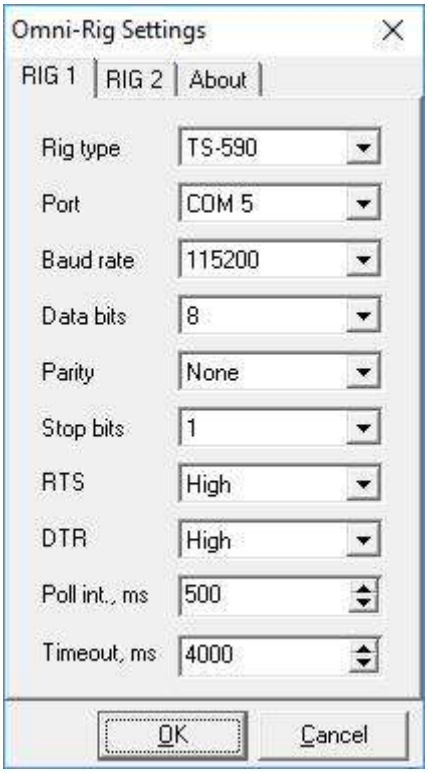
Again, the emulated port created by VSPE does *not* show up anywhere in this display!

Note: you will have to start VSPE and start the serial port emulation every time your PC reboots. If your programs aren't tracking, be sure to check this first. You can look at the serial port within VSPE using the information screen by clicking the blue "i" button.

Copyright © 2026 Robert C. Furzer K4CY



To make any configuration changes to the virtual serial port, close all programs that might access the physical or virtual port, make the changes and then restart everything. Since I rarely reboot my PC, I have not found this to be a problem.



- **HDSDR**: one of the many available SDR programs that run with various SDR dongles. It supports the Nooelec dongle that I am using but the exact choice of dongle or SDR hardware should not matter. I preferred the user interface of HDSRD over the others and it works fine for me. Set the SDR software to use Omni-Radio.

  From Options menu choose “CAT to Radio (Omni-Radio)” then check the following items:

  - sync Rig1
  - sync to Omni-Radio
  - sync from Omni-Radio
  - sync Tune frequency
  - sync Modulation

  If HDSDR “loses” the RTL dongle, ZADIG may need to be re-run to re-install the WinUSB driver instead of the default RTL driver. The ZADIG tool is fully explained in the installation instructions for HDSDR.

- **Omni-Radio**: the following settings can be made directly in Omni-Radio or this dialog can be opened from within HDSDR. Set Omni-Radio to be on COM5 (the virtual port created using VSPE), port parameters to match radio’s settings. For my radio, this is: 115,200 baud / 8 bits / no parity / 1 stop bit. Polling interval 500ms, timeout 4000ms as shown. If you run into performance issues on your PC or your radio, reduce the polling *rate* by increasing the polling *interval*. Polling introduces a slight delay when it updates the frequency and mode, but this does not affect overall operation.

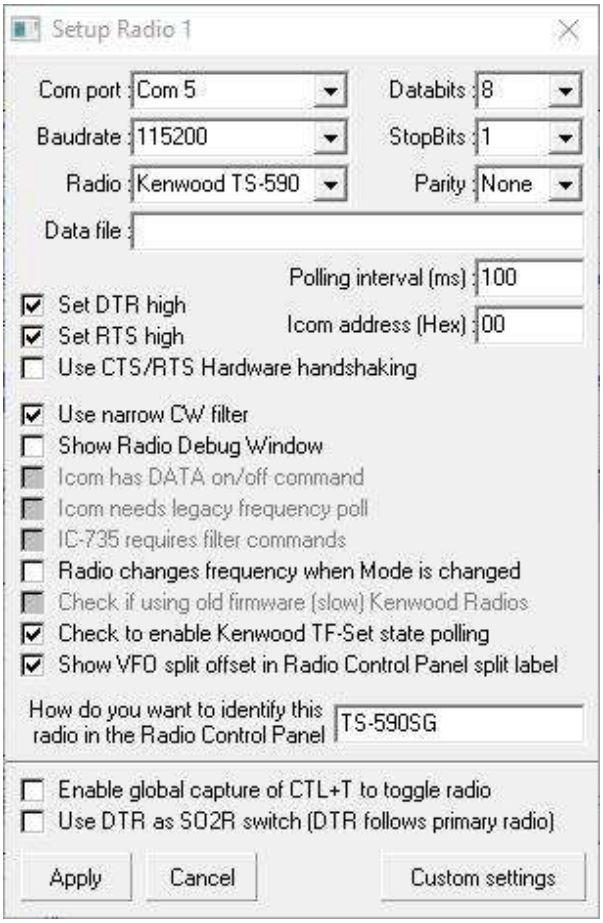
Copyright © 2026 Robert C. Furzer K4CY

- It appears that HDSDR uses the pmFreq definition within the Omni-Radio radio definition *.ini* file, while JT-65 requires the pmFreqA (or presumably pmFreqB) section. To resolve this conflict and restore HDSDR's ability to read/control the radio's frequency, I modified the radio definition file *C:\Program Files (x86)\Afreet\OmniRig\Rigs\TS590.ini* on my system. I copied the entire file for a safe back up, then copied the commands from the pmFreqA section to the previously empty pmFreq section.

  I have not made any other changes to this for the other programs that also work with this system.

  Since I am not entirely clear on the inner details of Omni-Radio / HDSDR / JT-65 I am not sure exactly why this is needed nor, frankly, why it actually works!

- **Logger32**: setup the radio on COM5 (the virtual port created using VSPE) with port parameters to match radio's settings using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶** Again, for my radio, this is: 115,200 baud, 8 bits, no parity and 1 stop bit.



This JT-65 HB9HQX Edition supports an Omni-Radio connection. The Hamlib connection did not work for me when attempting to connect to the virtual com port.

The Omni-Radio configuration file dates back to 2011. Inside, it indicates that pmFreq is not supported and there are no commands configured! For this reason in JT-65 HB9HQX Edition, you must set the Omni-Radio's Freq option to use pmFreqA so that it will use VFO A to control the radio's frequency. Without that, JT-65 will properly display the radio's frequency but is unable to set it. The other apps would still be able to change the radio's frequency.

The Mode portion of the Omni-Radio configuration may optionally be set to pmSSB_U which will automatically force the radio to upper sideband as needed for the JT-65 mode. A side-effect of this is that the TS-590SG will send the Morse code indicator for that mode change, every time!

Unfortunately, putting the radio into SSB mode is not enough. One must also put it into DATA mode to pass the audio via the USB connector. To facilitate this and to avoid interfering with the normal operation of pmSSB_U in case that's ever needed, I modified the pmDATA_U entry in the radio definition file by changing the default entry …

Copyright © 2026 Robert C. Furzer K4CY

; Mark Anderson WB2SMK; Make digital mode use USB but turn on DATA mode. I don't currently use direct FSK on the radio

; I had placed these commands in my Logger32 control panel so why not use them here too!

;$**Command LK00;**$ Unlock the front panel to make changes (possibly not needed)

;$**Command MD2;**$ Set USB mode

;$**Command FW2500;**$ Set filter to wide

;$**Command DA1;**$ Turn on DATA mode

;$**Command LK10;**$ Lock the front panel to avoid accidental changes

;$**Command AG0000;**$ Turn audio down to 0 (mute)

;$**Command PC030;**$ Low power at 30 watts

; Next we ONLY need the actual command, not the $**Command** prefix nor the $comment after …;

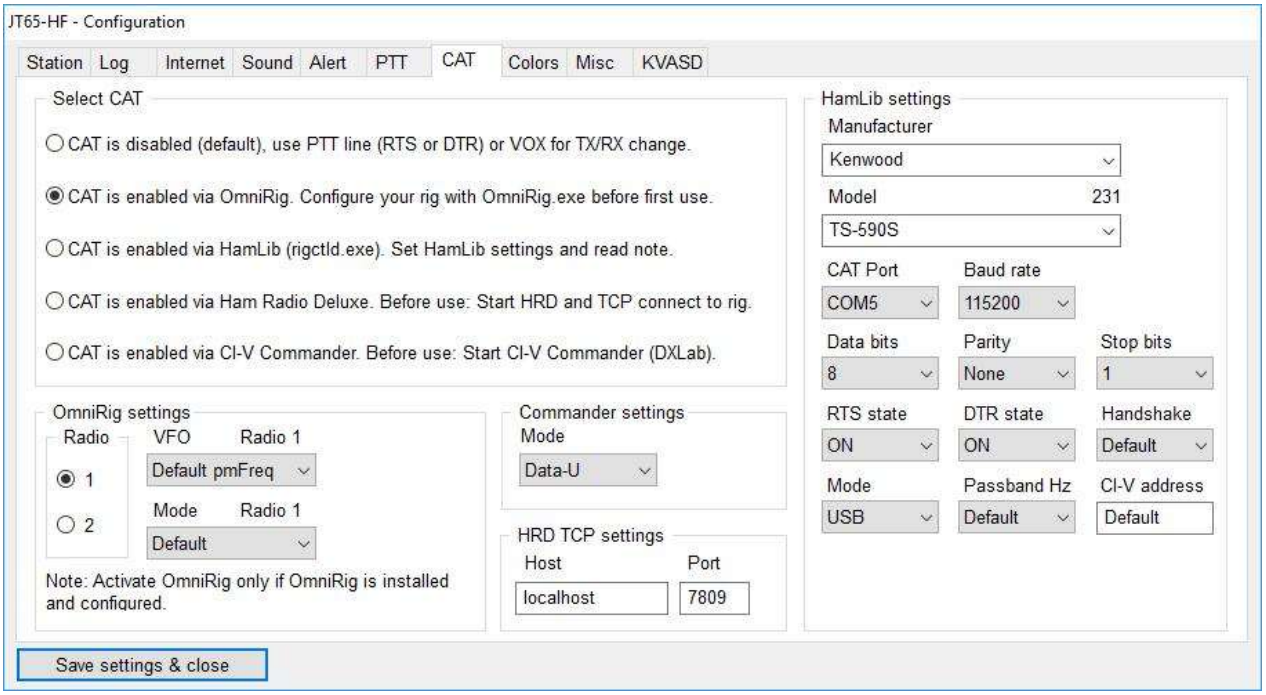
[pmDIG_U]

; Personalized command string - more than just "mode"

Command=(LK00;MD2;FW2500;DA1;LK10;AG0000;PC030;)

ReplyLength=0

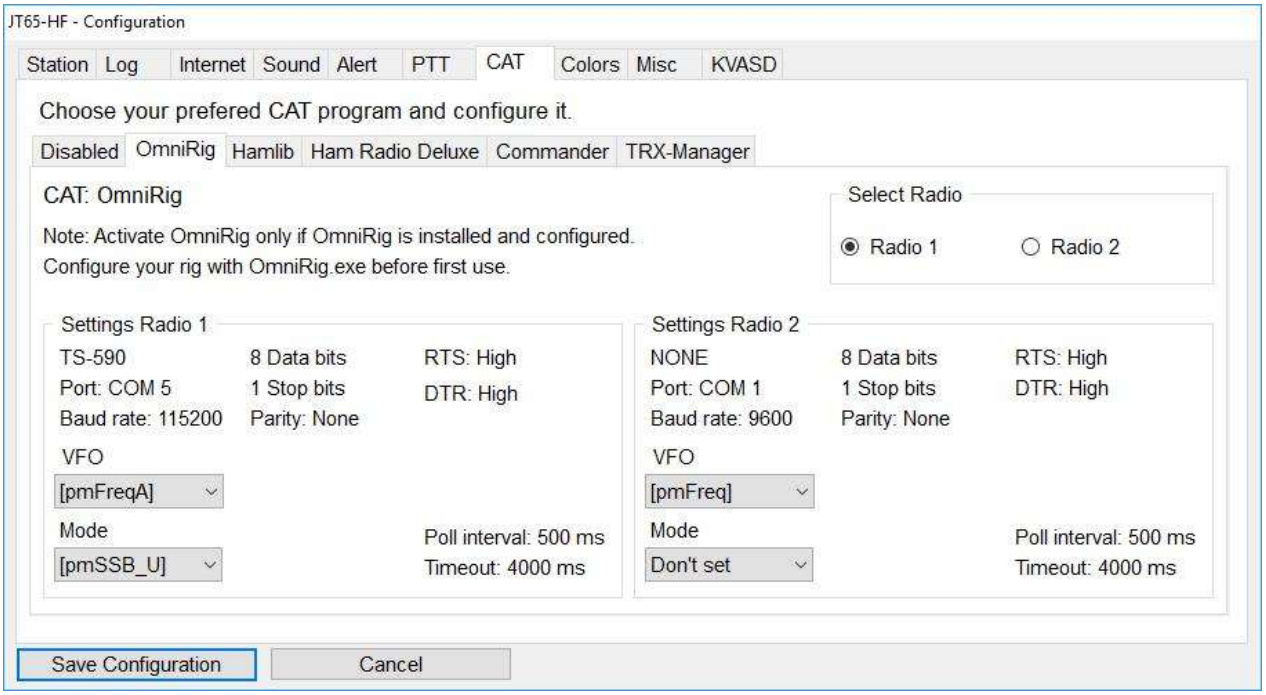
The overall configuration within JT-65 can be seen here ▼



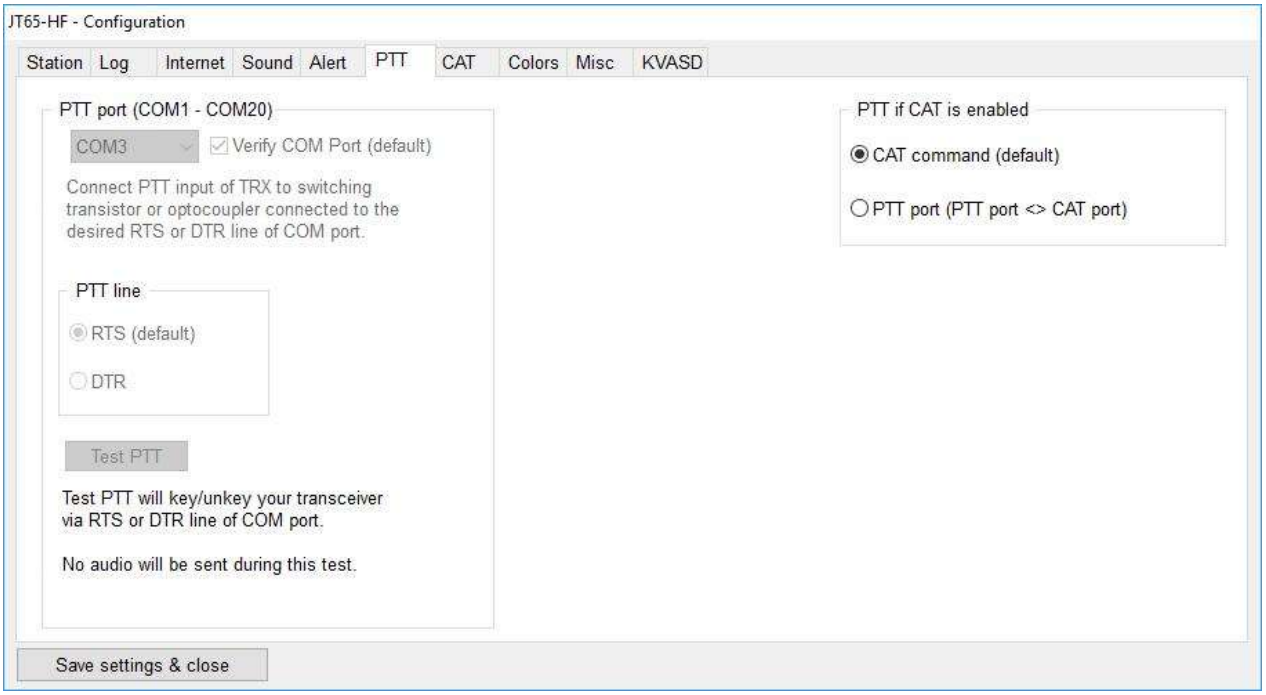
Note that this older version of Hamlib only supports the TS-590S, *not* the later TS-590SG.

Copyright © 2026 Robert C. Furzer K4CY

Selecting Omni-Radio opens a configuration page ▼



The PTT line must also be setup ▼

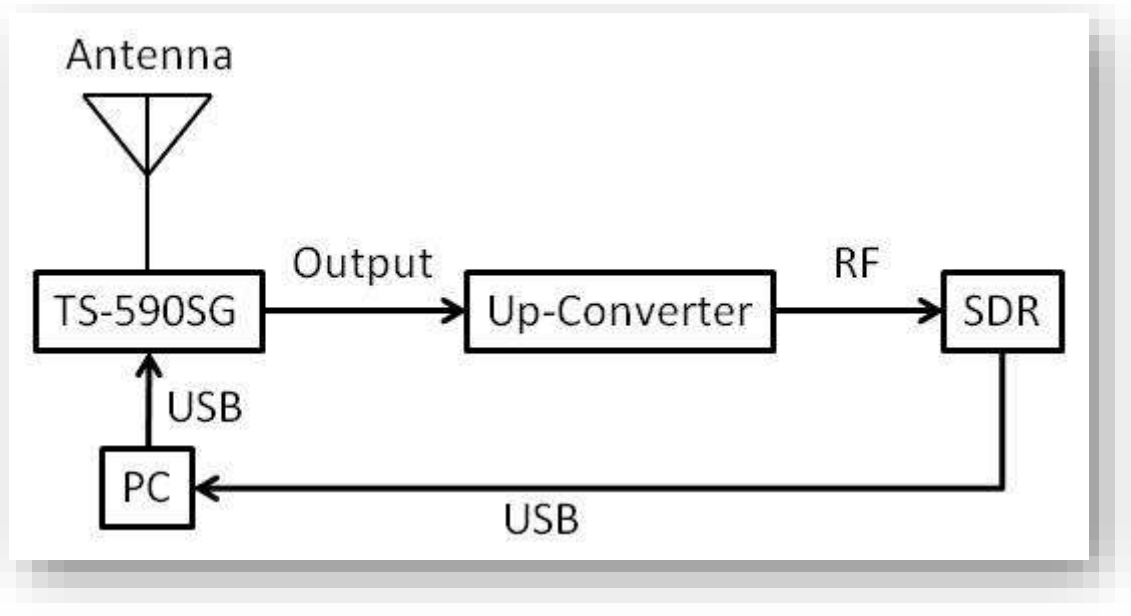


Note: this shows the manual edits to the radio definition file *e.g.* poll interval and timeout.

- **WSTJ-X**: I have also added support for FT# using the standard WSTJ-X package. Once the COM port was set to also use COM5, the package was ready to go. Other packages are likely to have very similar configuration details.

Copyright © 2026 Robert C. Furzer K4CY

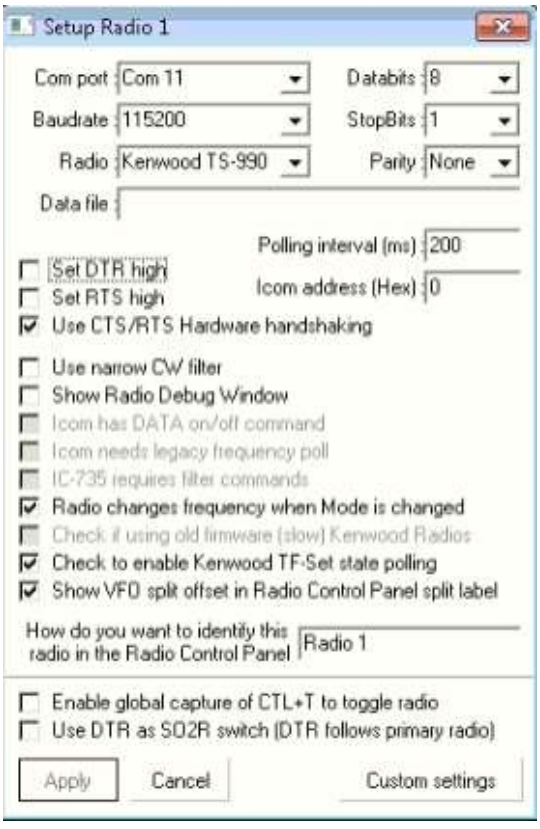
- **Physical connections**: straightforward ▼



  - **RF connections**: Ant → Radio → DRV connector → Up-converter → RTLSDR dongle → PC USB port
  - **CAT connection**: Radio → PC USB port as COM3
- **Web resources**: this working combination was achieved thanks to the following:
  - www.crystalradio.us/projects/ts-590s-panadaptor.htm [no longer online]
  - www.hdsdr.de
  - www.eterlogic.com/Products.VSPE.html

49.8 Kenwood TS-890

Select "Kenwood TS-**9**90" as the Radio using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

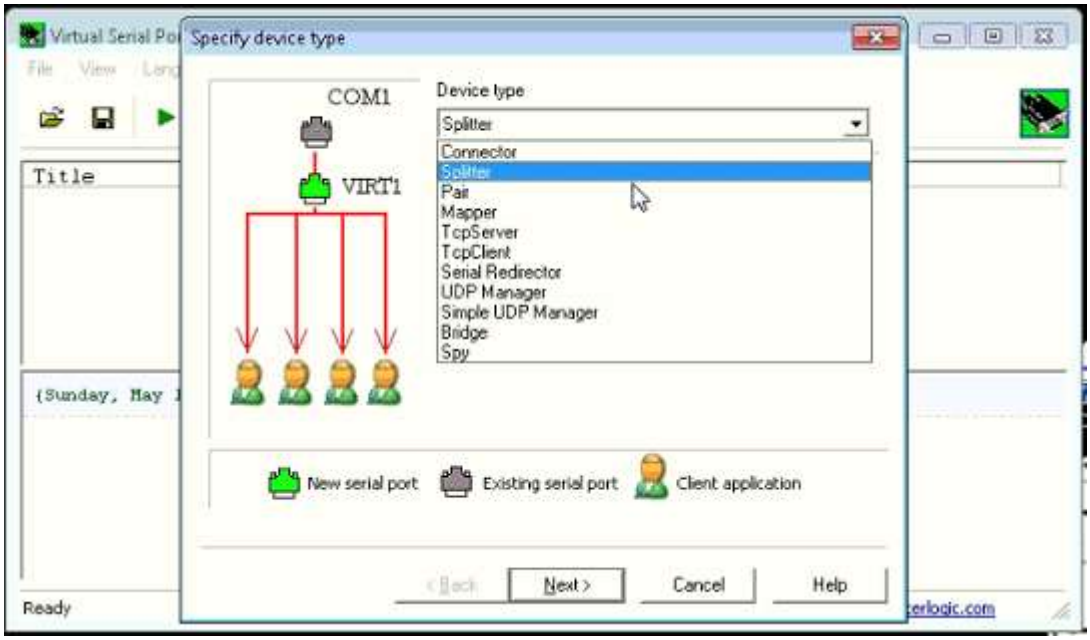


To share the radio COM port between Logger32 and JTDX, run JTDX directly (not from the <**Start**> menu in the UDP BandMap).

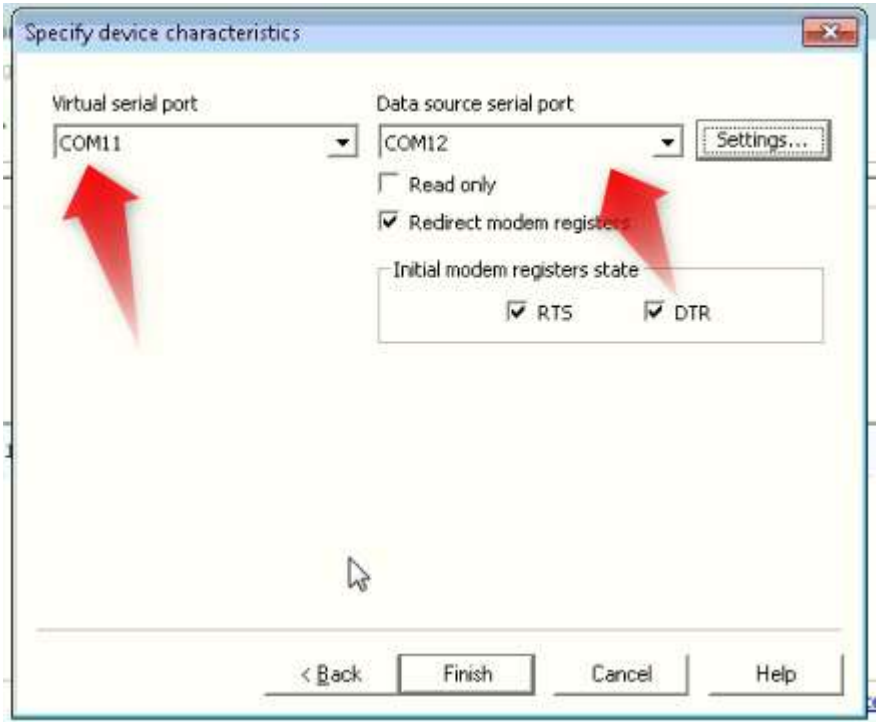
Example: while running FT8 using JTDX you may want to QSY to any frequency/mode which is spotted. You can share the COM port using VSPE. VSPE should be configured as below. Sharing a COM port using VSPE may not always work properly depending on the hardware.

Copyright © 2026 Robert C. Furzer K4CY

- Set the COM port device type to <**Splitter**>▼



- Set the source COM port, which you can confirm in Windows Device Manager, and set a Virtual COM port to be used on your software ▼



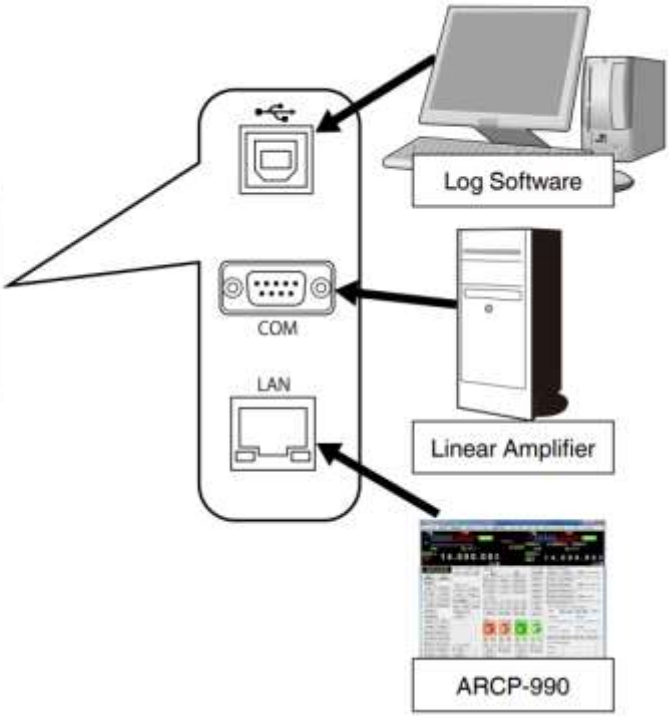
- Do not forget to go to <**Settings**> to set your baudrate.
- Last step is to start the virtual COM port you configured.

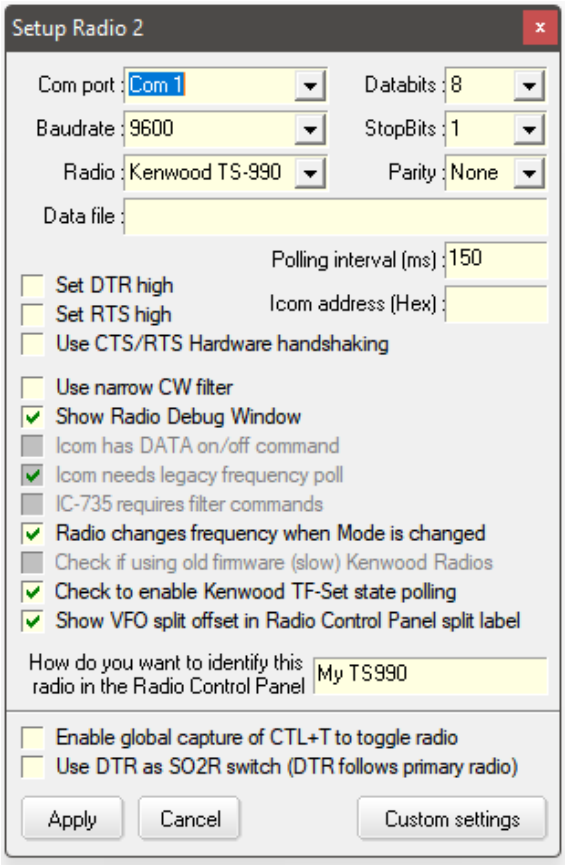
Copyright © 2026 Robert C. Furzer K4CY

49.9 Kenwood TS-990S

Kenwood's TS-990S PC control command reference guide shows connecting the TS-990 to a PC using its USB port for CAT (having first installed the Kenwood USB port driver on the PC), leaving the RS232 serial port to control PTT and band for a linear amplifier, and the LAN port for remote rig control via the Internet using Kenwood's ARCP-990 software ▼







◀ Configure the CAT port in Logger32 using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** something like this, using the COM port number from your PC and baudrate from the radio's menu.

CAT commands (to set/change or read/query something on the radio) and the radio's responses use Kenwood's conventional CAT message format consisting of ASCII sequences, each terminated with a semicolon. Tables in the command reference guide lay out the commands and parameters.

It is straightforward to compose, send and check out CAT commands using $**Command** *XXnnnn*;$ style macros assigned to buttons in Logger32's **R**adio **C**ontrol **P**anel. It is not quite so easy to figure out what the *XXnnnn* part of the macro should be, however: according to the guide, the TS-990S EX (EXtended?) CAT command, for instance, controls about 200 functions, each with numerous parameters. Read the guide carefully, and experiment.

Hinson tip: try to keep up with any firmware and manual updates from Kenwood, saving local copies to your PC in case, at some point, Kenwood withdraws support and takes them offline.

Copyright © 2026 Robert C. Furzer K4CY

Example: in order to operate his TS-990S on the 630m band, Ian G4GIR needs to set the radio's VFO frequency and mode, enable the RX antenna and enable drive. So far so good. He also needs the "Active High + Relay Control" setting for linear amplifier control on that band, whereas normally on the HF bands its uses "Active Low".

In a lengthy table, the Kenwood manual defines two "EX" commands to change the radio's linear amplifier control mechanism. Here's the row for linear control on the HF bands ▼

Advanced Menu											
P1	P2	P3	Function	P5							
				000	001	002	003	004	005	006 ~	
1	00	11	Linear Amplifier Control (HF Band)	Off	Active High	Active High + Relay Control	Active High + Relay & TX Delay Ctrl	Active Low	Active Low + TX Delay Control		

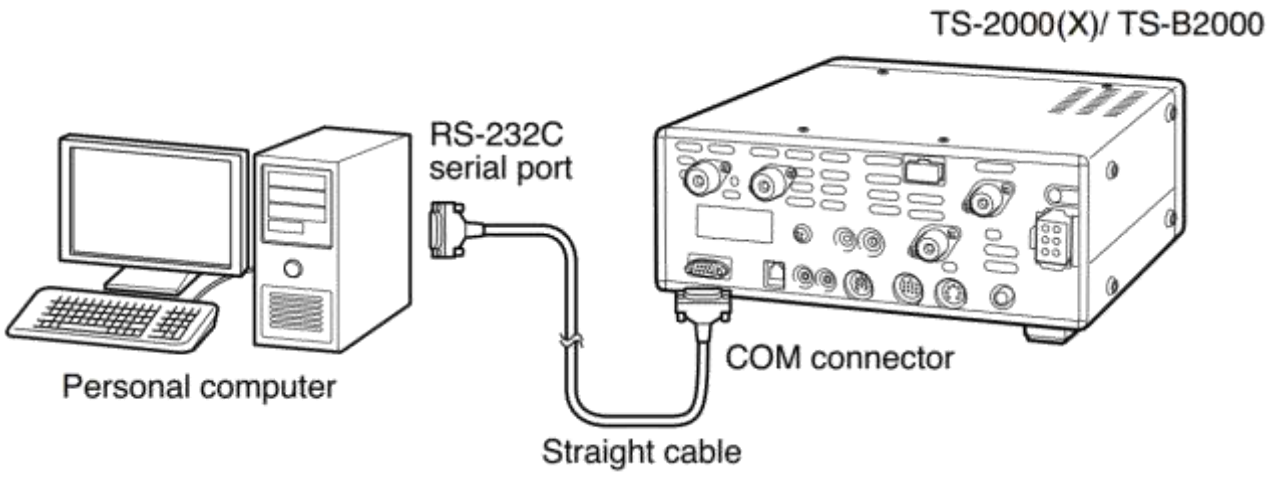
So, in Logger32's Radio Control Panel, Ian configured a button to trigger the following macro:

$**Command FA00000472400;**$	(Set VFO to 472.4kHz)
$**Command AN00111;**$	(Enable RX antenna)
$**Command OM03;**$	(CW mode)
$**Command EX10011 002;**$	(Set "Active high + relay control" for the linear)[469]

EX P1 P2 P3 P4 P5 Terminator

Bingo! Ian simply clicks the RCP button to configure his TS-990 and amplifier specifically for a spell on 630m, then clicks a different RCP button to trigger a second macro and revert to a preferred HF frequency and mode, disabling the RX antenna and using $**Command EX10011 004;**$ for "Active low" amp relay control for normal HF band operation.

49.10 Kenwood TS-2000



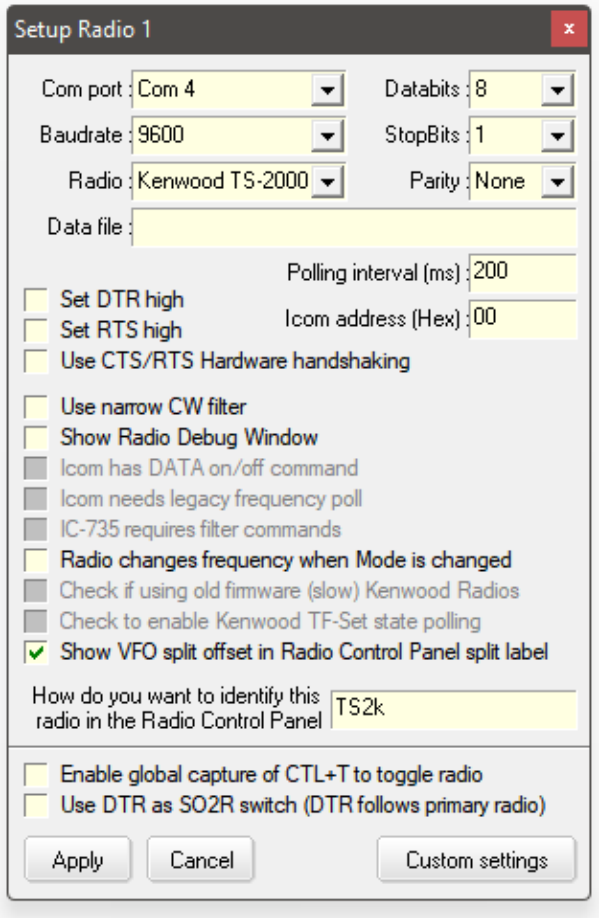
[469] Although parameter P4 is not specified in the table, the introduction to the EX command section of the manual says to use a space for the normal configuration or a 9 for initializing (whatever that means!).

Copyright © 2026 Robert C. Furzer K4CY

Under the TS-2000 menu item № 56 (COM CONNECTOR PARAMETERS), select one of the five speed options. For simplicity, we recommend the Kenwood default 9600 bps. Higher speeds give no appreciable improvement in practice, and might even make the CAT connection unreliable if either the PC or the radio can't quite keep up with the faster rate.

Then configure the CAT connection in Logger32 using

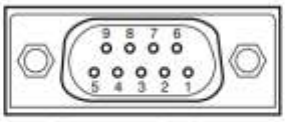
Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶



See the TS-2000 manual for more *e.g.* ▼

This transceiver uses a full-duplex, asynchronous, serial interface for communicating through the male 9-PIN RS-232C **COM** connector. Each data is constructed with 1 start bit, 8 data bits, and 1 stop bit (4800 bps must be configured as 2 stop bits). No parity is used. The pinout and the pin functions of the **COM** connector are shown below:

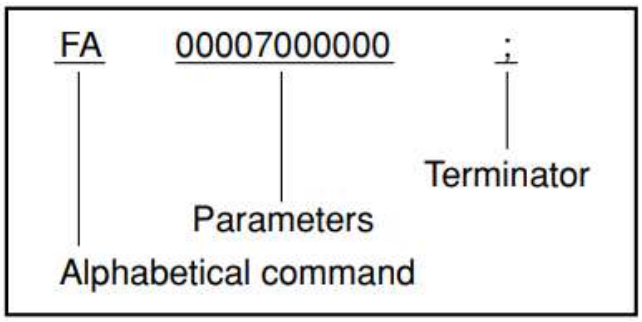
COM



Rear panel view

COM Pin No.	COM Pin Name (Ref.: Computer)	Function (Ref.: Transceiver)	I/O
1	NC	—	—
2	$\overline{RXD}$	Transmit data	Output
3	$\overline{TXD}$	Receive data	Input
4	NC	—	—
5	GND	Signal ground	
6	NC	—	—
7	RTS	Receive enable	Input
8	CTS	Transmit enable	Output
9	NC	—	—

Choose "Kenwood TS-2000" as the radio and the same Baudrate as you just selected on the radio, with 8 bits, one StopBit (or two at 4800 baud) and Parity None. Give it a suitable name for the Radio Control Panel and elsewhere in Logger32.



49.11 Kenwood FAQs

Q. VFO A works fine so why doesn't the VFO B frequency show up for my TS-590?

A. Have you configured the radio as a plain "Kenwood", by any chance, rather than a "Kenwood TS-590"? If so, that's your problem.

Copyright © 2026 Robert C. Furzer K4CY

50 TEN-TEC

> "At TEN TEC, we are making America and ham radio better, one radio at a time"
>
> *Mike Dishop N8WFF*

This section covers several TEN-TEC transceivers[470].

50.1 TEN-TEC OMNI VI

Here is an example command for the OMNI VI:

$**HexCommand FE FE 04 E0 06 00 04 FD**$ LSB, 0.5 kHz. 6.3 IF filter

The OMNI VI manual documents the mode but not the filter codes. Here is what you need to know. Each command begins with $**hexcommand FE FE 04 E0** (assuming your radio has the default address):

- Right after E0, **06** means set mode/filter.
- Next, **00** is the mode. **00**=LSB, **01**=USB.
- Next, **04** is the 6.3 MHz IF filter position. **02**=2400 Hz, **03**=1800 Hz, **04**=500 Hz, **05**=250 Hz. Note: **06** here toggles the narrow (9 MHz) filter On ⇄ Off *(Ed: I did not design the firmware that way. I am simply describing it!)*.

Here are some useful OMNI VI commands:

- $**HexCommand FE FE 04 E0 06 00 FD**$ LSB mode
- $**HexCommand FE FE 04 E0 06 01 FD**$ USB mode
- $**HexCommand FE FE 04 E0 06 04 FD**$ FSK mode

Each of these can be combined with filter selection codes. In fact, to set the filter, you must specify a mode in the same command *e.g.:*

- $**HexCommand FE FE 04 E0 06 00 02 FD**$ LSB, 2.4 kHz
- $**HexCommand FE FE 04 E0 06 00 03 FD**$ LSB, 1.8 kHz
- $**HexCommand FE FE 04 E0 06 00 04 FD**$ LSB, 500 Hz
- $**HexCommand FE FE 04 E0 06 00 06 FD**$ LSB, toggle the narrow 9 MHz IF filter

[470] TEN-TEC Programmers Reference Manuals are available for various TEN-TEC models at: www.tentec.com/category/product-documentation/old-products/

Copyright © 2026 Robert C. Furzer K4CY

For an OMNI VI+, with two narrow filters:

- $**HexCommand FE FE 04 E0 06 00 07 FD**$ LSB, narrow 2 (toggles the second narrow 9 MHz IF filter).

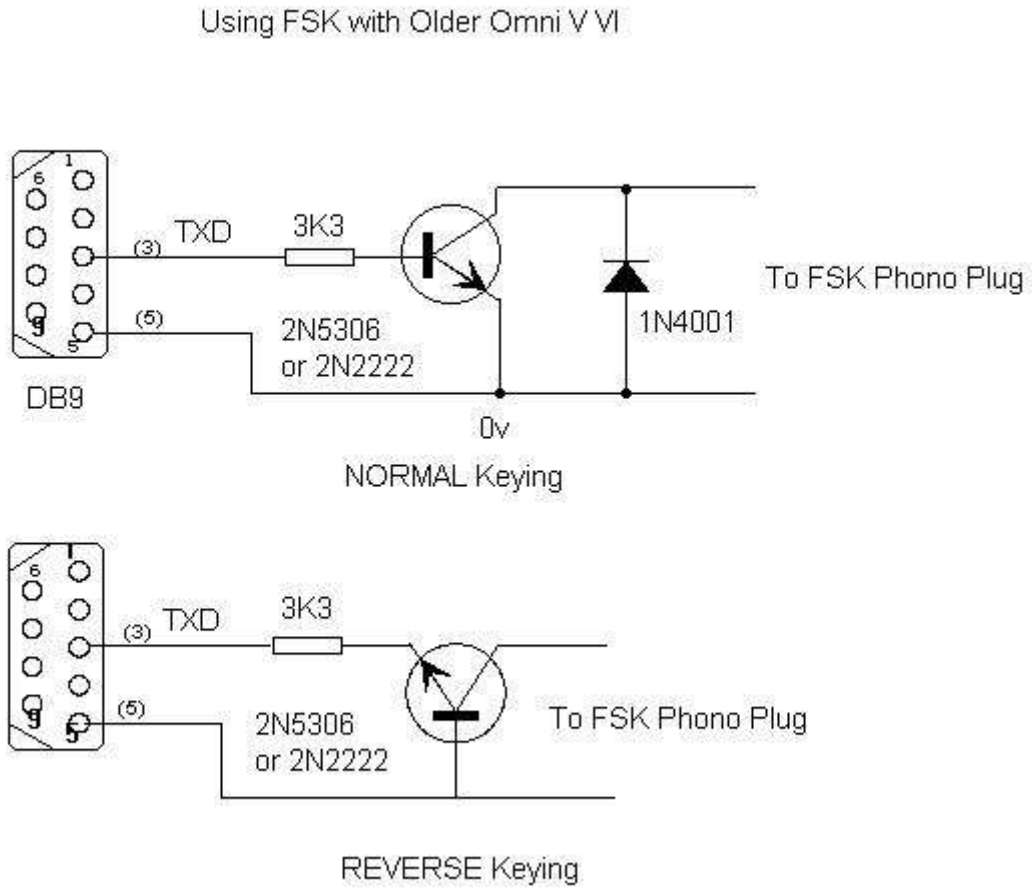
The $**QSX**$ macro works in all modes (USB, LSB and FSK) so there is no need to invoke special tricks to operate split.

50.1.1 Using FSK with OMNI V and VI

The following applies *only* to the FSK (Mark-Space) plug for the above radios. This fix applies *only* to serial ports.

Some older OMNI's transmit in reverse (inverted tones) with the conventional keying circuits mentioned. There are two ways to correct this problem:

Option 1: reverse the emitter and base connections to the keying transistor. Ground the base and connect the emitter of the NPN to the resistor. Keying will now be normal.



Option 2: connect TXD directly to a 1-10k Ohm resistor and feed that resistor directly to the FSK phone plug (Mark-Space) on the transistor. A value of 4.7k Ohms has been tested and works well.

Be sure there is a common ground between the computer and transceiver.

A very neat setup to key both the PTT and FSK involves building these circuits on a 5-pin connector, placing both circuits in a small prescription pill bottle. Push the serial port cable through the bottle before you begin wiring, and push the PTT and FSK leads through the top of the bottle before constructing the circuit.

Copyright © 2026 Robert C. Furzer K4CY

50.2 TEN-TEC OMNI V.9

The OMNI V.9 is a *modified* version of the TEN-TEC OMNI V using a replacement CPU with firmware developed by N4PY to emulate the ICOM CI-V interface protocol.

Setup info for the OMNI V.9 is:

Baudrate: 1200

Radio: Ten Tec Omni V.9

Polling interval: 500 ms

Icom address (Hex): 1E

50.3 TEN-TEC OMNI VII

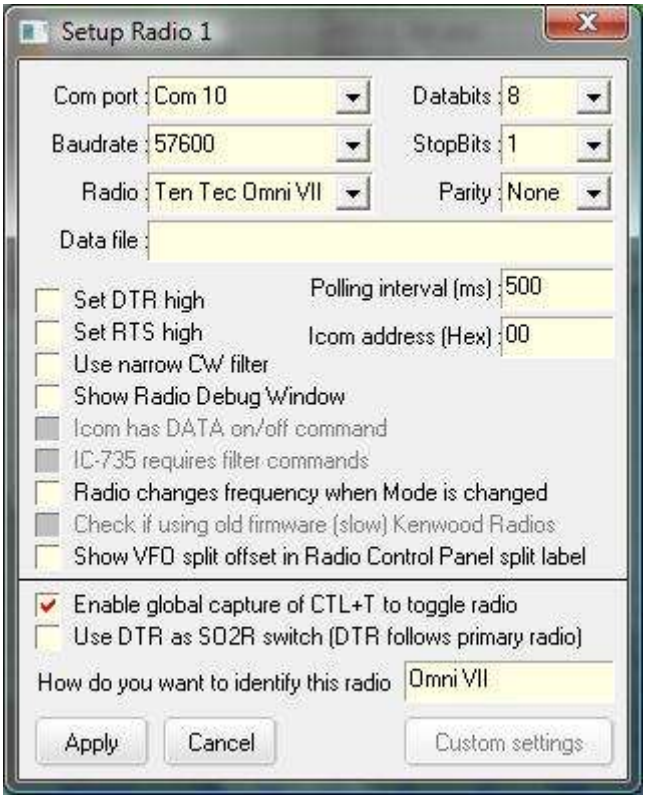
Use **Setup ⇨ Radio ⇨ Radio 1|2 configuration** to configure the CAT connection ▼

You can customize *C:\Logger32\Logger32.INI* to personalize the CW Normal, CW Narrow and FSK receiver bandwidths. The basic filter defaults are:

CW (Normal) - 1000 Hz

CW (Narrow) - 500 Hz and

FSK - 1000 Hz



Modify any of these values by adding/editing the *Logger32.INI* [Radio Port 1] or [Radio Port 2] sections to include the following:

OmniVII CW Filter Normal=xx

OmniVII CW Filter Narrow=xx

Omni FSK Filter=xx

… where xx denotes the filter width as outlined in the TEN-TEC Programmers Reference Manual.

For example:

[Radio Port 2]

OmniVII CW Filter Narrow=36

would set the Narrow CW filter for Radio 2 to 250 Hz.

Special adaptation of the $Command$ macro for use with the OMNI VII: the OMNIVII command structure uses a combination of ASCII characters and hexadecimal numbers. Whereas $**HecCommand**$ is for pure hex, the $**Command**$ macro can send ASCII characters and hex numbers enclosed with “<“ and “>“:

- $**Command** ASCII **<**hex number**>**$

Here’s an example. Let’s say we want to put the radio on 3505 kHz.

1. Convert the frequency to Hertz: 3505 x 1000 = 3505000

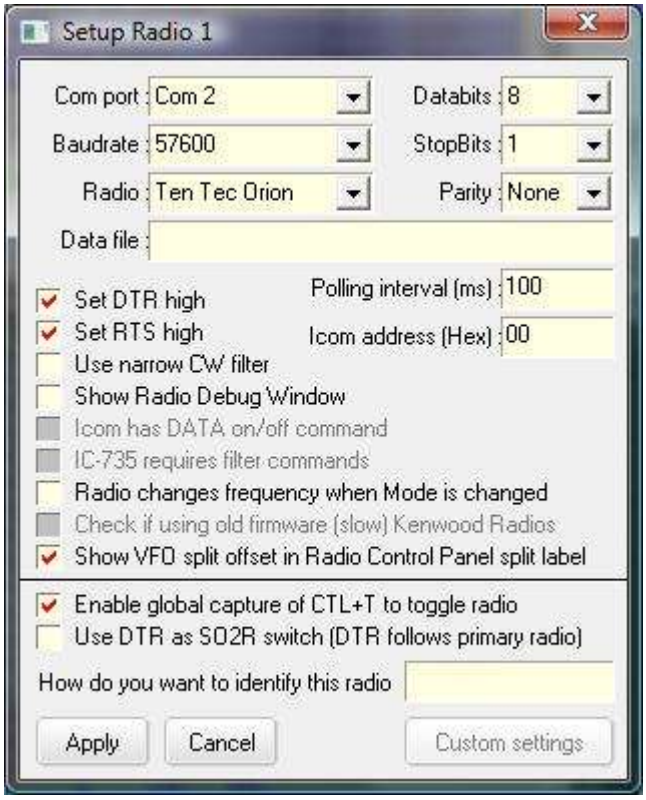
Copyright © 2026 Robert C. Furzer K4CY

2. Convert the frequency in Hertz to hexadecimal: Hex(3505000) = 357b68
3. Add leading zeros (padding) to make the hex string 8 characters long: 00357b68
4. Split the string into 4 pairs of bytes: 00 35 7b 68

Thus, the macro $**Command *A<00><35><7b><68>**$ sends the *A ASCII command to set the radio frequency, followed by a 4 byte hex number representing the frequency of 3505 kHz.

50.4 TEN-TEC ORION

Use **Setup ⇨ Radio ⇨ Radio 1|2 configuration** to confiture the CAT connection ▼



You can customize *C:\Logger32\Logger32.INI* to personalize the CW Normal, CW Narrow and FSK receiver bandwidths and the TX and RX bandwidth for SSB on the Orion.

These entries should be added to the [Radio Port] section of the file:

[Radio Port]

Orion Narrow CW=*RMF150

Orion Normal CW=*RMF750

Orion FSK=*RMF750

The syntax must be followed EXACTLY. The only spaces are after the word Orion and before the word CW. When Logger32 switches the Orion to FSK, the receiver bandwidth is set to 750 Hz.

For the SSB settings:

[Radio Port]

Orion SSB RX Filter=2100

Orion SSB TX Filter=2100

The value 2100 is for example only. Values greater than 2400 will be ignored.

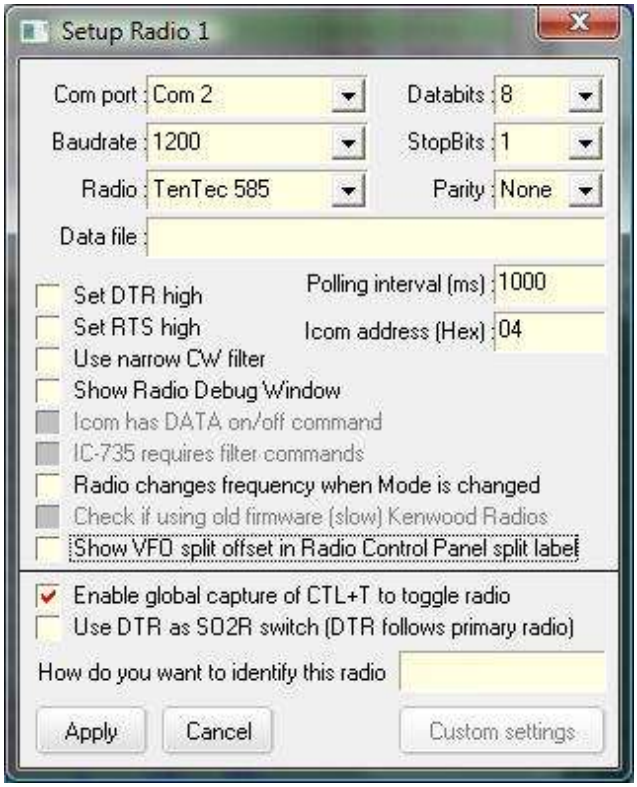
Notes:

- Given the mode “CW” in the Bands & Modes table, the Orion will select LCW on 160, 80, 40 and 30m UCW on 20, 17, 15, 12 and 10m.
- If <**Use narrow filter CW filter**> is enabled, the CW bandwidth is reduced from 500 to 250 Hz.
- When Logger32 switches the radio to LSB or USB, both the TX and RX filter widths are set to 2.4 kHz (unless otherwise modified as above).
- There are no special settings for the radio itself.
- PTT by radio command does not work on the Orion for CW but should be OK for data modes such as PSK, AFSK *etc*.

Copyright © 2026 Robert C. Furzer K4CY

50.5 TEN-TEC Paragon 585

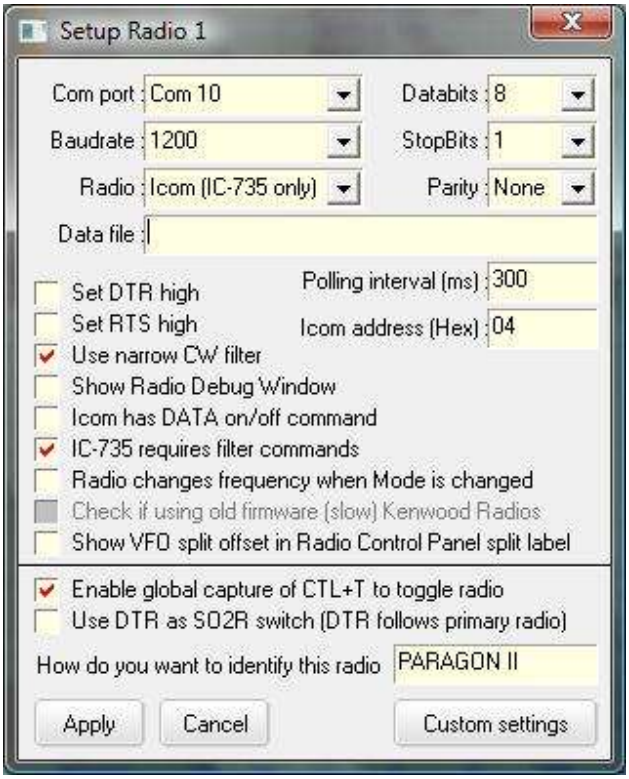
Configure the radio using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**, except with whatever Com port number your CAT connection uses. It may not be Com 2 ▶



50.6 TEN-TEC Paragon II

If your Paragon has been updated to use the N4PY chip that offers the ICOM CI-V protocol and is configured for the ICOM IC-735 control codes, use the ICOM (IC-735 only) radio setup.

This is a typical Paragon II (with N4PY chip) port setup using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** ▶



Filter selections must be entered into *C:\Logger32\Logger32.INI*. The normal default for SSB is 01 and the default for CW is 02. These need to be changed for the Paragon.

The following filter selections are available in the Paragon II:

Control code	Bandwidth
01	USB 6 kHz
02	USB 2.4 kHz
03	USB 1.8 kHz
04	CW 0.5 kHz
05	CW 0.25 kHz

Copyright © 2026 Robert C. Furzer K4CY

Set the desired bandwidths using the relevant control codes in the Icom Filter Normal and Icom Filter Narrow lines that you add to *Logger32.INI*. Below is an excerpt from *C:\Logger32\Logger32.INI* with the added filter lines shown in bold:

[Radio Port][471]

Automatic Open=True

ComPort=COM10,1200,n,8,1

Radio Type=Icom (IC-735 only)

Poll Interval=300

Set DTR High=0

Set RTS High=0

CW Filter=1

Icom Filter Normal=04

Icom Filter Narrow=05

Icom Address=04

Old Kenwood=0

IC-735 with Filters=1

[471] The port section in the *Logger32.ini* file may be headed [**Radio Port**] or [**Radio Port 2**] depending on whether the Paragon is configured as radio 1 or radio 2.

Copyright © 2026 Robert C. Furzer K4CY

51 Yaesu

> "Communicating through radio waves with people all over the world is what ham radio is all about and what keeps many people hooked"
>
> *Yaesu*

51.1 Supported Yaesu radios

The following Yaesu models can be selected under **Setup ⇨ Radio ⇨ Radio 1|2 configuration**:

FT-100	FT-2000[d] (works for FTdx-1200)	FT-757	FT-897	FT-991 (works for FT-891 and FT-710 too)
FT-1000D	FT-3000	FT-767	FT-900	FTX-1
FT-1000MP	FT-450	FT-817	FT-9000	
FT-101D (works for FT-710 too)	FTdx-5000	FT-840	FT-920	
FT-101MP	FT-747	FT-847	FT-950	
		FT-857	FT-990	
		FT-890		

51.1.1 *Un*supported … but usable … Yaesu radios

If your particular radio model is listed, use that selection – obviously enough – or something very similar (*e.g*. for the "FTDX101D" use "YAESU FT-101D"). If it is *not* listed (*e.g*. the FTdx-1200 or FTdx10), try configuring Logger32 to use a similar model, meaning a Yaesu of about the same vintage with a similar CAT format and commands (*e.g*. the FT-2000 or FT-101MP, respectively). You may need to experiment to find a setting that works OK for you (*e.g*. we understand from users that the unsupported FT-891 works OK if configured as an FT-991, and the FT-710's CAT resembles an FTdx101d's, so choose the Yaesu FT-101D option in Logger32's CAT port configurer). There may well be differences in the command sets between the two radio types, controlling specific features and functions that are only present on one or other radio. Parameters may also differ, just as they differ between individual radios with various options fitted *e.g.* filters.

Hinson tip: if there are CAT commands/features that you *need* but simply don't work with your particular Yaesu using any 'similar' setup, you can always ask for your model to be supported. We will need to know the format and meaning of the CAT commands, so please make it easy to help you: find out where the CAT commands are documented, then email the information to the Logger32 reflector (*e.g.* a URL to the manual and the page numbers) … and be nice about it. It will take a while to code the changes and you may be asked to help test them before general release.

Copyright © 2026 Robert C. Furzer K4CY

51.2 Yaesu CAT connections

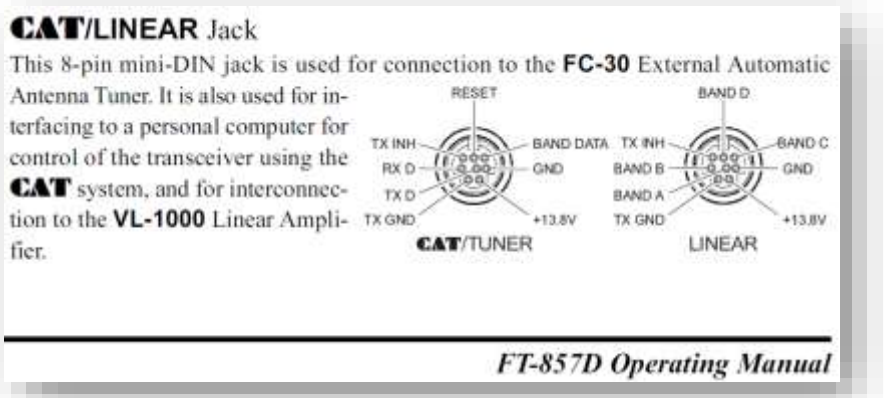
51.2.1 Yaesu CAT connector types

Yaesu provides three types of CAT connectors:

- **Type A**: the radio has a mini-DIN-8 jack for CAT ▶

  You need a CT-62 cable (or equivalent) to connect to PC's serial port*.

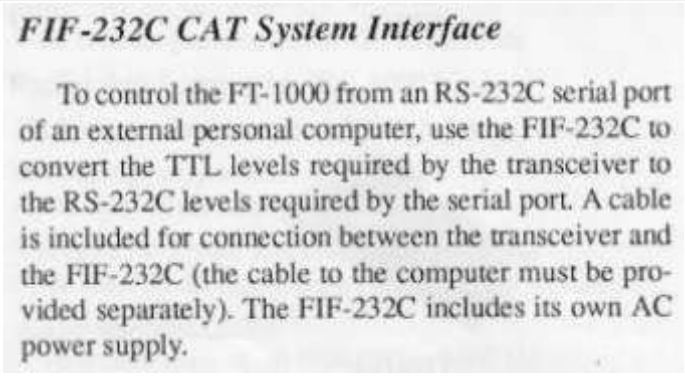
  Example type A Yaesus: FT-857, FT-897 …

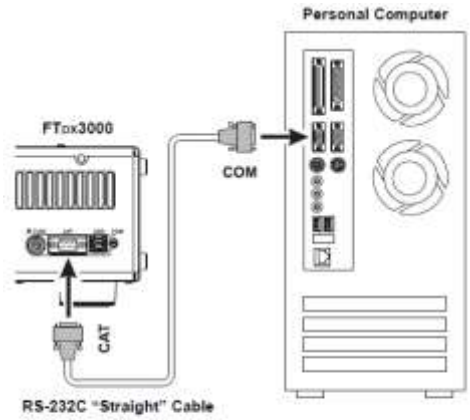


- **Type B**: the radio has an RS232 CAT port. ▶

  Connect a standard RS232 cable between the radio's RS232 port and the PC's RS232 serial port*.

  Example type B Yaesus: FT-1000MP, FT-2000D, FTdx-1200, FTdx-5000 …



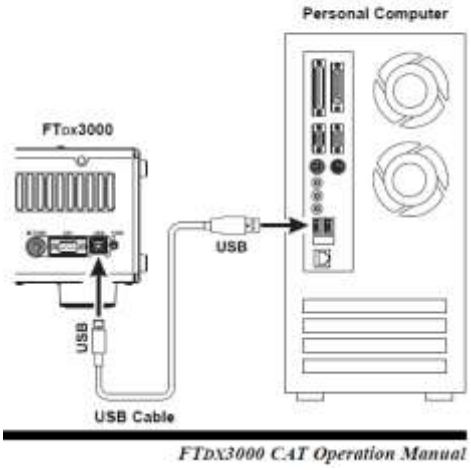


Type C: the radio has *both* RS232 *and* USB CAT ports. You may use *either* of them: choose one or the other.

To use the radio's RS232 port, connect a standard (straight pin-to-pin, not cross-over – apart from the FT 847 anyway) RS232 cable between the radio's RS232 port and the PC's RS232 port.

To use the radio's USB port, connect a USB cable between the radio's USB port and a PC USB port, and if necessary install the appropriate driver in the PC.

Examples of type C Yaesus: FTdx-3000, FT-991 …



*** Note for Type A and Type B Yaesus**: if your PC has USB but not RS232 ports, you will need a USB-serial adapter, preferably one using genuine FTDI chips. Cheap USB-serial adapters using fake/cloned/knock-off chips (usually Prolific) *may* not work if the fakery is detected and blocked by the driver software in recent Windows versions.

Copyright © 2026 Robert C. Furzer K4CY

51.2.2 Yaesu virtual COM port driver

Recent Type C Yaesu radios with built-in USB ports, or those that use the optional SCU-17 interface unit, require Yaesu's virtual COM port driver to be downloaded and installed on your PC. See the USB Driver Virtual COM Port Driver Installation Manual for details. The driver supports recent Windows versions and provides two virtual COM ports: one for CAT and another for audio.

Hinson tip: if you eagerly connected up the USB cable prematurely and find it isn't working, you *may* need to *un*install the default plug-n-play port driver installed automatically by Windows, then install the correct Yaesu driver. Do this through **Windows Control Panel ⇨ Device Manager ⇨ Ports (COM & LPT)**. For more help, try asking Yaesu or search YouTube. Good luck.

Hinson tip: you still need to match the radio's CAT settings with the port configuration in Logger32. Mismatched [virtual] COM port numbers, baudrates, even RTS/DTS settings can kill the CAT.

51.3 Yaesu CAT command protocols

Yaesu CAT command and response formats vary between models. Check your radio's instruction manual[472] to understand its capabilities. Yaesu adopted two types or families of CAT commands - one for older Yaesu radios, then something different for the newer ones …

51.3.1 CAT protocol for older Yaesu radios

In the original protocol for Yaesu's early CAT-enabled radios such as the FT-1000D, FT-847 and FT-897, CAT commands are sent as blocks of five *hexadecimal* bytes each, with no terminator.

Logger32 uses $***Hex*Command** $ to send hexadecimal CAT strings to these older Yaesus[473].

The *fifth* byte in each block is the command instruction (opcode), the *first four* bytes being arguments - either parameters for the instruction or padding: **every CAT block *must* be exactly five bytes long**, no matter how many bytes are actually required in the argument. Unused bytes should be padded with dummy values such as 00 (any hex value is acceptable but 00 is distinctive).

For example: $**hexcommand 03 00 00 00 07**$ sends the transceiver to CW because:

- 07 is the radio instruction (opcode) to change mode on VFO A;
- 03 is a mode parameter meaning CW;
- Three 00 padding bytes are added to reach the five required.

Hinson tip: to experiment with macros, I find it useful to open the **R**adio **C**ontrol **P**anel *plus* the radio debug window alongside to monitor the data (commands, parameters and responses) exchanged through the CAT connection. Right-click the **Radio 1|2** panel on the status bar.

Example of a basic CAT hex command sequence for an FT-847 or FT-897:

Hex command sent to radio	Interpretation
00 00 00 00 03	Query VFO A frequency and mode*

[472] On Yaesu.com, select Products ⇨ HF Transceivers/Amplifiers then scroll left or right to find your model and click to select it, then look under the **Files** tab for its CAT Operation Reference Book.

[473] HexCommand, Hexcommand, hexcommand *etc*. will do. LogGer32 doeSn't CaRe aBout thE casE.

Copyright © 2026 Robert C. Furzer K4CY

Hex response from radio

01 81 03 49 00 VFO A is on 18.103490 MHz LSB

Hex command sent to radio

02 10 25 00 01 Set VFO A frequency to 21.025 MHz

03 00 00 00 07 Set VFO A mode to CW

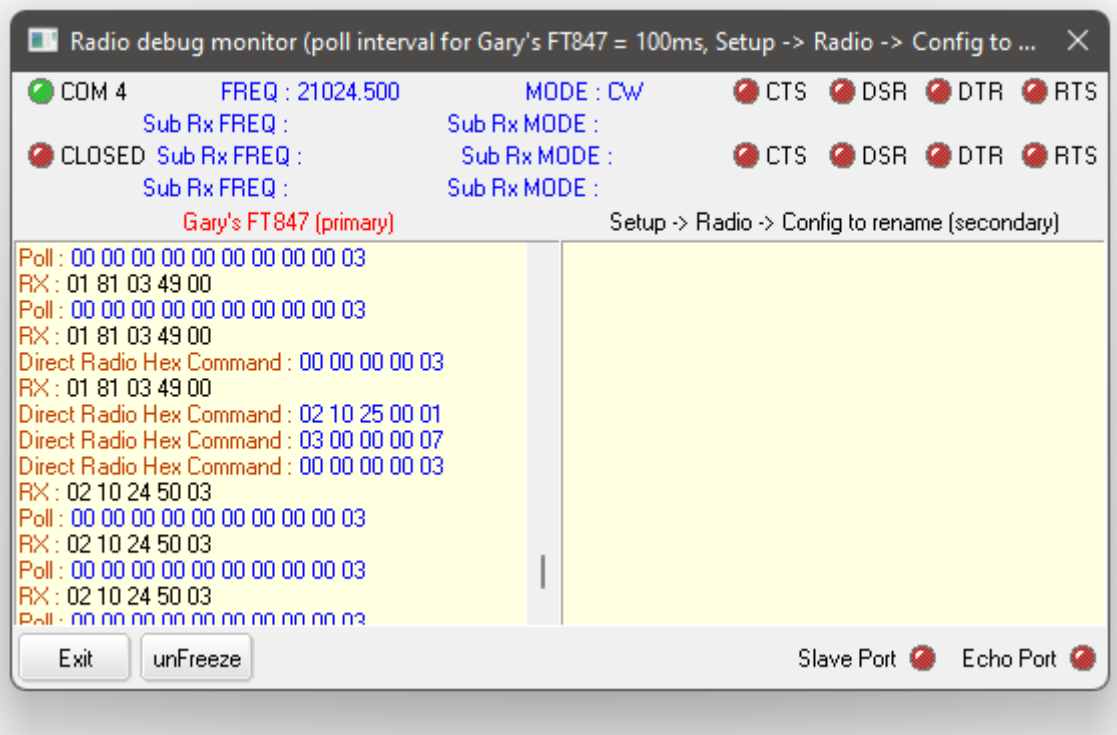
00 00 00 00 03 Query VFO A frequency and mode*

Hex response from radio

02 10 25 00 03 VFO A is on 21.025 MHz CW

* So long as the CAT port is open and connected to the radio, Logger32 automatically 'polls' the radio by sending **00 00 00 00 00** then **00 00 00 00 03** regularly as specified in the radio configuration form (*e.g*. every 100 ms). The first all-zero opcode is just padding that does nothing. The second opcode gets the radio's VFO A frequency and mode which are displayed on the log entry pane. Therefore these queries are superfluous but are shown as examples.

The radio debug window shows the hex commands and responses for my FT-847 ▼



51.3.2 CAT protocol used in newer Yaesu radios

Soon after the CATs got everywhere, Yaesu adopted the same approach as Elecraft and Kenwood *i.e*. command+parameter in plain ASCII (in CAPITALs or lower-case) with semicolon terminators. Although different models support various commands and params, here are some examples:

ASCII command sent to the radio via CAT:

IF;

ASCII response from the radio:

IFx…x; … where **x…x** is 28 bytes of data representing the radio's VFO frequency, operating mode *etc*.

Copyright © 2026 Robert C. Furzer K4CY

Command:

FA; … can optionally include a parameter to *set* the VFO A frequency;

Response:

FAy…y; … where **y…y** is 8 bytes of data representing the radio's VFO A frequency.

Command:

MD; … can optionally include a parameter to *set* the radio mode;

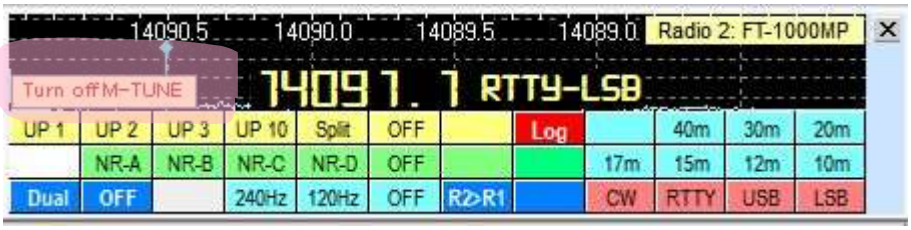
Response:

MDzz; … where **zz** is 2 bytes of coded data representing the radio's current operating mode (USB, LSB, CW *etc.*).

Hinson tip: study the Yaesu manual on your radio model for details of its CAT capabilities.

51.3.3 Yaesu MEM mode

When Logger32 polls the radio for the frequency and mode of VFO A, some Yaesu's (such as the FT-1000MP) only report the correct information while they are in VFO mode. If the radio is in MEM, M-TUNE or QMB mode, an error message (*e.g.* "Turn off M-TUNE") appears on the left of the Radio Control Panel spectrum with "**Turn off MEM**" at the bottom of the radio debug window.





51.3.4 Yaesu filter selection

Some Yaesu radios store the filter setting for each mode internally. If, in Logger32, you select a mode manually or by clicking a DX spot, the radio uses its stored filter settings.

Logger32 can set the filters according to mode by adding details into the Filter Setup window, with commands of the form:

CW=XXXXXX

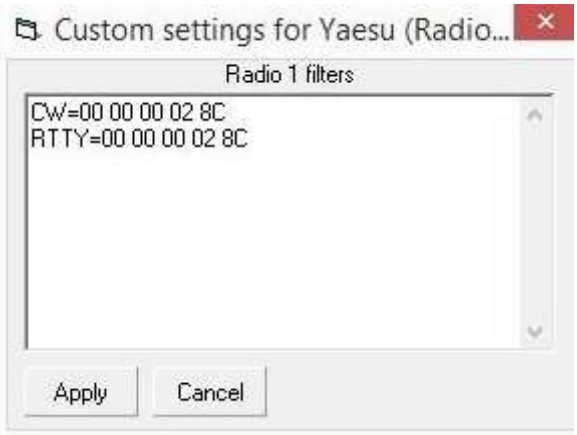
USB=YYYYYY

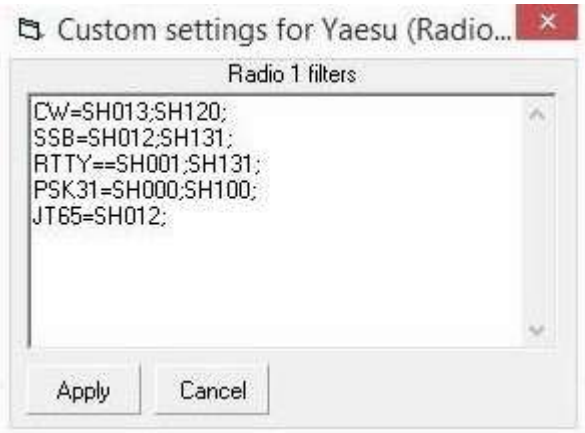
LSB=ZZZZZZ

… where XXXXXX, YYYYYY and ZZZZZZ are the respective CAT commands to set the filters. Consult your radio's CAT manual for the command codes.

Here is the setup for an FT-920 ▶

This is the syntax for older Yaesu radios using hexadecimal CAT commands, separated by spaces with no line terminator.





◀ Here is an FT-9000 example, showing the syntax for more recent Yaesu radios using ASCII Kenwood-style CAT commands.

Notice that *each and every* command ends with a semicolon terminator.

Although not required by the radio, separating the lines like this with returns make it easier for us hams to read and understand the commands.

51.4 Antenna selection for FTdx-5000, FTdx101D and FTdx101MP

Recent Yaesu radios that support the BS (**B**and **S**elect) CAT command automatically select the correct antenna socket for each band. So Logger32 doesn't need to select the antenna socket.

To take advantage of this feature initially, set each band on your radio manually to the correct antenna socket: simply pick a band then select the correct antenna for that band. Repeat for other bands. Now, when you click a DX spot, Logger32 moves your CAT-connected Yaesu to the correct frequency and the *radio* automatically selects the antenna socket associated with that band.

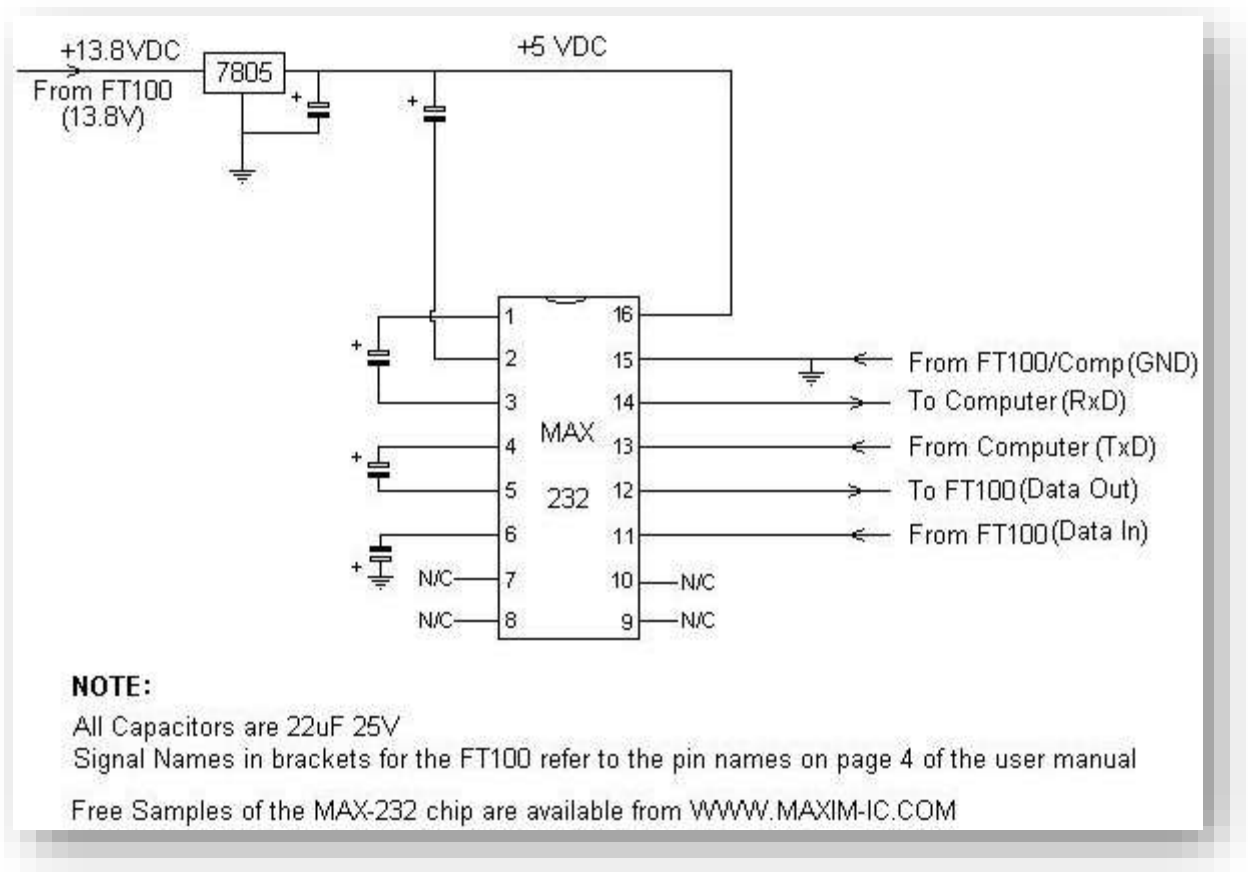
Copyright © 2026 Robert C. Furzer K4CY

51.5 Yaesu FT-100

The procedure for interfacing the FT-100 is relatively straightforward, provided you have a level converter to interface the computer's RS232 signals to the FT-100's TTL signals – either the Yaesu CT-62 cable or a homebrewed equivalent …

51.5.1 FT-100 CAT interface

Here is an interface circuit using just a handful of components.

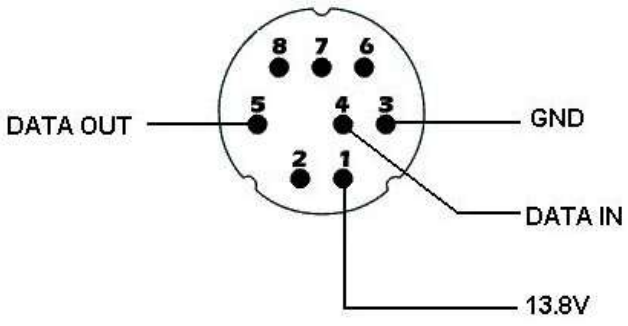


13.8 volts for the level converter can be obtained from the pigtail on the FT-100 but be very careful that the 13.8v line is never shorted to ground, as there is no separate fuse on this line! If you are not comfortable with this, use a separate power supply … or fuse the line.

51.5.2 FT-100 8-pin mini-DIN connector

An 8-pin mini-DIN male connector is required for the female CAT socket on the FT-100 ▶

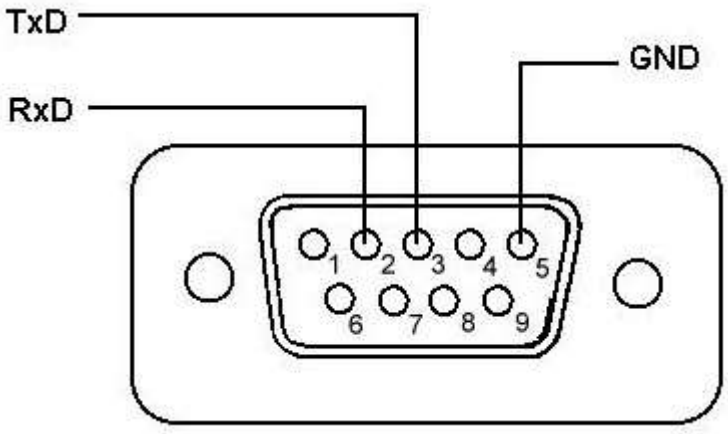
These are the connections viewed **from the wiring (back) side of the plug** (*not* the pin side!). The pin names correspond with the bracketed names on the interface circuit diagram above.



Copyright © 2026 Robert C. Furzer K4CY

51.5.3 DB-9 PC serial connector

A *female* DB-9 connector is typically required at the computer end of the cable to fit the *male* DB-9 RS232 connector on the PC's rear panel (if it has one – few do these days). These are the connections viewed from the **wiring (back) side** (*not* the pin side!) ▼



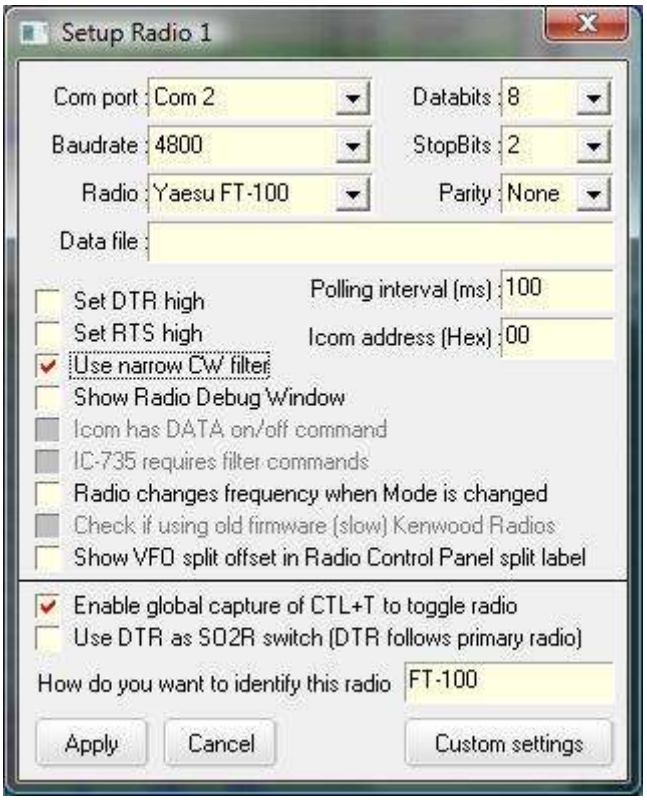
The pin names shown here correspond with the bracketed names on the circuit diagram above.

Hinson tip: if your eyesight and lighting are good, look closely for tiny pin numbers molded into the plastic form on the connector's backside. An Anglepoise-style magnifier with a ring light works for me.

51.5.4 FT-100 CAT configuration

No radio setup is required other than ensuring that the pigtail connector has its default setting of CAT/TUNER (set by an internal hardware switch within the FT-100: see the FT-100 user manual).

In Logger32, use **Setup ⇨ Radio ⇨ Radio 1|2 configuration** to set up the CAT connection ▶



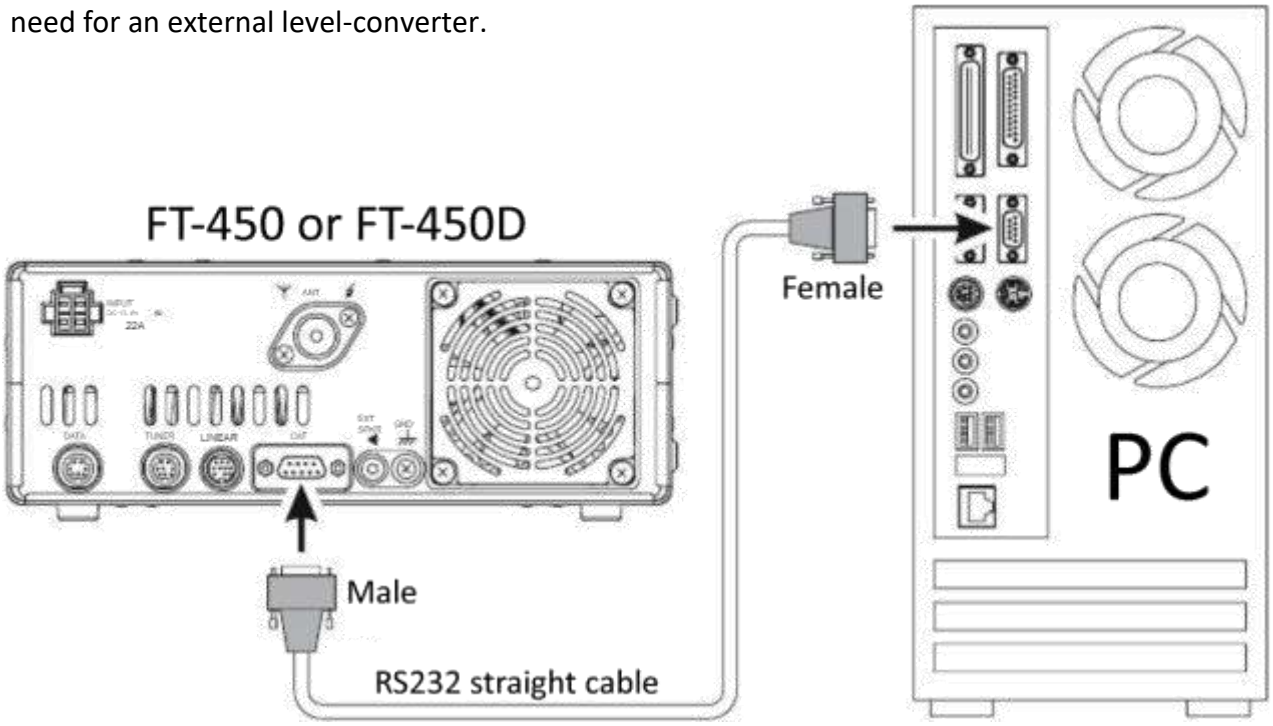
The suggested value of 100ms for polling interval is a starting point. Reducing the value of the polling interval (increasing the rate of radio polling) speeds up and smooths Logger32's display of band, frequency and mode changes, but may be unreliable or not work at all on your system, whereas increasing the interval gives the PC and radio more of a chance to communicate reliably.

Copyright © 2026 Robert C. Furzer K4CY

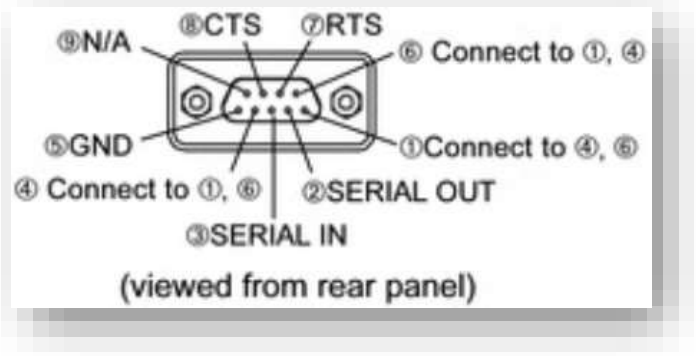
51.6 Yaesu FT-450 & FT-450D

For instructions on '450s CAT, consult the FT-450 operator manual and the separate CAT manual, most of which consists of tables describing the radio's rich complement of CAT commands.

A level-converter is built-in to the radio for normal RS232 PC serial port voltages so there is no need for an external level-converter.



▲ The radio's rear panel CAT connector is a 9-pin *female* DE-9 D-sub socket ▶



If your PC has a free RS232 serial (COM) port (usually a 9-pin *male* DE-9 D-sub connector on the rear panel), plug-in a straight-through RS232 cable (*not* the crossover/null-modem type) fitted with female and male connectors.

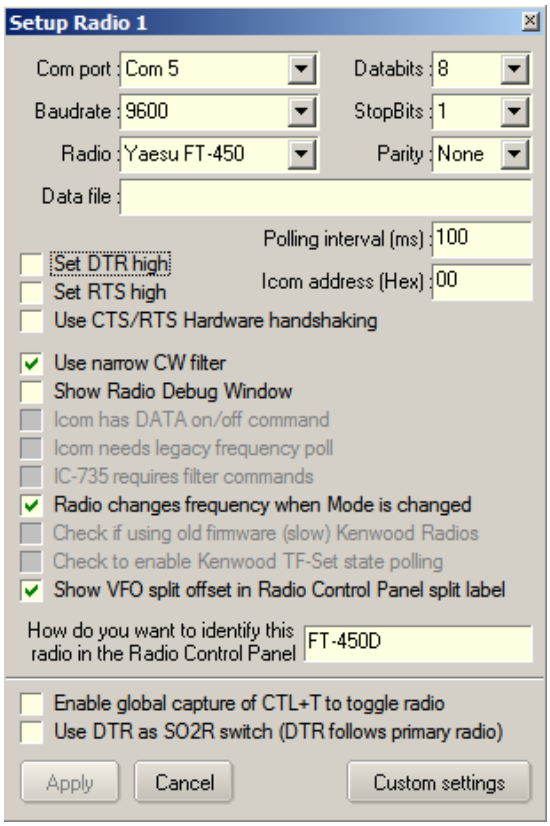
If your PC has no RS232 ports available but a free USB port, a USB-serial adapter *should* work OK provided it is not a knock-off using a counterfeit Prolific-clone chip, provided the driver is supported by your version of Windows, and provided it has a female connector to mate with the radio's male CAT connector. If all your USB ports are in use, try adding a USB hub.

Copyright © 2026 Robert C. Furzer K4CY



◀ USB-RS232 CAT cables are available through online retailers and shops. However, if it has a *female* 9-pin socket like this one, you will also need a 9-pin D-sub straight-through back-to-back male-male ▶ gender-changer adapter to fit the radio's female CAT socket. If the cable came from some dubious Far Eastern source, you still face the possibility that the chip/driver may not actually work with your Windows PC. Good luck claiming a refund!





If your PC has no free USB or RS232 ports, you *may* find an unused 2x5-pin 0.1" COM port male header on the PC motherboard into which you can plug an adapter cable for a full- or half-height rear-panel RS232 serial port connector like this ▶



◀ In Logger32, configure the radio using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**.

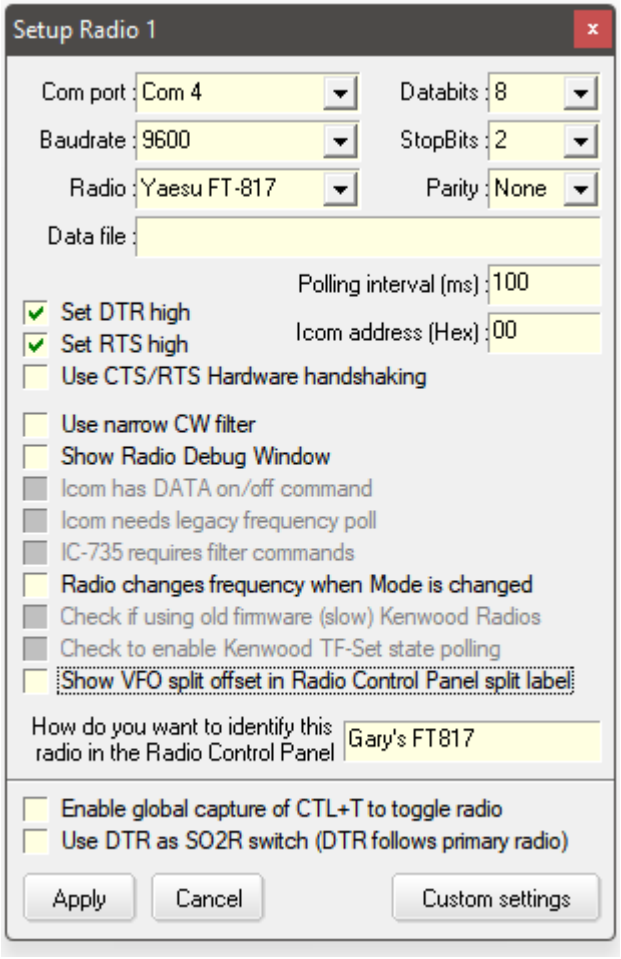
Choose the appropriate Com port and CAT speed for your radio: look for "CATRATE" on the radio's menu, with "CAT TOT" set to 10 and "CAT RTS" set to DISABLE. You *may* need to increase the polling interval for a stable CAT connection on a slow PC.

Give the radio a suitable name/tag to identify it in the **R**adio **C**ontrol **P**anel and elsewhere.

Copyright © 2026 Robert C. Furzer K4CY

51.7 Yaesu FT-817

Using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**, the best settings found during testing for an FT-817 were 9600, 8 databits, 2 stopbits and parity none with a polling interval of 100 milliseconds ▶



If the CAT interface does not seem to work, check that menu option 14 on the radio is set to 9600 baud. The default speed is 4800 baud.

You probably only need to tick <**Set DTR high**> and/or <**Set RTS high**> if your CAT cable uses these lines to power the converter chip.

If you wish to use the FT-817 for PSK, choose PSK-L in menu option 26 (Dig Mode). When in digital mode, the FT-817 normally selects LSB but does not tell the PC if the IF filter is set to PSK-U, putting the radio in USB – in which case the frequency ribbon in Logger32's sound card data window runs backwards and functions such as <**Align**> don't work correctly.

A side note about Yaesu radio names

Several Yaesu radios include "DX" in smaller letters as part of the name printed on the front panel *e.g.* ▶



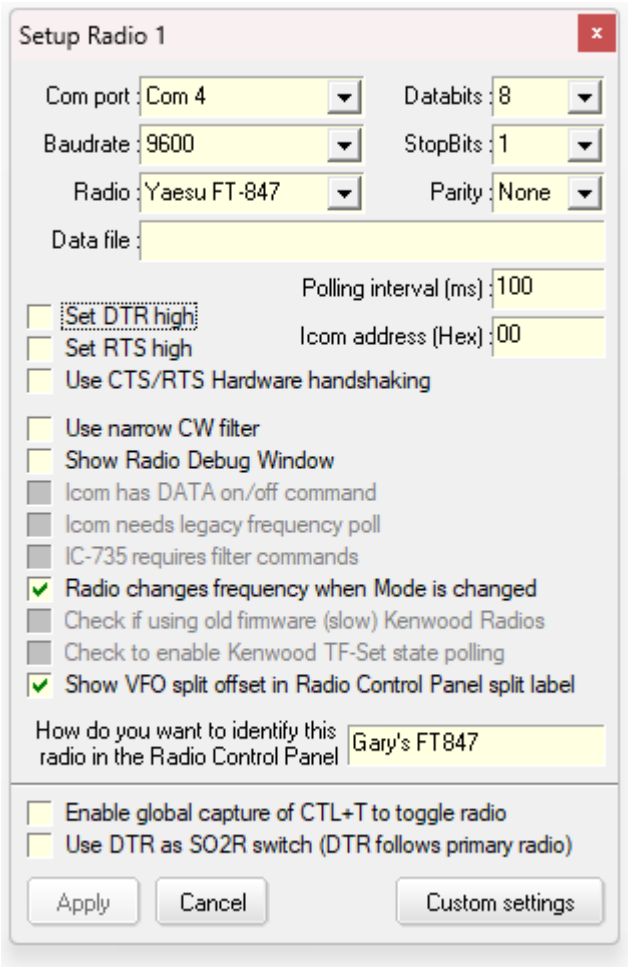
Presumably, radios *without* the embedded "DX" don't have as much DX in them.

Furthermore, Yaesu often adds supplementary letters to the base name of updated models to indicate their additional capabilities, and justify the associated price hike. To be fair, Yaesu is definitely not alone in this. My "GTi" car has go-faster stripes.

Logger32 doesn't have the stomach for such frivolities. Logger32 names the Yaesu radios simply, mostly with hyphens, without changes of font size and often without the supplementary model-variant letters. If you have, say, a "Yaesu FTDX999XXL" sitting in front of you in the shack, it *may* be titled "FT-999" in Logger32's internal list of radio type – if indeed it is directly supported. If not, well try a 'similar' listed Yaesu and see how it goes.

Copyright © 2026 Robert C. Furzer K4CY

51.8 Yaesu FT-847



Yaesu's "Earth Station" covers topband to 70cm, and is designed for satellite comms, with full duplex *cross*band operation (transmit on the uplink band *while* listening on the downlink band), Doppler tracking, and 4 separate antenna ports with PTT outputs for HF (100W), 6m (100W), 2m (50W) and 70cm (50W, N-type). It also supports 1200/9600 bps VHF/UHF FM packet radio with a TNC, EME and QRQ CW for meteor scatter.

For CAT, the radio's rear panel has a DB9 9-pin RS232 *male* connector with built-in level converter: lacking a crossover RS232 cable and RS232 port on my PC, I made a simple back-to-back RS232 female to RS232 female *crossover* adapter for a conventional, cheap USB-to-RS232 adapter.

◀ Setup the CAT configuration using **Setup ⇨ Radio ⇨ Radio 1|2 configuration**.

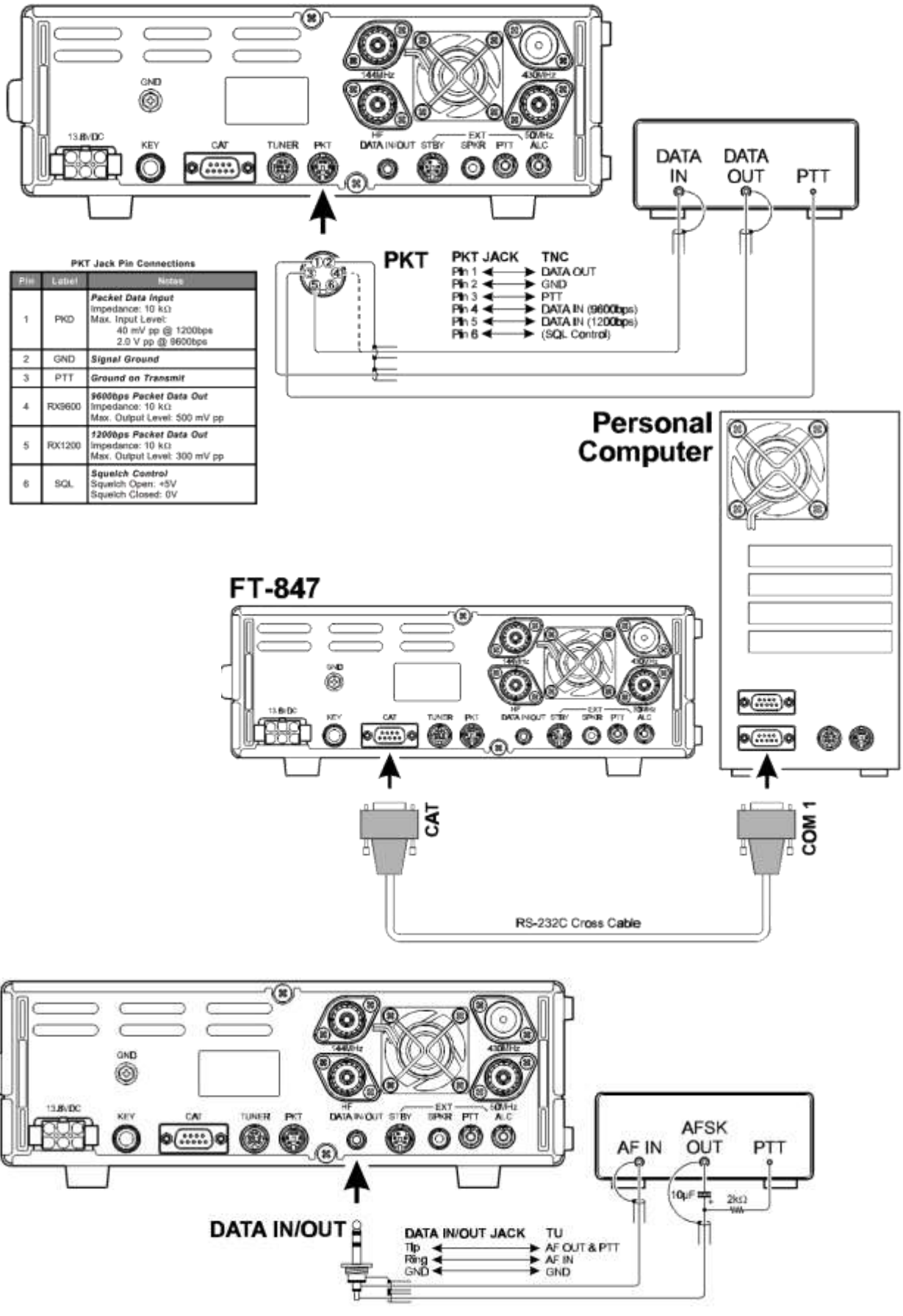
The baudrate setting in Logger32 *must* match the radio's CAT RATE setting (menu item 37) *i.e*. 4800, 9600 or (less reliably) 57600 bps. "CAT" appears on the right of the radio's front panel display while the CAT is connected and working.

Tick <**Use narrow CW filter**> to enable the rig's NAR setting automatically when you click a CW spot. Or not – your choice.

There is a separate rear-panel 6-pin mini-DIN connector for packet radio, and a 3.5mm stereo socket for AFSK audio input+output and PTT. Using the 3.5mm PTT connection enables the rear panel audio input and disables the front panel microphone. A 1:1 audio transformer may reduce hum (for inspiration, see the FT-920 sound card interface circuit below).

The early CAT interface of the FT-847 is distinctly limited, offering just 13 functions. It does not support PC control of split, RIT or XIT, nor A⇆B transfers. More info below.

Copyright © 2026 Robert C. Furzer K4CY



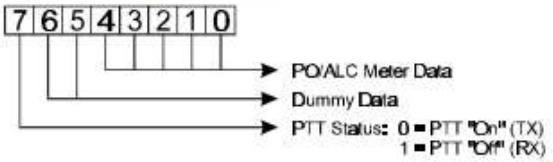
PKT Jack Pin Connections

Pin	Label	Notes
1	PKD	***Packet Data Input*** Impedance: 10 kΩ Max. Input Level: 40 mV pp @ 1200bps 2.0 V pp @ 9600bps
2	GND	***Signal Ground***
3	PTT	***Ground on Transmit***
4	RX9600	***9600bps Packet Data Out*** Impedance: 10 kΩ Max. Output Level: 500 mV pp
5	RX1200	***1200bps Packet Data Out*** Impedance: 10 kΩ Max. Output Level: 300 mV pp
6	SQL	***Squelch Control*** Squelch Open: +5V Squelch Closed: 0V

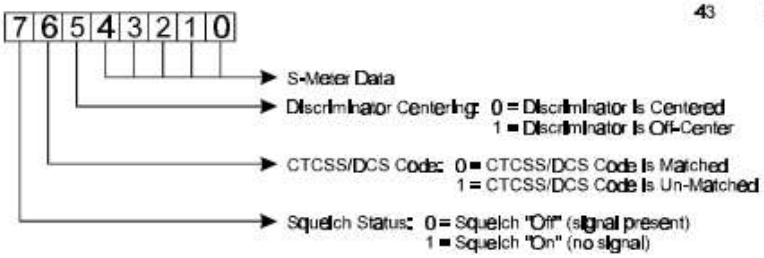
Opcode Command Chart

Comand Titel	Parameters				Opcode	Comments
CAT On/Off	※	※	※	※	P1	P1=00: CAT On P1=80: CAT Off
PTT On/Off	※	※	※	※	P1	P1=08: PTT On (TX) P1=88: PTT Off (RX)
Satellite On/Off	※	※	※	※	P1	P1=4E: Satellite Mode On P1=8E: Satellite Mode Off
Set Frequency	①	②	③	④	P1	①~④ Frequency Digits: 43, 21, 00, 00 = 432.1000 MHz P1=01: Set to MAIN VFO P1=11: Set to SAT RX VFO P1=21: Set to SAT TX VFO
Operating Mode	D1	※	※	※	P1	D1=00: LSB, D1=01: USB, D1=02: CW, D1=03: CW-R, D1=04: AM, D1=08: FM, D1=82: CW(N), D1=83: CW-R(N), D1=84: AM(N), D1=88: FM(N) P1=07: Set to MAIN VFO P1=17: Set to SAT RX VFO P1=27: Set to SAT TX VFO
CTCSS/DCS Mode	D1	※	※	※	P1	D1=0A: DCS On D1=2A: CTCSS ENC/DEC On D1=4A: CTCSS ENC On D1=8A: CTCSS/DCS Off P1=0A: Set to MAIN VFO P1=1A: Set to SAT RX VFO P1=2A: Set to SAT TX VFO
CTCSS Frequency	D1	※	※	※	P1	D1=00h~3Fh (Tone Frequencies per chart on page ??) P1=0B: Set to MAIN VFO P1=1B: Set to SAT RX VFO P1=2B: Set to SAT TX VFO
DCS Code	①	②	※	※	P1	①, ② from the DCS Code # (i.e. 07, 54=DCS Code 754) P1=0C: Set to MAIN VFO P1=1C: Set to SAT RX VFO P1=2C: Set to SAT TX VFO
Repeater Shift	D1	※	※	※	09	D1=09: "Minus" Shift D1=49: "Plus" Shift D1=89: Simplex
Repeater Offset	①	②	③	④	F9	①~④ set the Repeater Shift: 00, 50, 00, 00 = 5MHz Shift
Receiver Status	※	※	※	※	E7	S-Meter, Squelch, etc. (Note 1)
Transmit Status	※	※	※	※	F7	PO Meter, PTT, etc. (Note 2)
Frequency & Mode Status	※	※	※	※	P1	P1=03: Read MAIN VFO Frequency & Mode Status P1=13: Read SAT RX VFO Frequency & Mode Status P1=23: Read SAT TX VFO Frequency & Mode Status (Note 3)

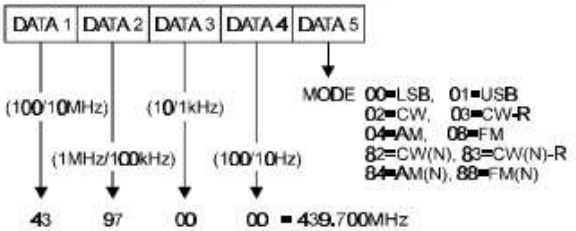
Note 1: Receiver Status



Note 2: Transmit Status



Note 3: Frequency & Mode Status



Copyright © 2026 Robert C. Furzer K4CY

51.8.1 FT-847 CAT command format

Like other early CAT radios, the FT-847 has only a basic CAT capability with few commands[474].

FT-847 CAT commands and responses are sent as blocks of 5 bytes each spaced by up to 200 ms. The 5 hex data bytes are a CAT command (one of 25 opcodes controlling 25 radio functions – see the CAT chapter in the FT-847 Operating Manual for details) and up to 4 parameters[475] … except they are sent in reverse order, vaguely reminiscent of Reverse Polish Notation ▼

One CAT data block								
Byte 1	Max 200 ms	Byte 2	Max 200 ms	Byte 3	Max 200 ms	Byte 4	Max 200 ms	Byte 5
4th parameter		3rd parameter		2nd parameter		1st parameter		Opcode

Logger32 handles the timing, formatting, sending and receiving of each byte, so all we need to do is compose the 5-byte block in the required reverse order using Logger32's ***$HexCommand*** *$* format – for example:

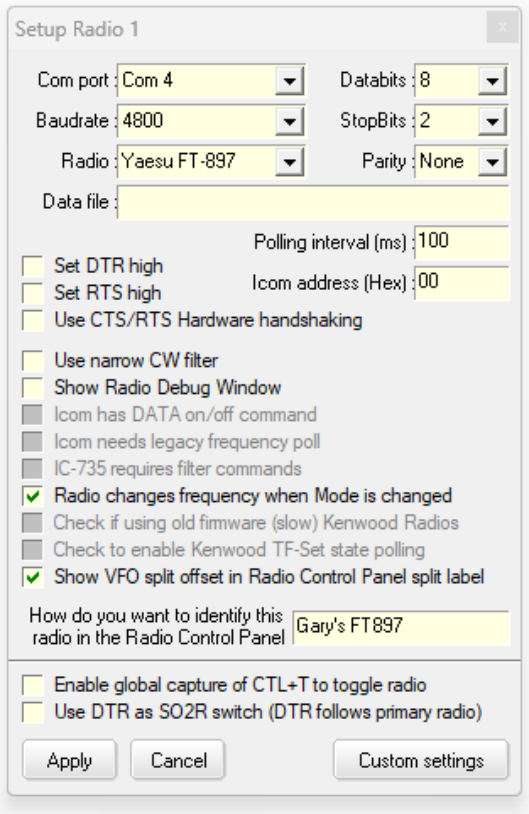
- ***$HexCommand 02 00 00 00 07$*** Set main VFO to mode CW
- ***$HexCommand 00 00 00 00 4e$*** Enable SAT (dual-band VHF/UHF)
- ***$HexCommand 02 00 00 00 17$*** Set SAT RX VFO to mode CW
- ***$HexCommand 02 00 00 00 27$*** Set SAT TX VFO to mode CW

51.9 Yaesu FT-897

On the radio's menu, set:

- CAT baud rate to 4800
- CAT/LIN/TUN to CAT

In Logger32, configure the CAT port like this using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** ▶



As with the FT-847, the FT-897 uses hexadecimal CAT commands, so use $**HexCommand** rather than $**Command** in your macros. Semicolon terminators are *not* needed after each HexCommand.

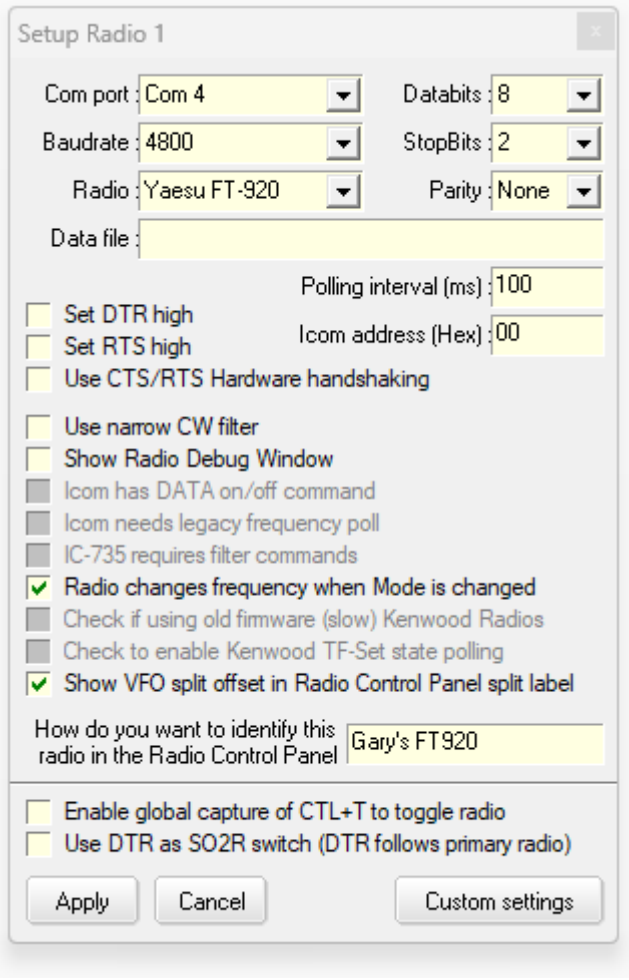
[474] CAT cannot be used at all while the matching FC-20 ATU is connected. Presumably, the radio's stone-page processor that normally handles CAT is tied up communicating with the ATU, passing frequency information to and receiving status responses from the ATU in much the same way.

[475] Use any value to pad the block out to 5 bytes: 00's are conventional and obvious padders.

Copyright © 2026 Robert C. Furzer K4CY

51.10 Yaesu FT-920

The FT-920 has a conventional DB-9 RS232 rear panel connector and built-in level converter for CAT control at 4800 baud.

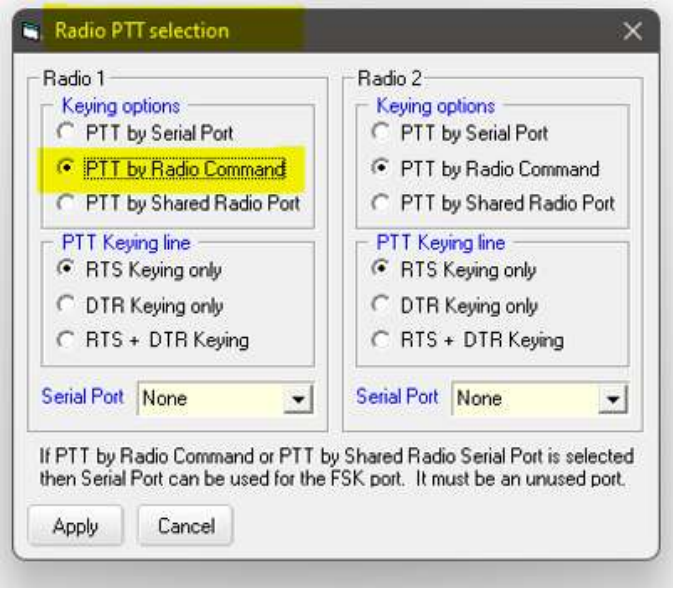


Provided you select the correct COM port, these settings should work using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** ▶

Adjust the polling rate to your liking.

CAT, PTT and CW functions can all use a single shared PC serial port. An external interface is needed for software CW keying, plus another for PTT unless you are using CAT.

PTT control: in the Sound card data window, open **Settings ⇨ Radio PTT options** ▼

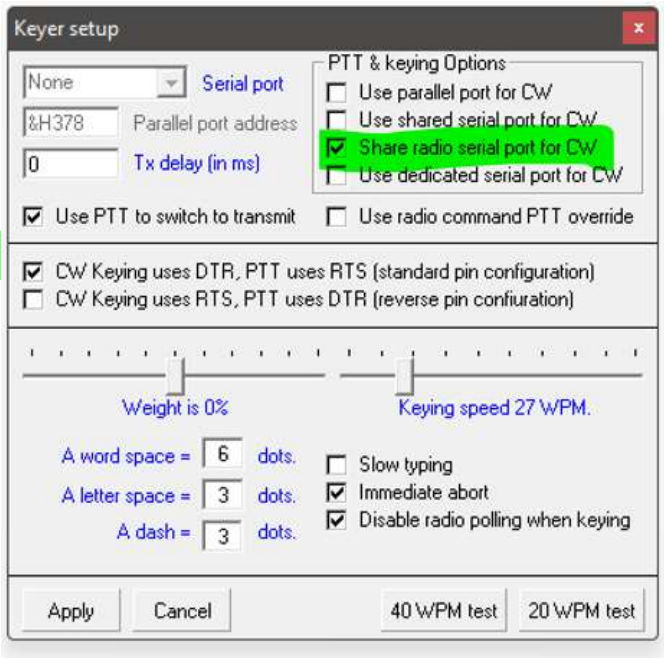


This is the simplest setup, with **<PTT by Radio Command>** *i.e.* Logger32 sends the *Transmit* and *Receive* CAT commands to the active radio at the appropriate times.

If you are *not* using CAT, you may use VOX or a hardware interface for PTT – a simple transistor switch will suffice, connected to the RTS line on a serial port.

Copyright © 2026 Robert C. Furzer K4CY

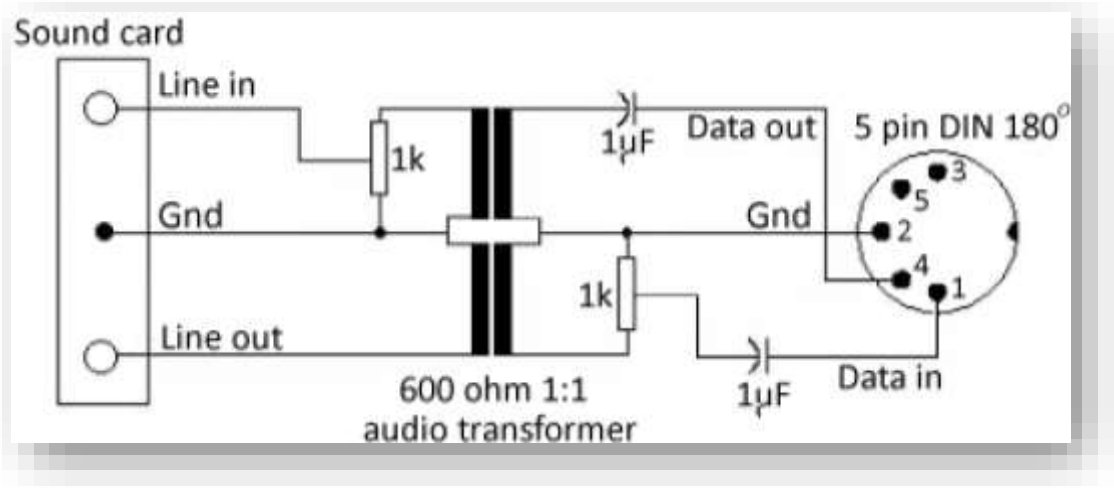
CW keying: open the CW Machine click <**Config**> then <**Keyer setup**> ▶



If you have CAT capability, the same serial port can also be used for both CAT and software CW keying. Tick <**Share radio serial port for CW**> on this form and connect your keying interface (a transistor switch) to the DTR line on the CAT port. Otherwise, enter the address of the serial or parallel port you intend to use for CW keying, and connect your keying interface there, ticking <**Use shared serial port for CW**> instead.

51.10.1 FT-920 interfaces and settings

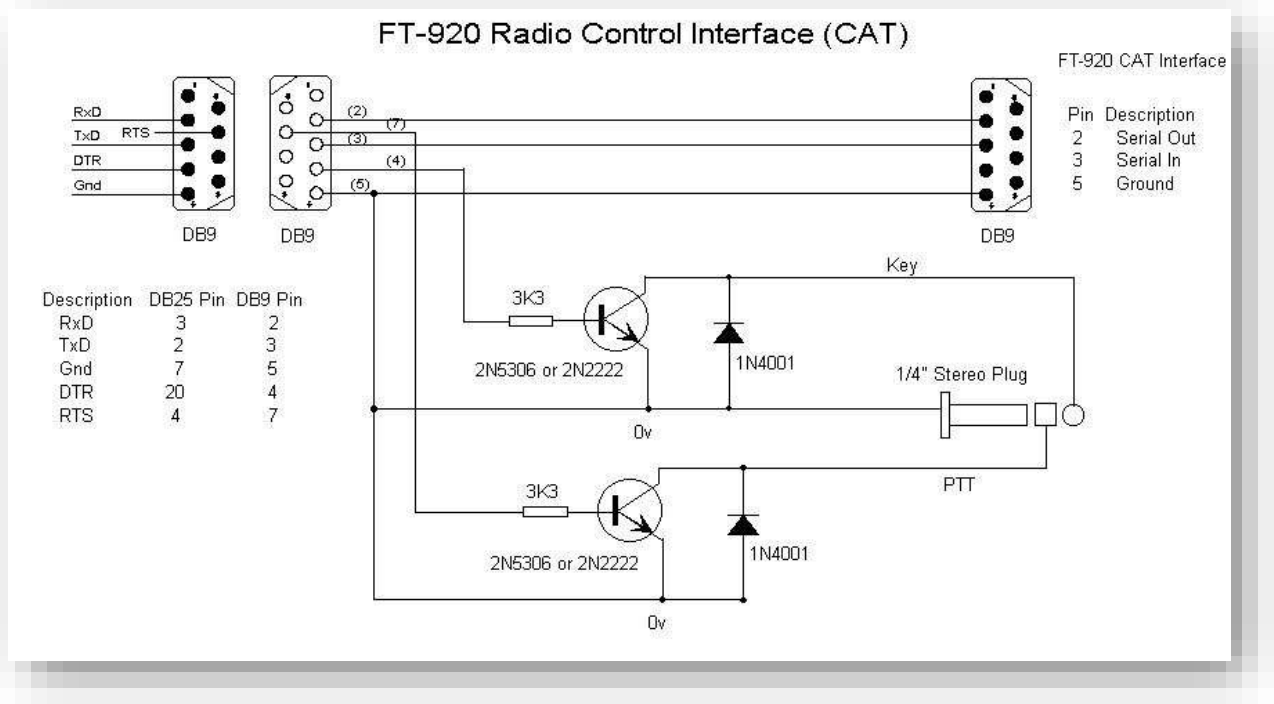
- **Sound card Interface**: any commercial sound card interface should work on the FT-920 with appropriate wiring. Audio to/from a built-in PC or external USB sound card *can* be connected directly to the rear panel DIN connector, however an audio transformer will reduce mains hum. The following very simple interface design isolates the computer from the radio using a 600Ω 1:1 audio isolation transformer, with two 1kΩ potentiometers for level adjustments and two capacitors (value non-critical: 1µF, 1.5µF or 2.2µF should work, preferably non-polarized, low voltage) ▼



Hinson tip: there is no need to connect the PTT on DIN pin 3. If you are using CAT, Logger32 sends the CAT commands to go into transmit before sending a message and return to receive afterwards. If you are *not* using CAT, the radio's VOX should achieve the same effect.

Copyright © 2026 Robert C. Furzer K4CY

- **CAT, CW and PTT interface**: CAT, CW and PTT functions can all share the same serial port ▼



- **Radio settings for sound card operation**: the AFSK-FSK slide switch on the back of the radio should be in the AFSK position. Select the desired SSB mode (USB or LSB) and then press the DATA button to disconnect the microphone and activate the DATA connection. Follow normal procedures for setting up your power levels and sound card levels. See the Sound card data window for details.
- **Radio settings for CW and PTT operation**: to use the "KEY" connection on the back of the radio for CW and PTT, you must set two switches for operation. The PDL-KEY switch on the rear of the radio should be set to "KEY" for PTT functions, activating the PTT line for all modes. The "KEYER" switch on the front panel must be "ON" for CW operation.
- **Radio settings for RTTY/FSK operation**: the FT-920 can be set up to transmit FSK signals using the Sound card data window and MMTTY/MMVARI. Received RTTY signals will be decoded by MMTTY using the audio output from the radio. Transmissions will use the FT-920's internal FSK tone generators, with FSK data keying provided by MMTTY.

  You need to dedicate a parallel (LPT) or serial (COM) port solely for FSK keying: you cannot operate FSK using the same COM port that is used for CAT radio control. You also need a simple transistor interface circuit for the FSK keying line. See the FT-920 manual for appropriate hook-up. The transmit and receive signals and PTT can be connected to the DATA terminal on the back of the radio.

  Suggested FSK FT-920 settings for the Sound card data window:

  - The AFSK-FSK slide switch on the back of the radio must be in the "FSK" position. You must have an FSK interface.
  - FT-920 MODE = DATA-LSB
  - RTTY FSK mode configures the radio for internal FSK tone generation. This mode can be operated with LSB or USB tone injection.

Copyright © 2026 Robert C. Furzer K4CY

- Menu U-42 = HI 2125 L. This sets the high tone to 2125 Hz with LSB injection. Use this setting to align the frequency scale in the sound card window with the radio.
- Menu U-43 = shift 170. Default RTTY shift
- Menu U-44 = Normal. Used for LSB operation
- Menu U-44 = Reverse. Reverses the Mark and Space keying - this will configure the FT-920 transmissions for USB signals.

- **Sound card settings for RTTY FSK operation**: *EXTFSK.DLL* is an MMTTY port driver for FSK. Put the file into the Logger32 folder. This driver allows you to select the port and data line for FSK keying. It also allows you to reverse the data keying if needed by your interface, in addition to the FT-920 menu U-44 item, and MMVARI has a menu option to reverse the FSK keying under **Settings ⇨ MMVARI settings ⇨ RTTY keying**. If it's too confusing, find a friend to check out your signal on-air, tell you if it is normal/reversed RTTY, and confirm the levels are OK.

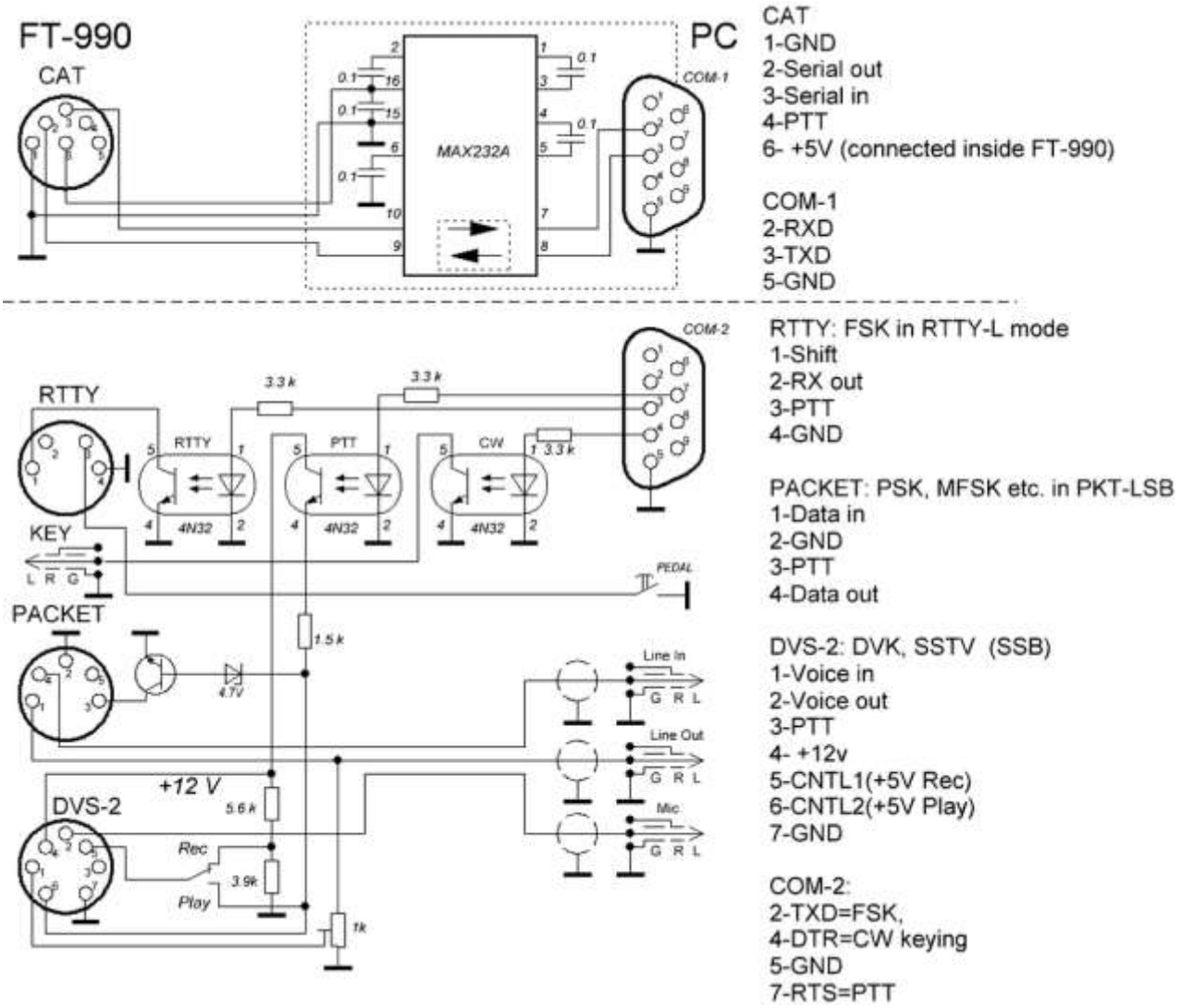
  For specific Sound card data window settings for FSK operation, see here.

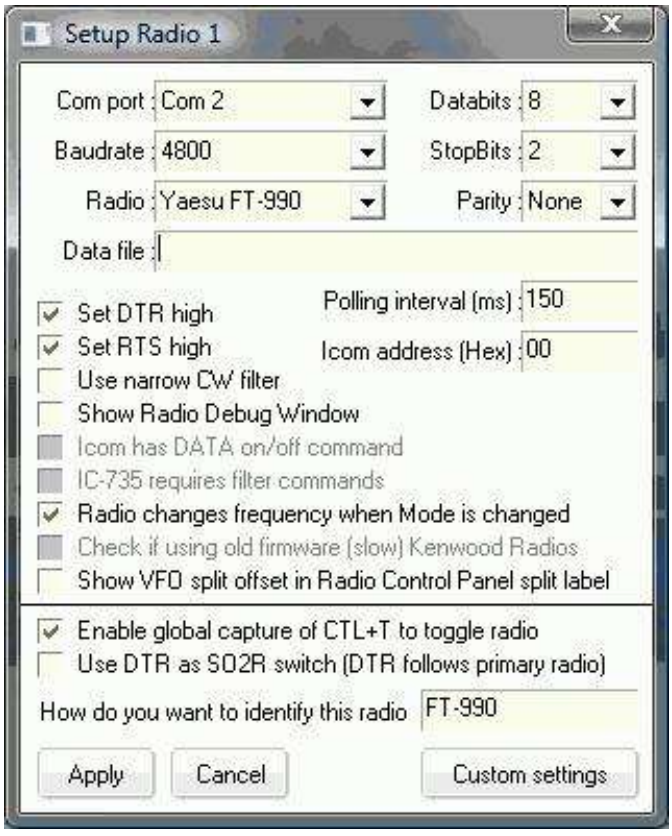
51.11 Yaesu FT-990

51.11.1 FT-990 PC interface (UA1AJW design)

The FT-990 needs ROM version 1.3 or later to work properly with Logger32. Its CAT port is a 6-pin DIN on the rear panel using TTL levels (0V or +5V), which means it needs a level-converter interface for the PC's RS232 levels. The MAX232A chip makes it easy.

Copyright © 2026 Robert C. Furzer K4CY





◀ Using **Setup ⇨ Radio ⇨ Radio 1|2 configuration** set the FT-990 CAT connection to 4800 baud with 8 data bits, two stop bits and no parity.

Set DTR|RTS High: if you are using a CAT interface that needs power, tick to assert either or both DTR or RTS lines to supply *limited* power to the interface. If your interface has a separate power cable (which the IF-232C does), *un*tick both options, releasing the DTR and RTS lines for other purposes (such as PTT control and software CW keying).

Copyright © 2026 Robert C. Furzer K4CY

If your FT-990 is not communicating at all with Logger32, check that the CAT cable is connected to the correct sockets and everything is powered up and switched on as it should be. Check also that you have the correct COM port selected and the baudrate is set to 4800 specifically.

You may need to adjust the polling rate to establish reliable and reasonably responsive CAT comms. Too fast and the PC or radio may not keep up. Too slow and Logger32's radio frequency display will be sluggish, lagging well behind the spinning VFO knob on the radio: something in the range 150 to 300 ms should work OK (fingers crossed!).

If the radio receives and actions CAT commands from Logger32 but does not send information (such as its VFO frequency and mode) back, it *may* be that your CAT interface does not sufficiently load the transceiver's data output data line. The FT-990 CAT connection has a source driver data output which is best pulled low through a small 1.5k Ohm resistor between pin 1 (ground) and pin 2 (serial output) of the radio's DIN6 CAT plug.

51.11.2 FT-990 CAT command format

CAT commands and responses are sent as blocks of 5 bytes up to 200 ms apart. The 5 hex data bytes are a CAT command (one of 25 opcodes controlling 25 radio functions – see the CAT chapter in the FT-990 manual for details) and up to 4 parameters[476] … except they are sent in reverse order:

One CAT data block								
Byte 1	Max 200 ms	Byte 2	Max 200 ms	Byte 3	Max 200 ms	Byte 4	Max 200 ms	Byte 5
4th parameter		3rd parameter		2nd parameter		1st parameter		Opcode

51.11.3 Filter selection for the FT-990

Tick <**Use narrow CW filter**> on the port configuration form to select the radio's 500 Hz CW filter automatically whenever you click a CW DX spot. Don't if not.

The FT-990 has a CAT command (the **8C** opcode) to select filters, with a single parameter for the desired filter setting:

00 = 2.4 kHz.

01 = 2.0 kHz.

02 = 500 Hz.

03 = 250 Hz.

So a macro to select the 250 Hz filter would be:

$**HexCommand 00 00 00 03 8C**$

To select the 2.4 kHz filter:

$**HexCommand 00 00 00 00 8C**$

Macros for other filters follow the same pattern.

The FT-990 has IF-shift controls to move the passband center frequency, but there are no CAT codes for this.

[476] Use any value to pad the block out to 5 bytes: most of us prefer 00's for padders.

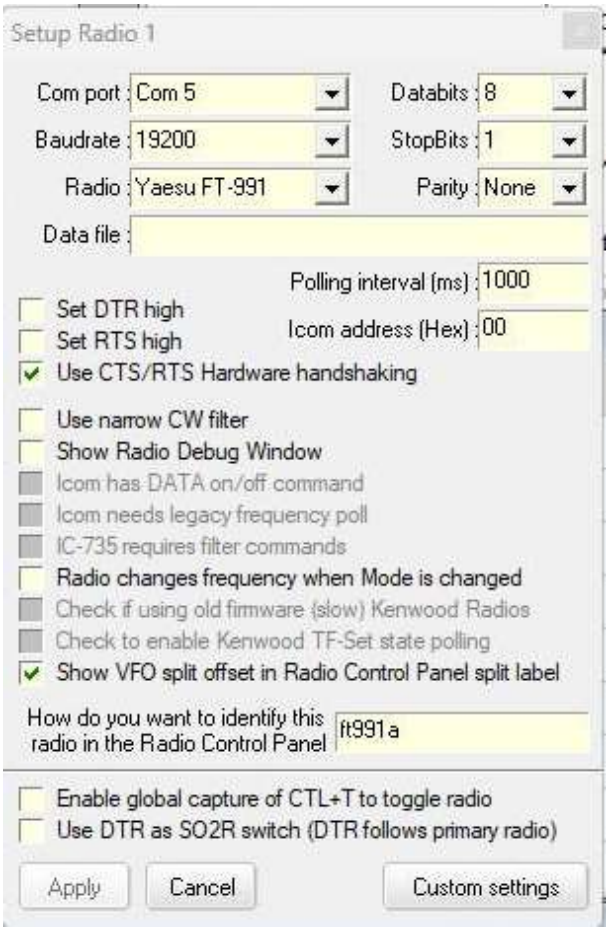
Copyright © 2026 Robert C. Furzer K4CY

If you send a CAT command sequence to select a filter that is not physically installed in the radio, it is simply ignored – no harm done.

The FT-990 CAT function is hungry enough to accept one or two commands on a single macro line *i.e*. $**HexCommand** <space> *5 or 10 hex bytes*$.

51.12 Yaesu FT-991A and FT-891

PE1RF reports that this CAT port setup works for his FT991A ▶



You need to load the radio's CAT driver from Yaesu on your PC. Consult the relevant CAT manual *e.g*. the FT-991A CAT manual for details. Having downloaded and installed the driver correctly, and with the radio connected to your PC via a USB lead and powered-on, *your* CAT port may not be numbered Com5 in Windows.

The radio's CAT data rate *must* be the same as the baudrate used in the port setup Logger32 for the two to communicate: adjust it using the radio's menu and/or the port setup form. 19200 baud is probably OK but if not, lower rates will make little difference in practice.

Other radio menu settings may also need changes too. Check the radio's user manual for info.

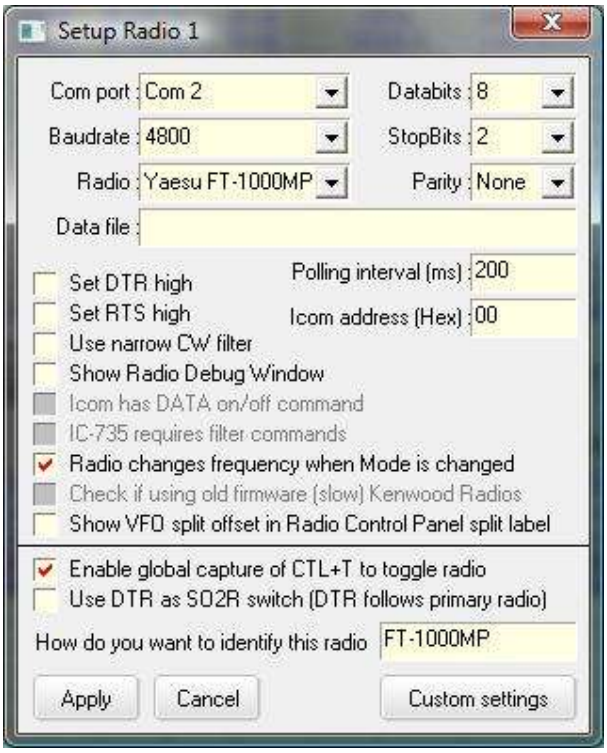
Depending partly on the speed/capacity of the PC, the 1000 ms <**Polling interval**> could probably be reduced, maybe as far as 250 ms or below, without the CAT connection becoming unreliable. That will improve the responsiveness of the logbook entry page frequency and mode display, plus the marker on the band map, as you tune your FT991's VFO knob or change band or mode on its front panel. It will be less 'jumpy'.

Copyright © 2026 Robert C. Furzer K4CY

51.13 Yaesu FT-1000MP

The FT-1000MP has a built-in level-converter, allowing direct connection from the rear panel CAT connector to an RS232 serial port on your computer (or an RS232-to-USB adapter) without the need for any external boxes.

Set up the CAT connection using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**



Adjust the polling rate to your liking. A polling interval of 200 or 300 milliseconds usually works reliably but some systems can poll faster, down to about 100 ms. “System” here refers to the combination of radio, CAT connection, PC and Logger32.

51.13.1 Codes for filter selection for the FT-1000MP

The FT-1000MP has two IF filters that can be controlled, one at 8.215 MHz and the other at 455 kHz. Here are some examples for filter selection for the FT-1000:

$**HexCommand 02 00 00 03 8C**$	Sets the 455 kHz filter to 250 Hz.
$**HexCommand 01 00 00 03 8C**$	Sets the 8.215 MHz filter to 250 Hz.
$**HexCommand 02 00 00 02 8C**$	Sets the 455 kHz filter to 500 Hz.
$**HexCommand 01 00 00 02 8C**$	Sets the 8.215 MHz filter to 500 Hz.
$**HexCommand 02 00 00 00 8C**$	Sets the 455 kHz filter to 2.4 kHz.
$**HexCommand 01 00 00 00 8C**$	Sets the 8.215 kHz filter to 2.4 kHz

The FT-1000MP has IF-shift controls to move the passband center frequency when the narrow IF filters are selected, but there are no opcodes for this. With the IF-shift control in its central position, the nominal passband is centered on 1 kHz in USB and 2 kHz for Packet.

The FT-1000MP does however have opcodes to set the EDSP, with which you can experiment. Some examples are given below.

51.13.2 HP filter (USB)

The upper frequency of the filter remains fixed on 2300 Hz but the lower frequency may be changed to reduce the bandwidth as follows:

$**HexCommand 00 00 50 42 75**$	Produces a filter with the range 1500-2300 Hz.
$**HexCommand 00 00 60 42 75**$	Produces a filter with the range 1800-2300 Hz.
$**HexCommand 00 00 67 42 75**$	Produces a filter with the range 2050-2300 Hz.

Copyright © 2026 Robert C. Furzer K4CY

51.13.3 LP filter (USB)

The lower frequency of the filter remains fixed on 2 kHz and the upper frequency may be changed to reduce the bandwidth as follows:

$**HexCommand 00 00 90 41 75**$ Produces a filter with the range 2000-2700 Hz.

$**HexCommand 00 00 80 41 75**$ Produces a filter with the range 2000-2500 Hz.

$**HexCommand 00 00 70 41 75**$ Produces a filter with the range 2000-2250 Hz.

$**HexCommand 00 00 00 40 75**$ Switches off the EDSP.

These macros have no effect in packet mode on the FT1000MP.

51.13.4 Combining commands in a macro

Control codes can be combined to perform a complex procedure in one macro in one of two ways. Either each command is placed on a separate line, or they can be combined on one line as shown below. Both examples set USB with the 2.4 kHz filter

$**HexCommand 00 00 00 01 0C**$ Sets USB

$**HexCommand 00 00 00 00 8C**$ Sets the 2.4 kHz filter

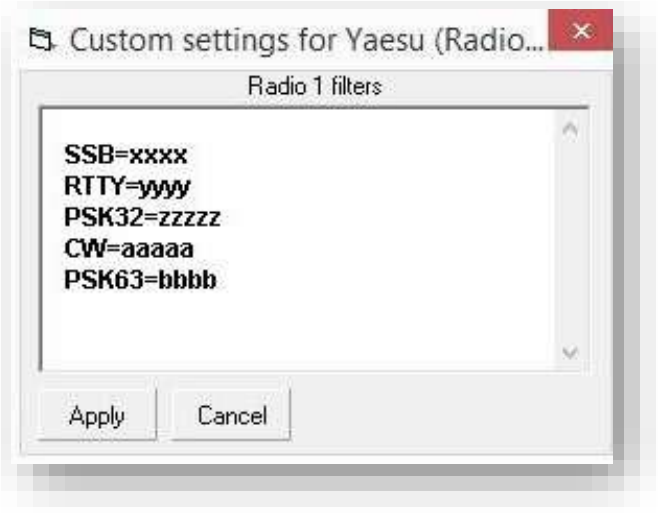
… combined becomes …

$**HexCommand 00 00 00 01 0C 00 00 00 00 8C**$

The FT-990 CAT function is hungry enough to accept one or two commands in one line. The FT-1000MP accepts up to three per line: its CAT must be *ravenous*.

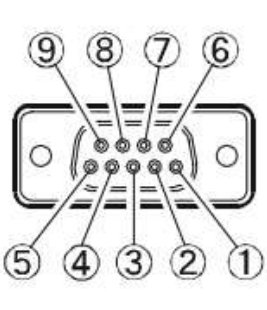
51.13.5 Custom settings by mode

Under **Setup ⇨ Radio ⇨ Radio 1|2 configuration**, click to select (tick) <**Use Narrow Filters for CW**> then click <**Custom settings**> ▶



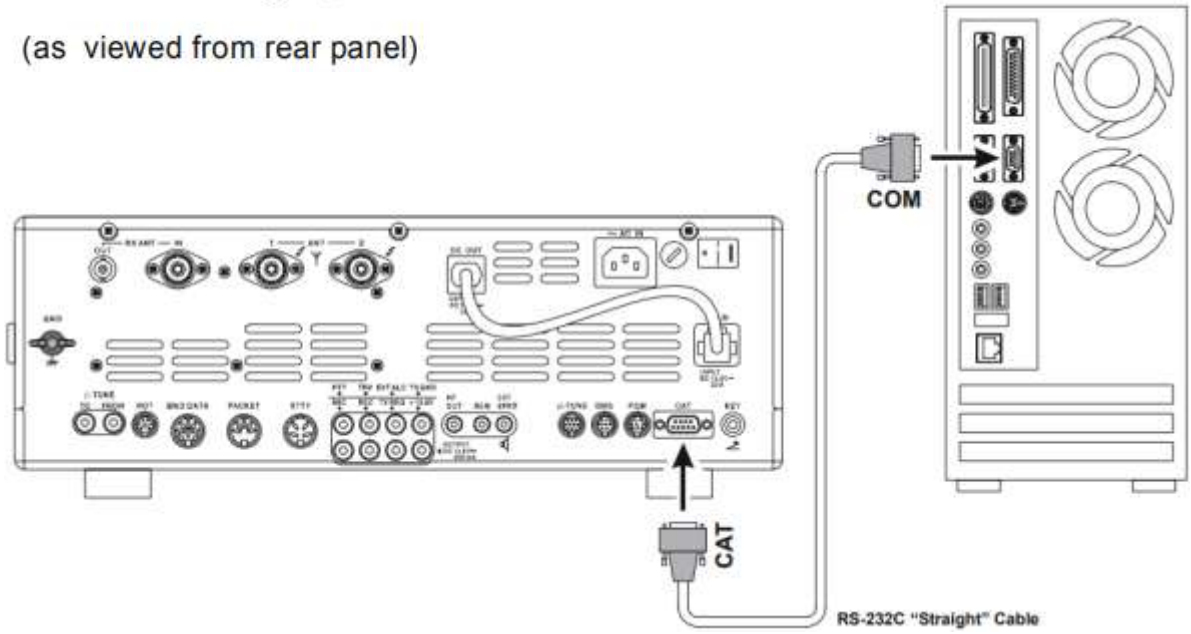
Copyright © 2026 Robert C. Furzer K4CY

51.14 Yaesu FT-2000 and FT-2000d



◀ The FT-2000 radios have a DB9 pin RS232 CAT port on the rear panel.

See the operating manuals for details of the CAT configuration menu settings (028 to 031), and the FT-2000 CAT Operation Reference Book for details of the CAT commands available. The radio's default CAT settings are 4800 baud with RTS enabled.

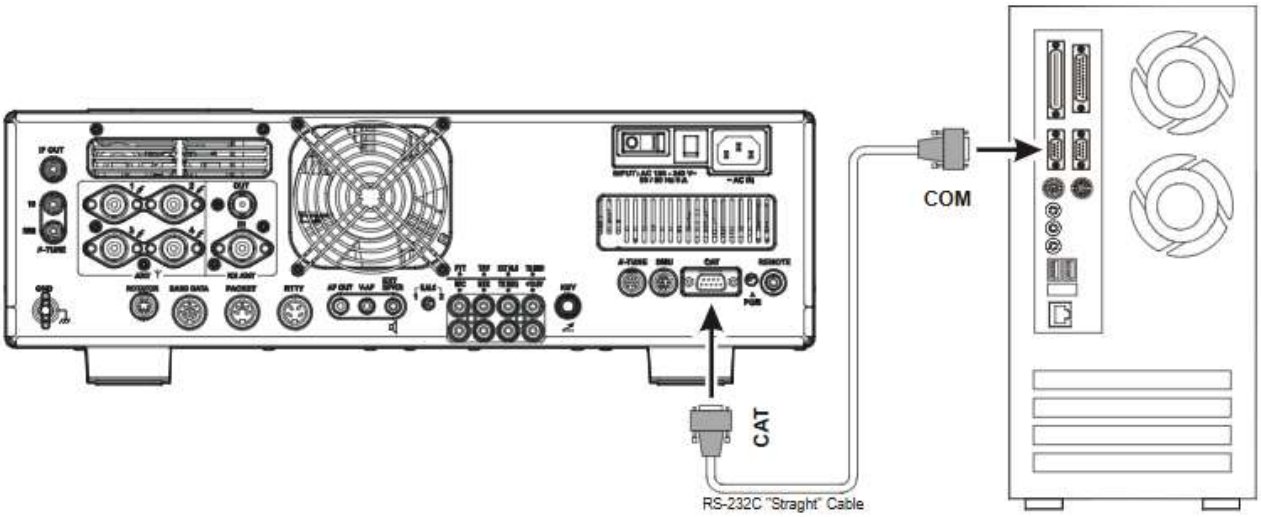




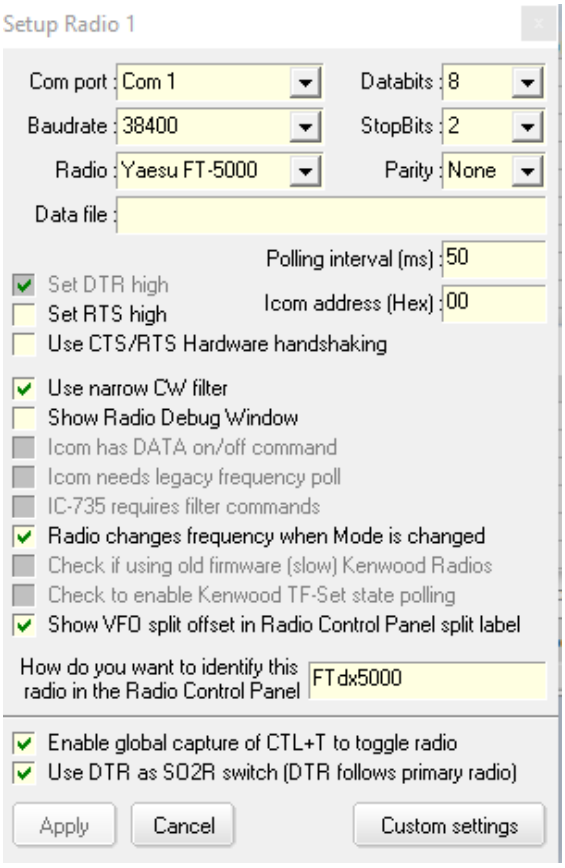
The *optional* Yaesu SCU-17 interface ▲ contains an RS232 to USB converter for CAT control of the radio, and a USB audio system/sound card for digimode operation, powered from the PC's USB port. It supports several Yaesu radios *e.g*. FT-450, FT-817, FT-950, FTDX-5000 and FT-2000. You'll need Yaesu's virtual COM port driver on your PC to use this unit: with a single USB cable linking the SCU-17 to your PC and the appropriate cables linking the SCU-17 to the radio, it provides *two* virtual COM ports on the PC, one for CAT and another for audio.

Copyright © 2026 Robert C. Furzer K4CY

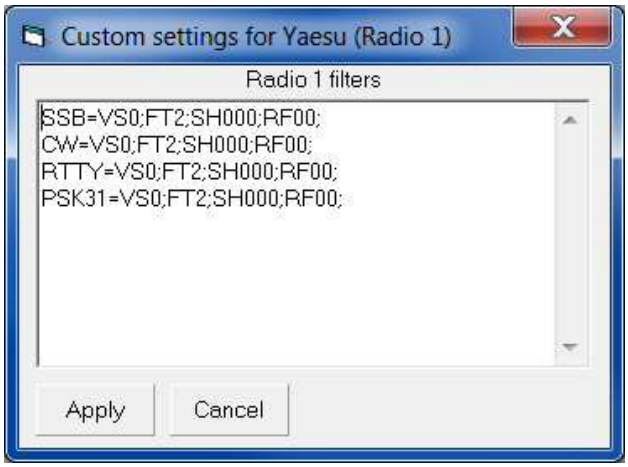
51.15 Yaesu FTdx-5000



Set up the CAT connection using
Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶



Unless the roofing filter is set to 'auto', the roofing filter will not change when a DX spot is clicked.



▲Those commands (defined through the <**Custom settings**> button) allow you to click a CW spot and set the roofing filter to 'auto'. VFO 'A' is selected and the final command moves the 'TX' back to VFO A.

Note: If the orange 'B' VFO is selected, this moves it back to the red VFO 'A'.

Darren G0TSM uses the following CAT commands strings to set up his FTdx5k for various modes:

- SSB=VS0;NA00;SH020;GT04;RF00;AB;MG087;PL081;FT3;PC080;FR0;ML0001;
- CW=VS0;SH006;GT04;RF00;AB;NA10;CO0001;CO0106;KP31;FT3;PC080;SH120;FR0;MG087;PL081;ML0001;
- RTTY=MD06;VS0;GT04;NA01;SH010;RF00;AB;NA10;FT3;PC056;SH116;FR0;MG087;PL081;ML0001;
- FT8=MD0C;VS0;GT02;NA00;SH016;RF00;AB;FT3;PC043;FR0;MG087;PL081;ML0000;

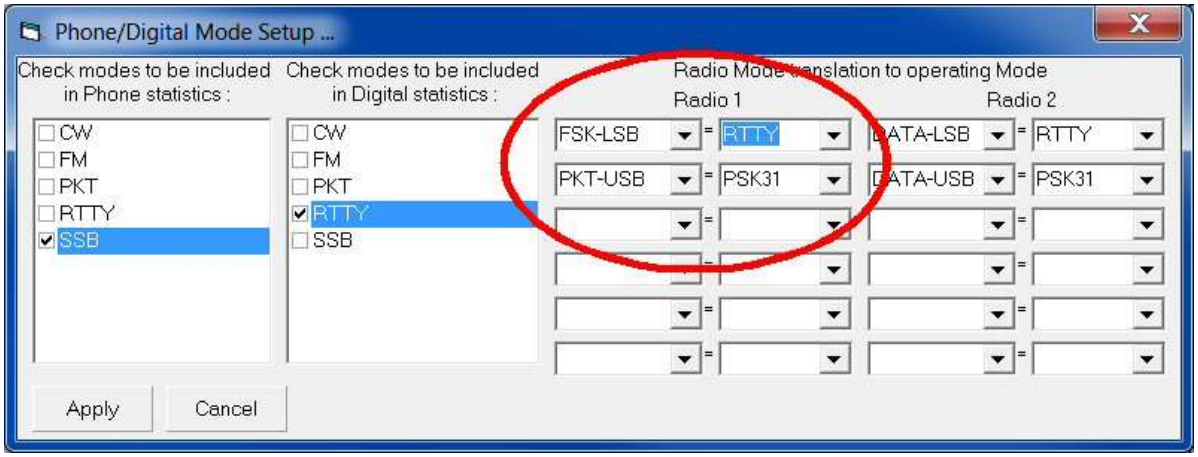
Copyright © 2026 Robert C. Furzer K4CY

- FT4=MD0C;VS0;GT02;NA00;SH016;RF00;AB;FT3;PC043;FR0;MG087;PL081;ML0000;

The strings set things such as split, power, audio and filters: check the FTdx-5000 CAT Manual for information about the CAT commands and parameter values.

51.15.1 FTdx-5000 data modes

For the correct data mode filters to be selected the operating modes must first be defined. In the example listed below operation on RTTY uses the mode FSK-LSB and for PSK31 PKT-USB is used.



Clicking a PSK31 spot changes the FTdx-5000 to PKT-USB and the RF00;VS0;FT2; CAT commands are sent to the radio.

51.15.2 FTdx-5000 CAT commands and scripts

You too can become a CAT programmer! The full list of CAT commands is given in the Yaesu FTdx-5000 series CAT Operation Reference Book. Here are the main ones to get you started – try 'em!

- $**Command FT1;**$ Split on
- $**Command FT0;**$ Split off
- $**Command VS0**;$ VFO A
- $**Command VS1;**$ VFO B
- $**Command AB;**$ A=B
- $**Command BA;**$ B=A
- $**Command MK1;**$ Mode Key USB
- $**Command MK0**;$ Mode Key LSB
- $**Command MK5;**$ Mode Key RTTY
- $**Command MK2**;$ Mode Key CW
- $**Command mk3;**$ Mode Key AM/FM
- $**Command MK6**;$ Mode Key PKT
- $**Command FT0;**$ Toggle TX A/B
- $**Command SV**;$ Swap VFOs A⇆B
- $**Command LK1;**$ Lock main VFO
- $**Command LK0;**$ Unlock main VFO

Copyright © 2026 Robert C. Furzer K4CY

- $**Command LK0;**$ Lock VFO B
- $**Command LK2;**$ Unlock VFO B
- $**Command NB01;**$ Noise blanker main on
- $**Command NB00;**$ Noise blanker main off
- $**Command BC01;**$ Auto notch main VFO on
- $**Command BC00;**$ Auto notch main VFO off
- $**Command NR01;**$ Noise reduction main on
- $**Command NR00;**$ Noise reduction main off
- $**Command PS1;**$ Power on
- $**Command PS0;**$ Power off
- $**Command AG0000;**$ VFO A volume 0 (mute)
- $**Command AG0040;**$ VFO A volume 40
- $**Command AG0055;**$ VFO A volume 55 … *etc*.
- $**Command AG1000;**$ VFO B volume 0 (mute)
- $**Command AG1075;**$ VFO B volume 75 … *etc*.
- $**Command pc059;**$ Power 20
- $**Command pc072;**$ Power 30
- $**Command pc085;**$ Power 40
- $**Command pc090;**$ Power 45
- $**Command pc095;**$ Power 50
- $**Command pc100;**$ Power 55
- $**Command pc110;**$ Power 65
- $**Command pc145;**$ Power 100
- $**Command pc160;**$ Power 125
- $**Command pc185;**$ Power 150
- $**Command pc205;**$ Power 175
- $**Command pc255;**$ Power 200

Simple example scripts:

- $**Command MK0;**$ Mode Key LSB
- $**Command FT2;**$ Split off
- $**Command FA03930000;**$ QSY to 3930 kHz

- $**Command MK0;**$ Mode Key LSB
- $**Command FT2;**$ Split off
- $**Command FA01855000;**$ QSY to 1855 kHz

Copyright © 2026 Robert C. Furzer K4CY

DX spot macros: right-click the RCP then click <**Setup**> …

Top window

- $**Command LK2;**$

Middle window

- $**Command AB;**$ Copy VFO A frequency and mode to VFO B.
- $**Command FB#DXspotSplit#;**$ Take VFO A freq, add the split offset, send to VFO B.
- $**Command FR2;**$ Turn on both main and sub-receivers
- $**Command FT3;**$ TX on VFO B

Bottom window

... nothing: leave it blank.

WSJT-X or JTDX Setup shortcuts window

If you are using the rear data port on the FTdx-5000, these commands will change to the rear port and select the wide filter when starting WSJT-X or JTDX, then change back to the front and narrow the filter after shutting it down …

Top window (before starting)

- $**Command EX1031;**$
- $**Command SH025;**$

Bottom window (after closing)

- $**Command EX1030;**$
- $**Command SH013;**$

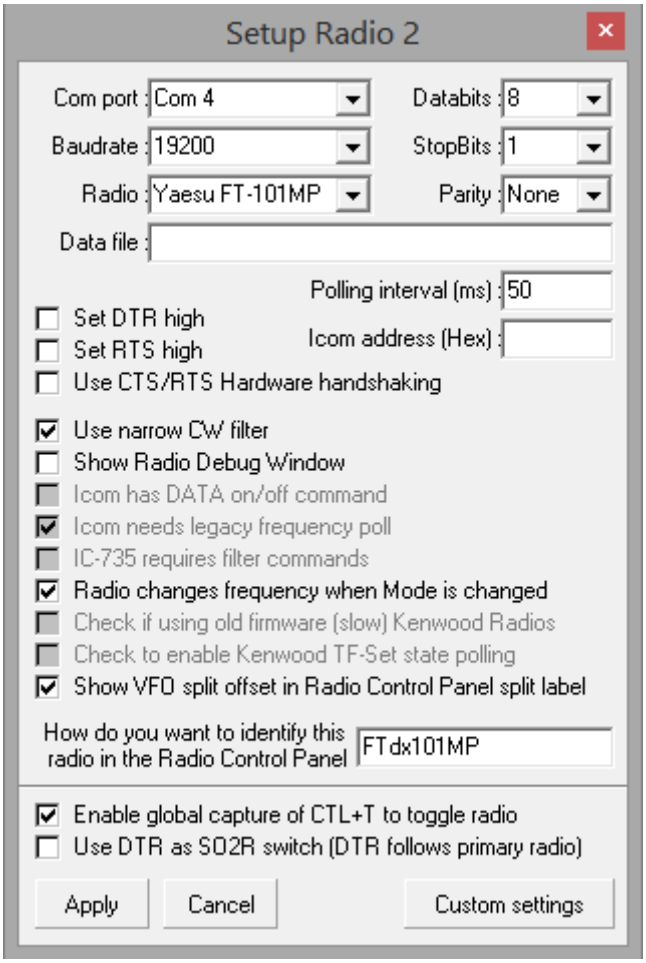
51.15.3 FTdx-5000 and the CW Machine with Winkeyer

Darren G0TSM says *"The FTDX5000 is a bag of poop in full breakin/QSK. At 24WPM you can't hear anything between the dits and dahs. It's also too clever, or dumb, or both: when I enable the WinKeyer PTT, it cancels the semi breakin delay in the radio, in effect putting itself into QSK.*

I don't like using semi breakin anyway as I need a long delay to reduce amp relay noise. Using QSK (or semi breakin with no CW delay) with Winkeyer PTT gets the rig back to receive quickly at the end of the macro, which is a bonus (love those echoes!). Setting the Winkeyer PTT tail delay to 50ms completely stops the amp relay flicking ***during*** *a message."*

Copyright © 2026 Robert C. Furzer K4CY

51.16 Yaesu FT-101D (FTdx101D) and FT-101MP (FTdx101MP)



An ordinary USB A-B cable is the simplest way to connect your PC's normal USB A socket to the FTdx101's rear panel USB B socket … but first you need to download Yaesu's virtual COM port driver and install it on the PC, creating a *pair* of virtual serial Com ports. See the FTdx101MP/FTdx101D Operation Manual and USB Driver Virtual COM Port Driver Installation Manual for details, and the FTdx10 section below for hints.

The FTdx101D and FTdx101MP have the same CAT port setup and commands.

Open the CAT configuration form using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

Select whichever Com port connects to *your* FTdx101 – maybe not Com 4 – and the port speed set on your radio – maybe not 19200.

Hinson tip: in the Windows device manager, one virtual Com port is "standard" (used for audio) and the other is "enhanced" (for CAT).

Use the other settings shown here.

The 50ms polling interval is faster than normal for quicker radio responses to changes initiated from Logger32, and quicker updates to the radio info in Logger32 when you adjust the radio directly (*e.g.* if you QSY the VFO or change band, mode or filters). If your PC or the radio can't keep up, increase the polling interval until it behaves itself reliably – try 100 ms, maybe more.

Under <**Custom settings**>, you can set up different filter widths *e.g.* to use the default bandwidths except on SSB:

- **SSB=VS0;FT2;SH0015;RF00;** Width 2.5
- **CW=VS0;FT2;SH0000;RF00;** Default
- **RTTY=VS0;FT2;SH0000;RF00;** Default
- **PSK31=VS0;FT2;SH000;RF00;** Default

For RTTY, define the radio mode **"RTTY-LSB"** in your Bands & Modes table, since plain "RTTY" defaults to the wrong sideband (apparently). Having setup for "RTTY-LSB", Logger32 sends the CAT command MD06; to the radio when you click a DX spot on one of the nominal RTTY frequencies. Likewise for data modes such as FT# and PSK, try commanding the radio to its DATA-U mode by defining **"PKT-USB"** as the radio mode in your Bands & Modes table.

Copyright © 2026 Robert C. Furzer K4CY

51.16.1 FTdx101 CW keying

The CW machine, working as a software keyer, can toggle either the DTR or CTS lines of a standard virtual com port to key the carrier. Choose DTR or CTS keying in the CW Machine, then configure the FTdx101 the same way using the radio's menu under MODE CW ⇨ PC KEYING. The PC and radio must both use the same DTS or CTS serial line for keying or it won't work.

51.16.2 FTdx101 CAT commands

Here are some common CAT commands for the FTdx101 radios and simple scripts that can be used in Logger32 macros. Consult the FTdx101MP/D CAT Operation Reference Manual for full details on these and other CAT commands available:

- $**Command MD0;**$ USB D
- $**Command MD01;**$ LSB
- $**Command MD02;**$ USB
- $**Command MD03;**$ CW
- $**Command MD04;**$ FM
- $**Command MD05;**$ AM
- $**Command MD06;**$ RTTY
- $**Command ST1;**$ Split on
- $**Command ST0;**$ Split off
- $**Command FT2;**$ Transmit on VFO A (main)
- $**Command FT3;**$ Transmit on VFO B (sub)
- $**Command VS0;**$ Select VFO A
- $**Command VS1**;$ Select VFO B
- $**Command AB;**$ A=B
- $**Command SV;**$ Swap VFOs A⇆B
- $**Command ST2;**$ Split +5 (each time you run this macro, VFO B increments +5 kHz).
- $**Command LK1;**$ Lock on
- $**Command LK0;**$ Lock off
- $**Command NB01;**$ **N**oise **B**lanker on
- $**Command NB00;**$ **N**oise **B**lanker off
- $**Command NR01;**$ **N**oise **R**eduction on
- $**Command NR00;**$ **N**oise **R**eduction off
- $**Command ML0001**;$ Monitor on
- $**Command ML0000**;$ Monitor off
- $**Command AG0000;**$ Volume 0
- $**Command AG0080;**$ Volume 80
- $**Command AG0100;**$ Volume 100
- $**Command PA00;**$ Set VFO A (main) IPO (pre-amp) to 0
- $**Command PA01;**$ Set VFO A (main) IPO (pre-amp) to amp 1
- $**Command PA02;**$ Set VFO A (main) IPO (pre-amp) to amp 2

Copyright © 2026 Robert C. Furzer K4CY

- $**Command VT01-0;**$ Set main VC Tune
- $**Command SF02;**$ Set sub dial MPVD to VC Tune
- $**Command PC005;**$ Power 5 watts
  $**Command PC010;**$ Power 10 watts … and so on … up to …
- $**Command PC200;**$ Power 200 watts (for the FTdx101MP)
- $**Command BC01;**$ Auto-notch on
- $**Command BC00;**$ Auto-notch off
- $**Command BI1;**$ Enable break-in
- $**Command BI0;**$ Disable break-in
- $**Command PS1;**$ Power on (NB: this command *only* works through the radio's built-in USB port, *not* using the serial com port)
- $**Command PS0;**$ Power off

Example FTdx101 macros

- $**Command FT2;**$ Transmit on VFO A
- $**Command MD02;**$ USB
- $**Command FA014195000;**$ Set VFO A to 14195 kHz

- $**Command FT2;**$ Transmit on VFO A
- $**Command MD01;**$ LSB
- $**Command FA003730000;**$ Set VFO A to 3730 kHz

- $**Command MD03;**$ CW
- $**Command BI1;**$ Enable break-in
- $**Command KY1;**$ Send keyer memory #1
- $**Wait5**$ Pause a while
- $**Command MD01;**$ LSB
- $**Command BI0;**$ Disable break-in

  Sends my callsign stored in CW from memory #1, waits briefly for the radio to send the message, then switches back to LSB. This technique can be used for any of the memories but the wait time is unreliable so it *may* not switch back to LSB. You may need to add a second $**Wait5**$ after the first. If only Logger 32 had more time options!

Copyright © 2026 Robert C. Furzer K4CY

DX spot macros: right-click the RCP. Click <**Setup dx spot macros**> then enter these commands:

Window 1 (clear radio split)

- $**Command AC000;**$ Turn off antenna tuner[477]
- $**Command PA00;**$ Enable pre-amp 1 on main receiver
- $**Command VS0;**$ The big knob now controls VFO A as usual
- $**Command FT2;**$ Transmit on VFO A
- $**command FR01;**$ Receive on VFO A only, muting the sub-receiver

Window 2

- $**Command AB;**$ Copy A to B
- $**Command FB#DXspotSplit#;**$ Take the VFO A frequency, add the split offset indicated in a DX spot comment, then send VFO B to the resulting frequency.
- $**Command ST1;**$ Turn on split
- $**Command AC000;**$ Turn off antenna tuner
- $**Command PA00;**$ Enable pre-amp 1 on main receiver
- $**Command FT3;**$ Transmit on VFO B
- $**Command FR00;**$ Enable *both* the main *and* sub-receivers to hear the DX *and* pileup
- $**Command VS1;**$ Use the big knob to tune your transmit frequency (VFO B) in the pile

Window 3 (commands after clicking a simplex DX spot)

... nothing: leave it blank.

Here are some alternative DX spot macros to try out on the FTdx101D, courtesy of Pete N5KD ▶

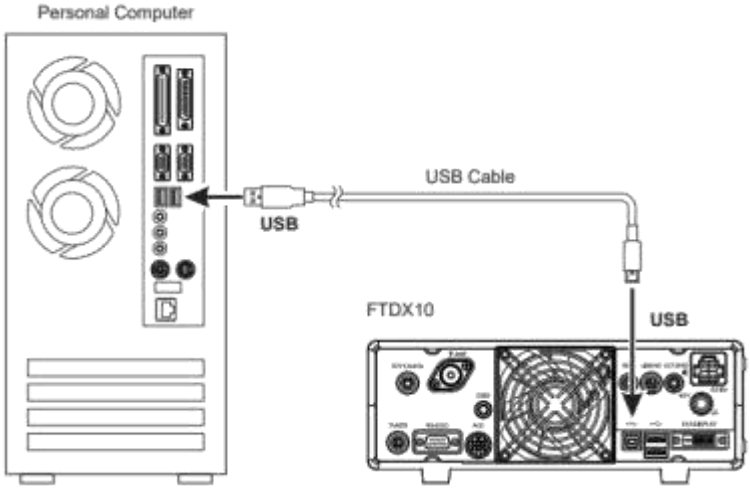


[477] This is just an example. You may not want to disable your antenna tuner! Feel free to experiment with the CAT commands and their sequence. To save typing and speed-up execution, the commands plus their closing semicolons can be conCATenated without the comments in the form of $**Command AB;FB#DXspotSplit#;ST1;AC000;PA00;FT3;FR00;VS1;**$ … but you may regret omitting the explanatory comments the next time you decide to check/update/explain your macro. So keep notes in your shack notebook?

Copyright © 2026 Robert C. Furzer K4CY

51.17 Yaesu FTdx10

The FTdx10 has a built-in Silicon Labs dual CP210x USB to UART bridge chip feeding a USB-B CAT connector on the rear of the transceiver ▼



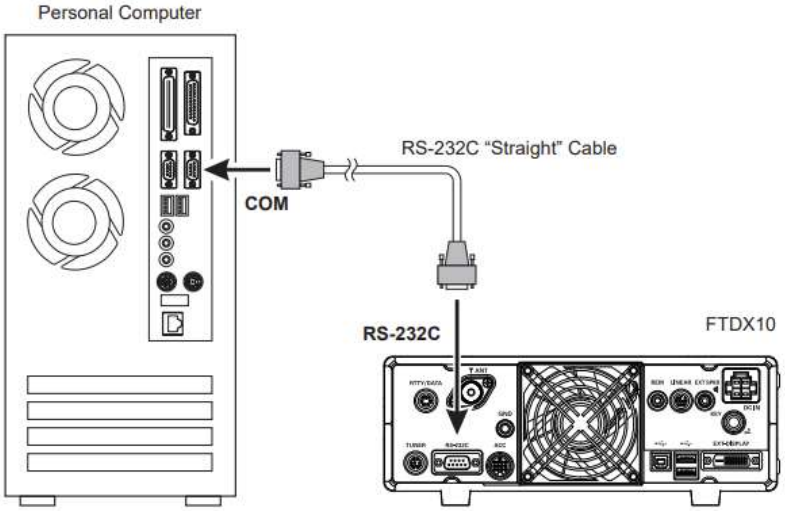
Before physically connecting your new FTdx10 to your Logger32 PC with a standard USB A-B cable, carefully follow the instructions in the Virtual COM Port Driver Installation Manual.

If it doesn't go to plan, check the FTdx10 reflectors at Groups.IO, or videos on YouTube, or contact Yaesu for advice.

Hinson tip: if you eagerly connected up the USB prematurely and the port isn't working, you *may* need to *un*install the default plug-n-play port driver installed automatically by Windows, then install the correct driver. Do this through **Windows Control Panel** ⇨ **Device Manager** ⇨ **Ports (COM & LPT)**.

Alternatively, a straight-through 9-pin RC-232 cable will work ▶ *if* your PC has a conventional RS232 connector, and follows the RS232 standard.

Use one or the other – either USB or RS232, not both!



Once your radio is connected and communicating with the PC, refer to the FTdx101 section above to configure it in Logger32. Pete N5KD reports that the "**YAESU FT-101MP**" radio setting in Logger32 works with the FTdx10, using 38400 baud, 8 bits, 2 stop bits, no parity bits with CTS/RTS hardware handshaking enabled.

Hinson tip: by all means explore the FTdx10 macros with the FTdx10 CAT Operation Reference Manual and try them out through Logger32's **R**adio **C**ontrol **P**anel. If you figure out anything really useful or interesting, why not share it through the Logger32 reflector – or email me to include it here in the manual for the benefit of other FTdx10 DXers.

Copyright © 2026 Robert C. Furzer K4CY

52 Software Defined Radios (SDRs)

> "My wireless transmitter does not use Hertzian waves, which are a grievous myth, but sound waves in the aether"
>
> *Nikola Tesla*

The following SDRs are supported by Logger32:

- **Pegasus**: using either TEN-TEC or N4PY control software.
- **Orion, Argonaut V, Elecraft K2, RX320, RX331, RX340, RX350 and Jupiter**: using N4PY control software.
- **SmartSDR:** supports the FLEX-6300, FLEX-6400, FLEX-6400M, FLEX-6500, FLEX-6600, FLEX-6600M and FLEX-6700 radios.
- **Other SDRs**: still using the deprecated PowerSDR software.

52.1 Pegasus

Logger32 reads a file created by the control software called *Pegasus.out*

Logger32 writes radio commands to a file called *Pegasus.in*

When setting up the Pegasus radio, specify the path to the folder: just the path, not the filenames. It is not necessary to configure the COM port in the Radio Control window since serial port communication is not used.

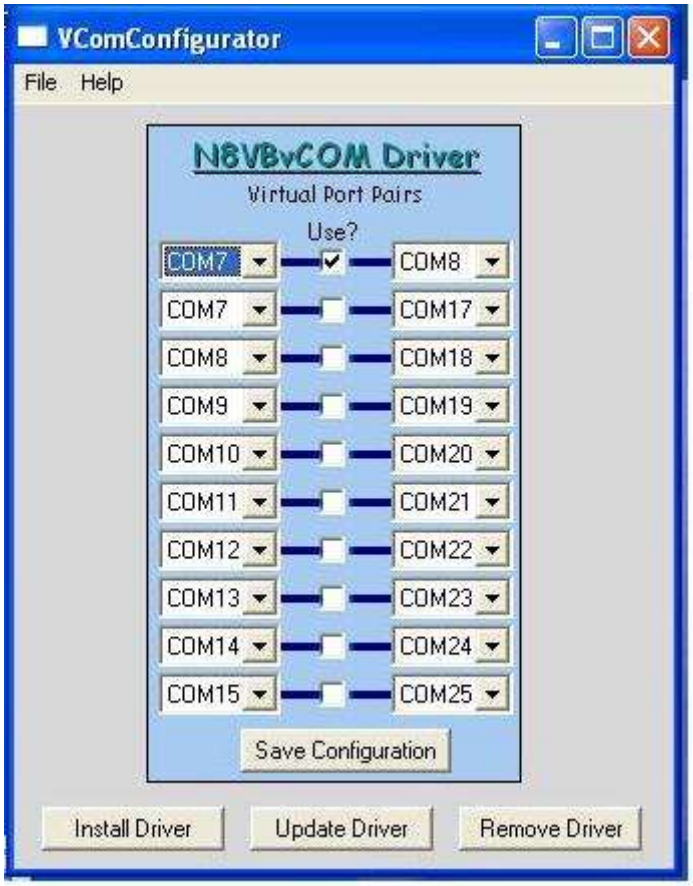
The $Command$ and $HexCommand$ macros do not work with PC-controlled radios that read and write directly to disk files for data transfer.

Copyright © 2026 Robert C. Furzer K4CY

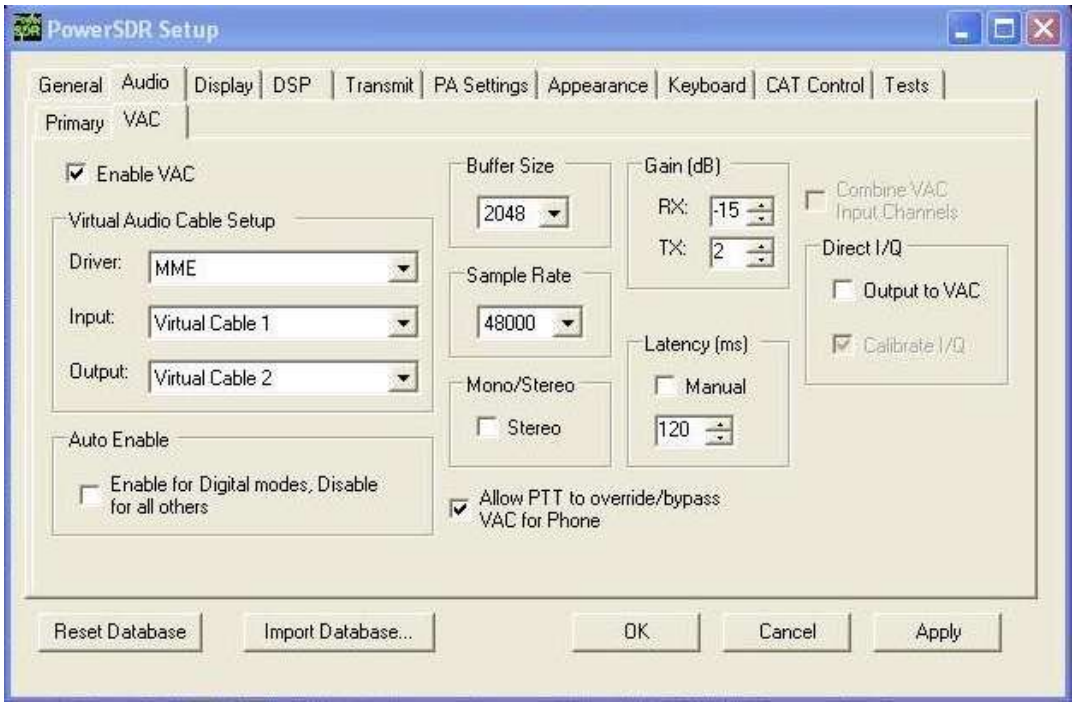
52.2 Power SDR

- Open the N8VB VComConfigurator window and select the proper Virtual Port Pair ▶

  The example shows COM7 and 8.



- Open PowerSDR Setup, <**VAC**> tab ▼

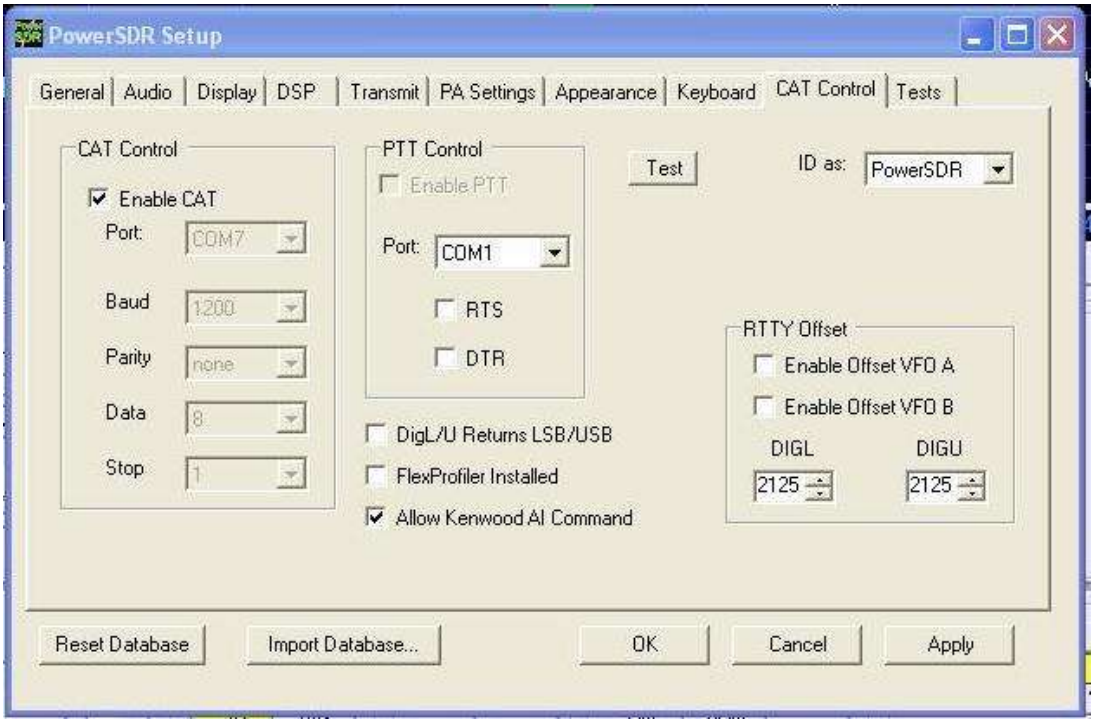


Set the following:

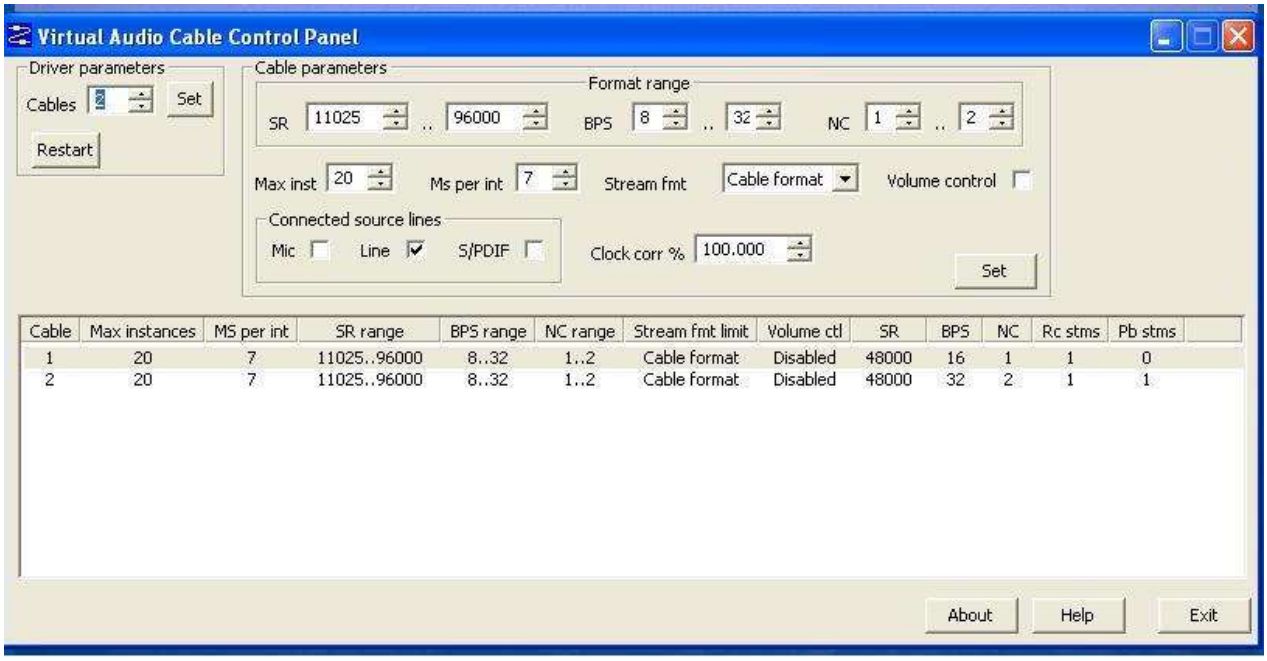
- Tick <**Enable VAC**>
- **Input** = Virtual Cable 1
- **Output** = Virtual Cable 2
- Buffer Size = 2048

Copyright © 2026 Robert C. Furzer K4CY

- Open PowerSDR Setup, <**CAT Control**> tab ▼

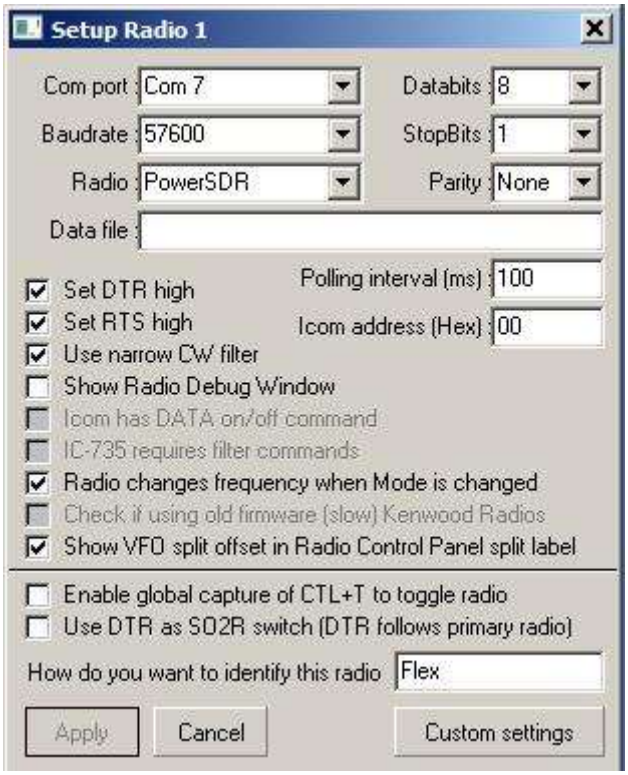


- Set it to match the VCom settings.
- Tick <**Enable CAT**>
- Select the PTT port
- Open the Virtual Audio Cable Control Panel and apply these settings ▼



Copyright © 2026 Robert C. Furzer K4CY

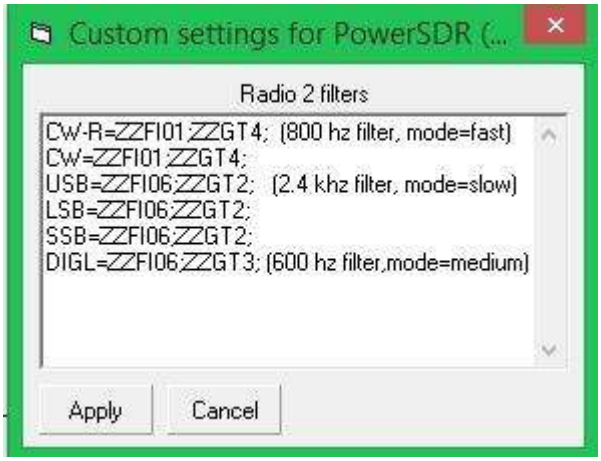
52.2.1 Logger32 CAT setup for Power SDR



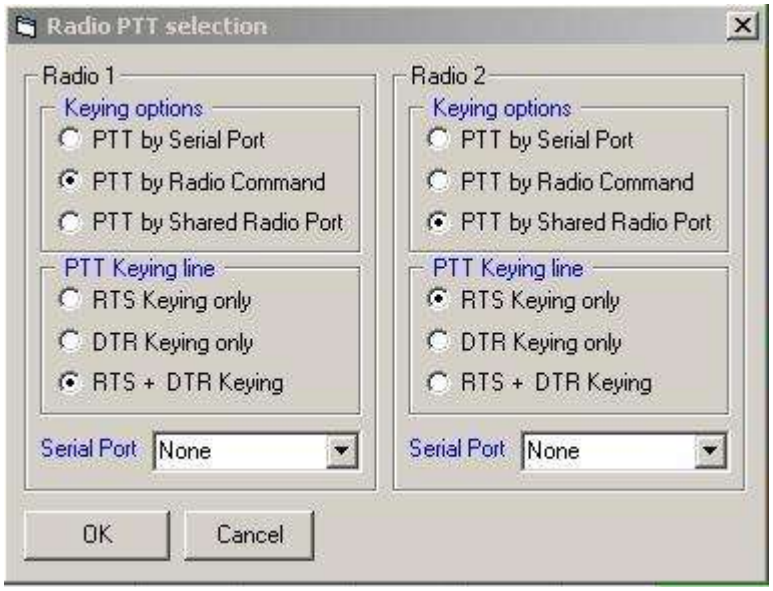
Open **Setup ⇨ Radio ⇨ Radio 1|2 configuration**. Set the COM port to match the VCom settings ▶

If you are using the SmartSDR software, select <**PowerSDR**> for the radio type.

Click <**Custom settings**> to configure the filtering for various modes. Here are some examples ▼



- Open MMVARI through the Sound card data window. Open **Settings ⇨ Radio PTT options** to set your PTT keying and COM port. These settings must match the PowerSDR software configuration ▼

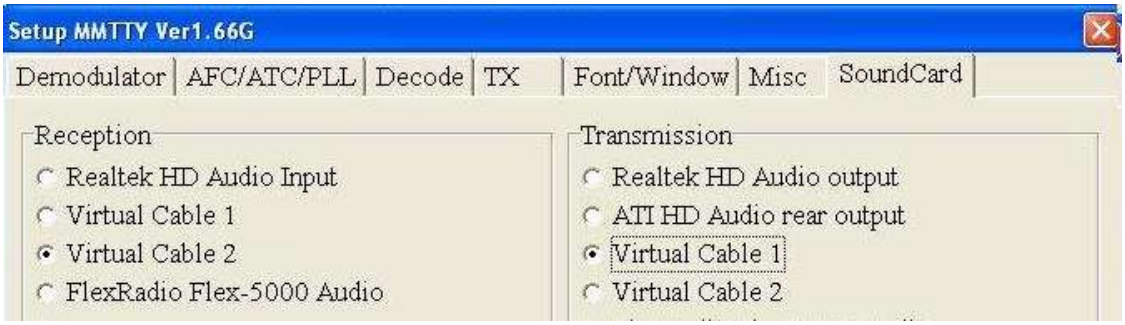


Copyright © 2026 Robert C. Furzer K4CY

- Open the Sound card data window with MMVARI ▶
  - **Input** = Virtual Cable 2
  - **Output** = Virtual Cable 1

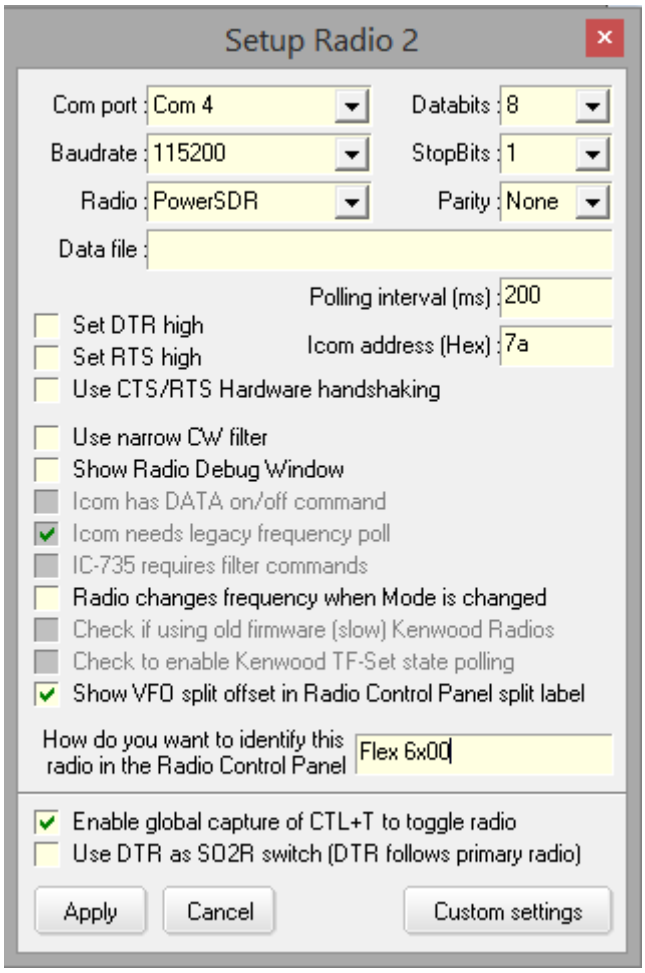


- Open the Sound card data window with MMTTY.
- Select the MMTTY setup and click the <**Sound Card**> tab.
- Make the following selections for sound card reception and transmission ▼



Copyright © 2026 Robert C. Furzer K4CY

52.3 Flex 6000 series



Configure CAT for the Flex 6300, 6400, 6600, 6700, 6800 … "Signature series" using **Setup ⇨ Radio ⇨ Radio 1|2 configuration ▶**

Try:

- Radio type SmartSDR or PowerSDR
- 38400 or 115200 baud (it autoadapts)
- 8 Databits
- 1 StopBit
- Parity None
- COM port connected to Slice A in SmartSDR CAT
- Polling at 200 ms (with a modern PC, you should be able to poll faster for a more responsive system: reduce the interval step by step, checking that it remains reliably connected at each stage- if not, revert).

In your Bands & Modes table, use your choice of CWL (lower sideband) or CWU (upper sideband) rather than plain CW as the Radio Mode, and DIGU for digital modes.

While you are working on the SmartSDR file, you might like to create a WinKeyer port as well, configuring the same WinKeyer port in the CW Machine (unless you prefer the CWX feature in SmartSDR). Study the Flex-6000 Signature Series SmartSDR CAT User Guide for clues.

52.3.1 Flex CAT commands

Flex radios support a subset of the Kenwood TS-2000 CAT command set, plus a bunch of FlexRadio commands starting with ZZ, plus some OTRSP commands for SO2R control and a select few WinKeyer 2 commands for the built-in CW keyer.

Flex uses the Kenwood-style CAT commands *i.e.*:

command [space] optional parameters separated by spaces [semicolon]

e.g. ZZDE 1; enables diversity mode on a Flex6700

See the SmartSDR CAT User Guide for more information on the commands, parameters *etc.*

AI **A**uto **I**nformation on/off (reports VFO A/B frequency changes automatically)

FA Read/set VFO **A** frequency (Hz, 8/11 digits with leading zeroes)

FB Read/set VFO **B** frequency (Hz)

FR Read/set active **R**eceive VFO (0=A, 1=B)

FT Read/set **T**ransmit VFO (0=A, 1=B)

GT Read/set VFO A AGC mode (0=Off, 2=Slow, 3=Medium, 4=Fast)

ID Report ID (904=Flex-6700, 905=Flex-6500, 906=Flex-6700R, 907=Flex-6300)

Copyright © 2026 Robert C. Furzer K4CY

IF Reads transceiver status (15 parameters)

KS Read/set CW **K**eyer **S**peed (5-100 WPM)

KY Convert a string of up to 256 ASCII characters to Morse code and send CW

MD Read/set VFO A **D**SP mode

NB Read/set VFO A wide **N**oise **B**lanker state (on/off)

PC Read/set RF power level (0-100)

PT Read/set CW **P**itch/**T**one frequency (Hz, 3 digits)

RC **C**lear **R**IT

RD **D**ecrement (reduce) **R**IT frequency

RT Read/set VFO A RIT state (on/off)

RU Increment VFO A RIT frequency (**R**IT **U**p)

RX Go to receive

SH Read/set VFO A filter **H**igh-cut frequency index

SL Read/set VFO A filter **L**ow-cut frequency index

SM Read the **S**-**M**eter

TX Go to transmit

XT Read/set VFO A **X**IT state (on/off)

ZZAG Read/set VFO A **A**udio **G**ain (0-100)

ZZAI **A**uto **I**nformation state (on/off)

ZZAR Read/set VFO A **A**GC threshold (0-100)

ZZAS Read/set VFO B **A**GC threshold (0-100)

ZZBI Read/set **BI**naural RX state (on/off)

ZZDE Read/set VFO A **D**iversity (DIV) state (on/off) [FLEX-6700 only]

ZZFA Read/set VFO **A** **F**requency (11 digit Hz)

ZZFB Read/set VFO **B** **F**requency (11 digit Hz)

ZZFI Read/set VFO A DSP **F**ilter **I**ndex

ZZFJ Read/set VFO B DSP **F**ilter index

ZZFR Toggle VFO A/B active

ZZFT Toggle VFO A/B transmit

ZZGT Read/set VFO A AGC mode

ZZIF Read transceiver status (15 parameters)

ZZLB Read/set VFO A audio pan (0-100, left-to-right)

ZZLE Read/set VFO B audio gain (0-100)

ZZLF Read/set VFO B audio pan (0-100, left-to-right)

ZZMA Read/set VFO A mute (on/off)

ZZMD Read/set VFO A DSP mode

ZZME Read/set VFO B DSP mode

ZZMG Read/set transmitter **M**icrophone **G**ain (0-100)

ZZNL Read/set VFO A wide **N**oise **B**lanker threshold (0-100)

ZZNR Read/set VFO A **N**oise **R**eduction state (on/off)

ZZPA Read/set **P**an**A**dapter data IP port address

ZZPC Read/set RF output drive Level (0-100)

ZZPE **E**nable/disable **P**anadapter

ZZRC Clear RIT

ZZRD Decrement RIT frequency (**R**IT **D**own)

ZZRG Read/set VFO A RIT frequency (+/- 5 digit Hz)

ZZRT Read/set VFO A RIT state (on/off)

ZZRU Increment RIT frequency (**R**IT **U**p)

ZZRW Read/set VFO B RIT frequency (+/- 5 digit Hz)

ZZRX Read receive state (on/off) [inverse of MOX]

ZZRY Read/set VFO B RIT frequency (+/- 5 digit Hz)

ZZSM Read S-meter

ZZSW Set transmit VFO (0=A, 1=B)

ZZTX Set MOX state (on/off)

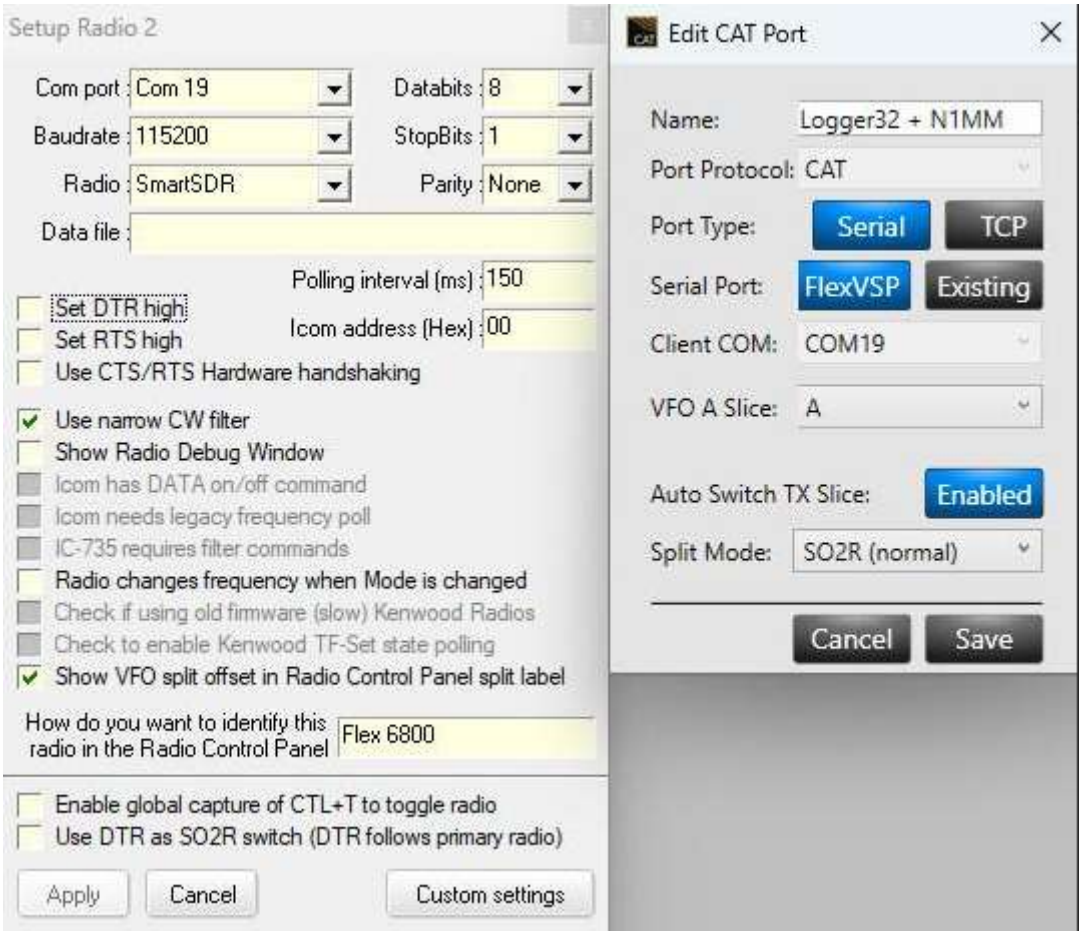
ZZXC Clear XIT frequency

ZZXG Read/set VFO A XIT frequency (+/- 5 digit Hz)

ZZXS Read/set XIT state (on/off)

Copyright © 2026 Robert C. Furzer K4CY

"These are my settings for the Flex 8600:



Always start Smartsdr and Smartcat (this also applies to Thetis and before all that PowerSDR) ***before*** *starting Logger32 or any other logging software for that matter (N1MM, DXlog etc) and let the logging software find Smartsdr.*

Polling rate matters, I have found that I can use a polling rate as low as 50ms ***provided*** *that after launching Logger32 I wait a few minutes for everything settle down and confirm that Logger32 is tracking Smartsdr by scrolling the waterfall. The delay seems to be the key."*

Frank GI4NKB

Copyright © 2026 Robert C. Furzer K4CY

Just got a new radio a Flex 8600m. I am learning to configure it and Logger32 but came across an interesting anomaly. Using the Kenwood TS2000 setting for the radio configuration it works nicely. Using SmartSDR (the ZZ prefixed codes) there is a problem with Logger32. Clicking on a CW spot in the bandmap or typing in a CW frequency the mode will change to USB on the higher HF bands.

The problem stems comes from the Bands & Modes table in that my table has CW in it while SmartSDR wants that to be CWU or CWL.

Being pragmatic and not fancying editing the whole of the Bands & Modes table, I decided to stay with the Kenwood TS2000 setting …

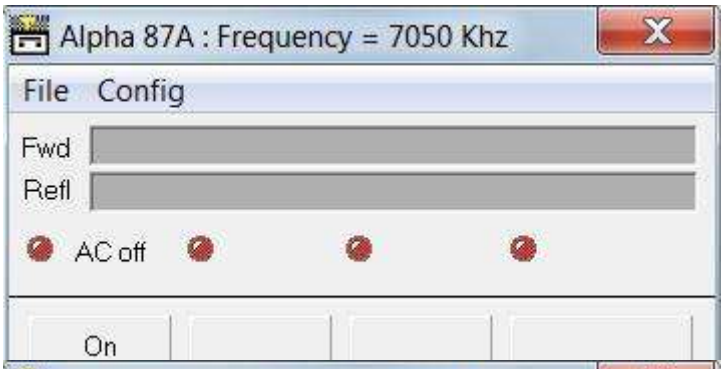
David G3YYD

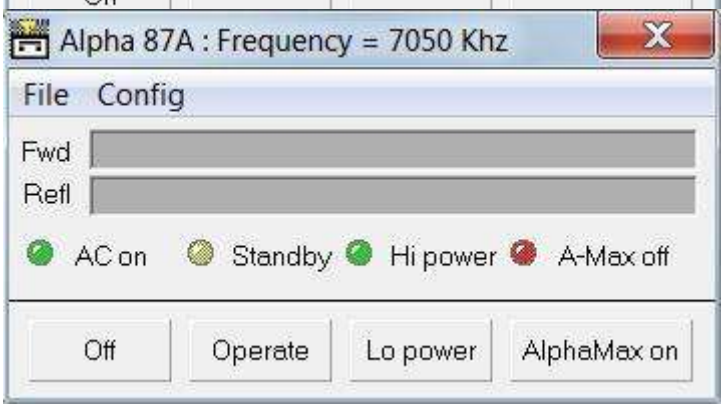
Copyright © 2026 Robert C. Furzer K4CY

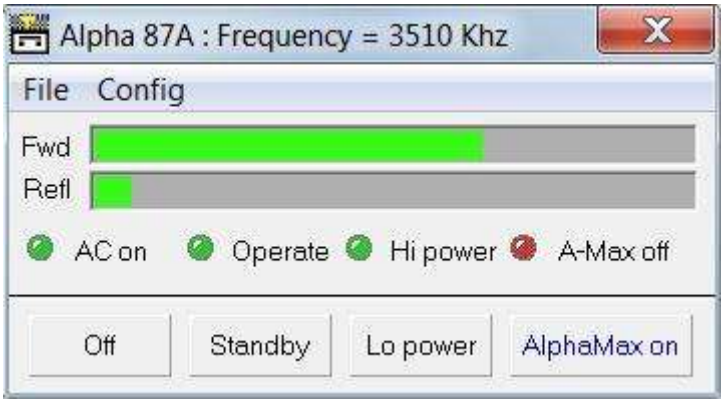
53 Alpha 87A linear amplifier

The venerable Alpha 87A autotune full-legal-limit HF amplifier includes provisions for external monitoring and control of some of its functions via a built-in RS232 serial port. In addition to monitoring Forward & Reflected output power from the 87A, Logger32 provides remote control and monitoring of the following amplifier conditions:

> "If you can't be good, be loud"
>
> *Rich Mullins*







- Power On/Off
- Standby/Operate
- High/Low Power
- AlphaMax[478] On/Off

Open the Alpha 87A window via icon #26 on the toolbar ▼



"Auto track frequency" causes Logger32 to send the active radio's VFO frequency to the amplifier, which uses that information to change bands and autotune for the new frequency.

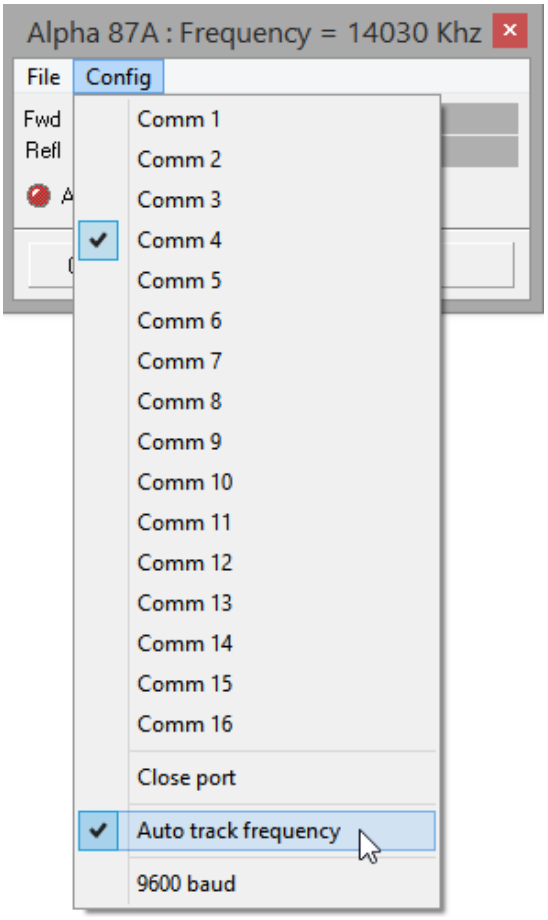
Although the 87A will change bands automatically by sensing the frequency of RF drive power from the transceiver, quicker band- and band-segment-changes that put less stress on the internal PIN-diode QSK system of the 87A and eliminate partial loss of the first dash or dots on CW (while the amplifier retunes in response to sensing a new operating frequency from the input RF) are made possible by having Logger32 supply the transceiver's current frequency through the serial interface.

[478] AlphaMax firmware in late model or updated 87A amplifiers dynamically fine-tunes the settings of the TUNE and LOAD capacitors during key down periods.

Copyright © 2026 Robert C. Furzer K4CY

Read the Communication Link section in the Alpha 87A user manual. Alpha uses some of the data pins to set the baud rate by pulling the pin(s) to zero volts. Make sure that any cabling between the computer and the amplifier includes the signaling wires TXD (2), RXD (3), CTS (5), RTS (4) and Gnd (7) and that the computer does not change the state of pins 9, 12, 13, 14 or 22 on this connector.

Configuring the Alpha is simple: select the serial port, its speed and/or whether you want the Alpha 87A to auto track and tune for the radio's current frequency (as opposed to RF sensing).



◀ Click the Alpha icon #26, then click <**Config**> and select the port number required. Click <**Config**> again and open the selected port.

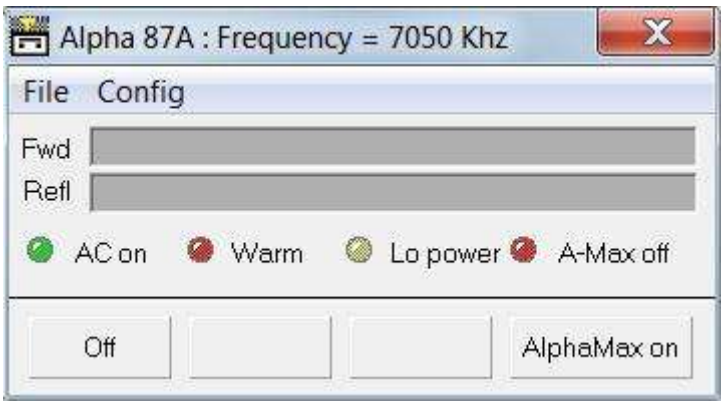
Logger32 supports the default 4800 baud rate for the Alpha 87A. If your Alpha 87A is set to 9600 baud, select (tick) 9600 baud at the bottom of the menu.

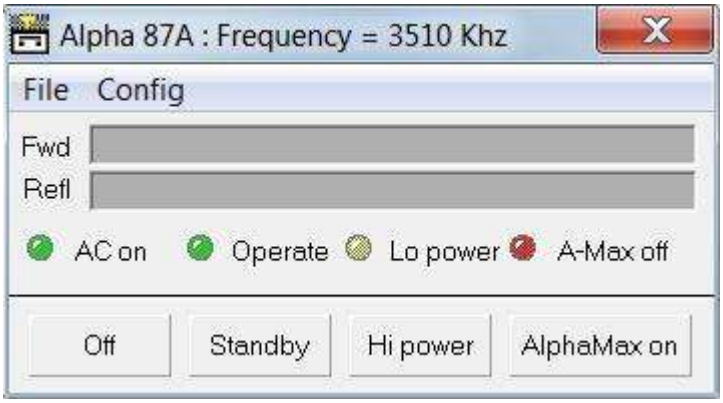
◀ To enable autotracking, click to select (tick) <**Auto track frequency**>[479].

[479] Logger32 normally supplies the frequency of the *active* radio. If the active radio is not the one feeding RF to the 87A, "Auto track frequency" may possibly put the 87A on the wrong band. However, if the serial port from the active radio is no longer open, frequency information previously provided by Logger32 to the amplifier may not be current and "Auto track frequency" will not be active, even if checked in the Alpha 87A Config menu. A quick check to be sure everything is functioning as you expected is to change bands on the radio that is supplying RF to the 87A and make sure the amplifier follows the radio to the proper bands.

Copyright © 2026 Robert C. Furzer K4CY

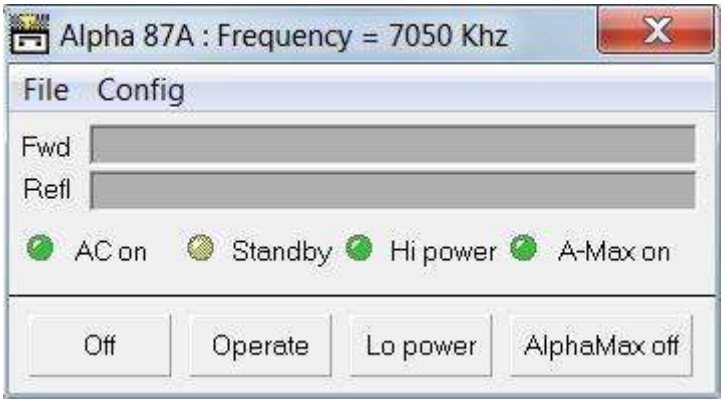
The window caption shows the frequency to which the amplifier is currently tuned and the "LEDs" tell you about its status *e.g.* AC on, warming up ▶

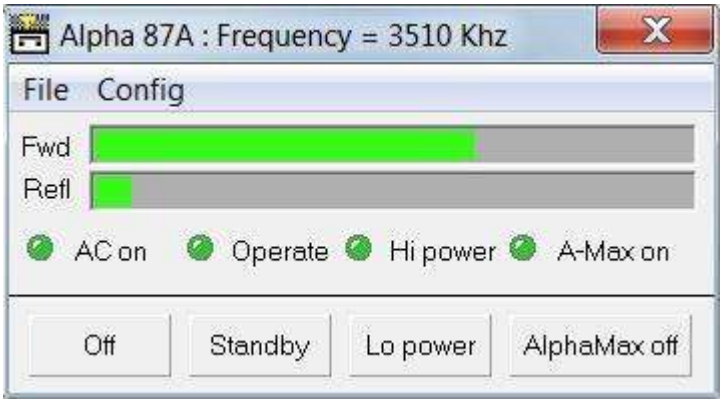




◀ Warmed up and ready to go on 80m, on low power with Alpha-Max disabled.

After clicking the <**Hi power**> and <**AlphaMax on**> buttons, and having retuned the radio to 40m ▶





◀ Transmitting on 80m.

◀ The bar graphs indicate forward and reflected power.

◀ Click <**Off**> to turn the amp off.

Provided the power cord is connected and live, the switched-off Alpha's CPU and serial port remain active, waiting for the signal to power on through the front panel ON switch or the <**On**> button in Logger32's Alpha 87A control function … hence the amp can be controlled remotely.

Copyright © 2026 Robert C. Furzer K4CY

54 *micro*HAM MK2R+

> "Big things come in small packages, and the world of technology is no exception"
>
> *Jeffrey Gitomer*

Amongst other things Logger32 can automatically command the *micro*HAM Radio 1 port (the 4 wire BCD coded TTL level) in the MK2R+ for antenna switching[480]. Using the $**uHam xx**$ macro, the Radio 2 port can also be controlled.

While Logger32 meets the requirement for port control, the *micro*HAM output is at TTL levels: a level-converting interface is likely to be required between the MK2R+ and any antenna relays *etc*.

54.1 Set up Logger32 for the *micro*HAM MK2R+



Passing control information to the MK2R+ requires a serial port number to be set using the Radio Control Panel menu in Logger32.

With the RCP showing, right-click the side bar to open the menu ▶



▼ Click <**microHAM Setup**>, select the port and click <**Open uHam port**> to connect.



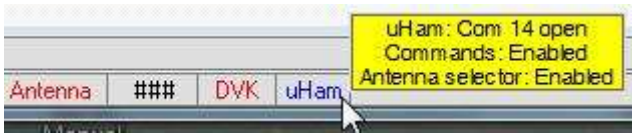
Tick <**Use microHam for control commands**> to hold the port open to send commands to the *micro*HAM device for other functions.

[480] If you are looking for info on the *micro*HAM ARCO rotator controller, it's over here.

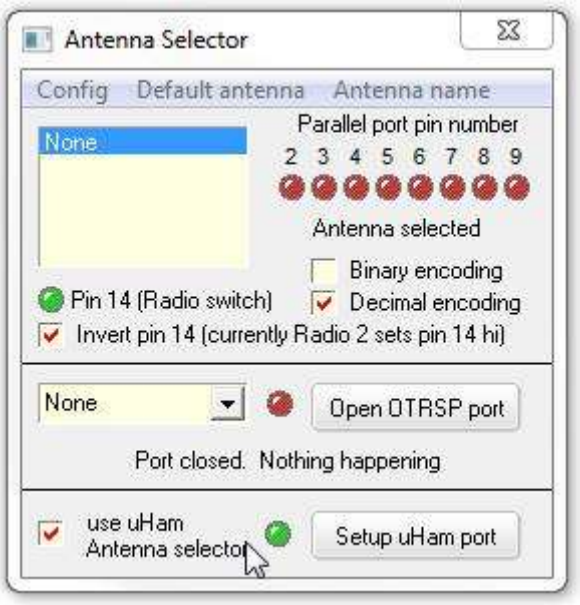
Copyright © 2026 Robert C. Furzer K4CY

There is a *micro*HAM panel on the status bar at the bottom of Logger32's main screen.

Mouseover it for a tooltip showing the status ▶



To use the *micro*HAM 4-bit port in preference to that provided in Logger32 while still maintaining the antenna selection function within Logger32, set the option to do so as shown here ▶



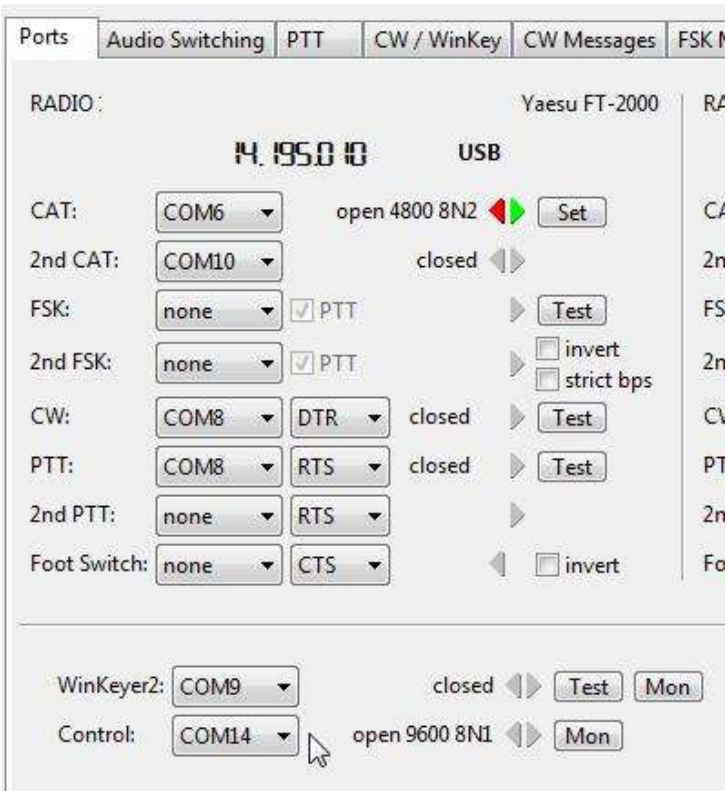
54.2 Set up the MK2R+

54.2.1 Virtual port

Need to set up a virtual port in the *micro*HAM router software? See the MK2R+ manual!

54.2.2 Set *micro*HAM control port

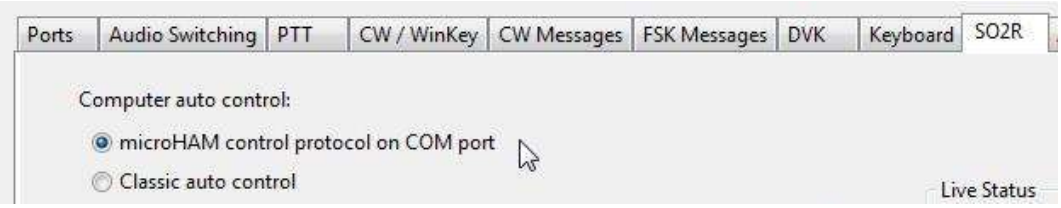
On the *micro*HAM router <**Ports**> tab, set the CAT control port ▶



Copyright © 2026 Robert C. Furzer K4CY

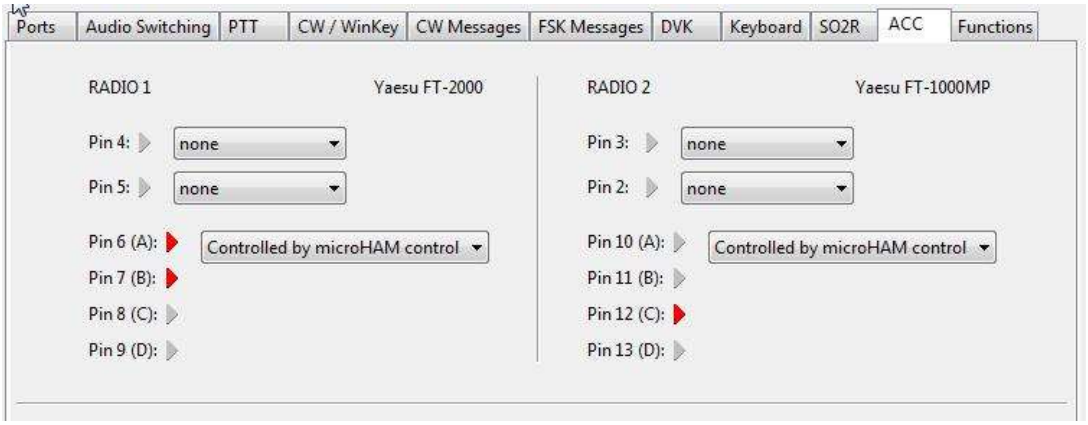
54.2.3 SO2R with the *micro*HAM

On the *micro*HAM SO2R tab set the computer auto control ▼



54.2.4 *micro*HAM ACC tab

On the <**ACC**> tab it is possible to observe the status of the 4 bit ports (Pins 6,7,8 and 9) and (pins 10,11,12 and 13) ▶



Configure them as “Controlled by microHAM control”.

54.3 *micro*HAM macro examples

54.3.1 uham macro

$**uham xx**$ works in the Radio Control Panel, where xx is the appropriate *micro*HAM op code. There is no need to terminate the command with <CR> as detailed in the MK2R user manual: Logger32 does this for you.

54.3.2 Example macros

Read the MK2R+ manual - Appendix B to understand these examples …

- $**uham SA1**$ Select function F1
- $**uham SA2**$ Select function F2
- $**uham FT1**$ Set TX focus to radio #1
- $**uham FT2**$ Set TX focus to radio #2

if controlling the MK2 DVK via the control port then:

- $**uHam MPm**$ Playback message (m) playback where m is between 1 and 9.

To control the 4 bit outputs from the ACC socket:

- $**uHam AS204**$ Sets R2 outputs to the binary value of 4 (pins 10,11,12 and 13)

- or -

- $**uham AM10001000000000000**$ Sets R1 outputs to the binary value of 3 (pins 6 7 8 & 9)

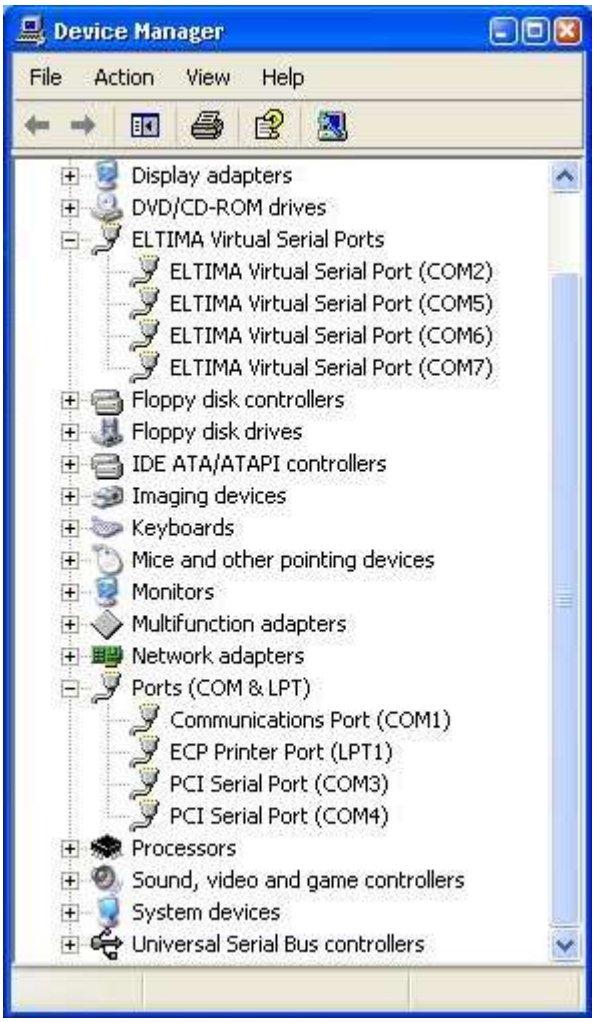
Copyright © 2026 Robert C. Furzer K4CY

54.4 Interfacing Logger32 to a *micro*HAM Keyer and FT-1000MP

These are notes about one ham's setup, to demonstrate what is possible. He can:

- Use PTT, foot switch or VOX-operated SSB.
- Use PSK31 and/or (AFSK) RTTY through the Sound card data window.
- Use PSK31 or FSK RTTY with a very small change to the setup.
- Operate CW from the built-in keyer in Logger32, supplemented by a WinKeyer and paddle.
- Set up for other operating modes using software other than Logger32.

54.4.1 Basic starting point



Most decisions taken during the setting up of my arrangement rested on two main items:

1. The hardware (RS232) ports that I had available in my computer.
2. How I wanted to operate RTTY (FSK or AFSK).

Station hardware: my computer has one built-in RS232 port (COM1) plus a PCI card that gives me two more ports (Com 3 and Com 4). The virtual serial ports seen in the above table are all creations of the microKEYER USB support software (see below).

I decided to try to run *all* of my system via just one USB port with the following port allocation. This is not a prerequisite at all: simply a personal desire as, in my case, this would leave me with 3 hard-wired COM ports for any other way I wanted to use them.

As for RTTY operation, I like to use AFSK. I do mention below, however, how to set up for FSK should that be your particular preference.

54.4.2 Connecting the radio

Follow the instructions that come with the equipment. If you bought the FT-1000MP interconnect cable, plug it in!

54.4.3 Setting up the software

This is where the fun starts……

First, follow the instructions given in the microKEYER instructions on how to load the basic software including the drivers *etc*.

Copyright © 2026 Robert C. Furzer K4CY

You now need to establish which ports you are going to use for which function. I chose the following:

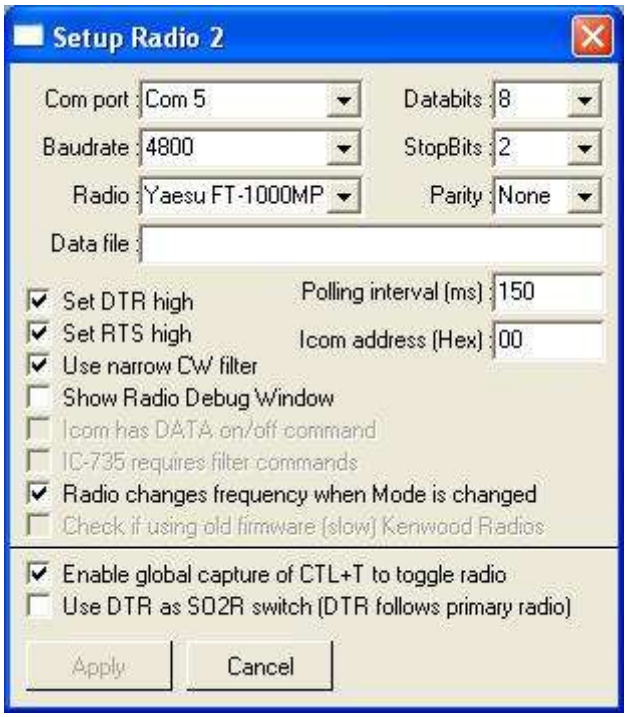
- COM 2: WinKeyer software (if needed).
- COM 5: Radio CAT interface.
- COM 6: CW keyer.
- COM 7: sound card PTT.

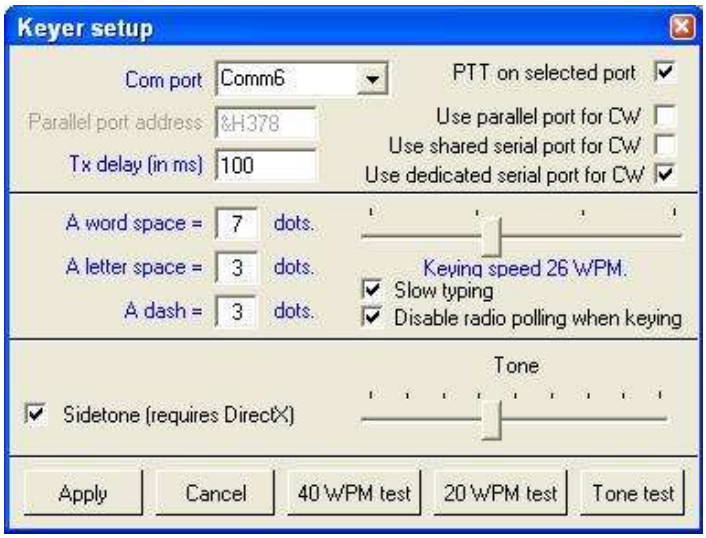
There is nothing hard and fast about these allocations. You can select what you wish but having made that selection I suggest you stick with it until you gain more experience with what is possible. Having made this decision, both Logger32 and the USB driver software need to be set up to match. In my case, COM ports 1, 3 and 4 are reserved for items such as a TNC, rotator control or anything else requiring a hard-wired COM port. (I actually use one as a loop back port so I can send DXspots type messages back to myself for test purposes)

54.4.4 Logger32 configuration

- Setup the CAT port to control the radio ▶

The FT-1000MP will respond to a fast poll rate so I have selected the fastest I can achieve with my radio that gives satisfactory results. In my case this is 150 ms.





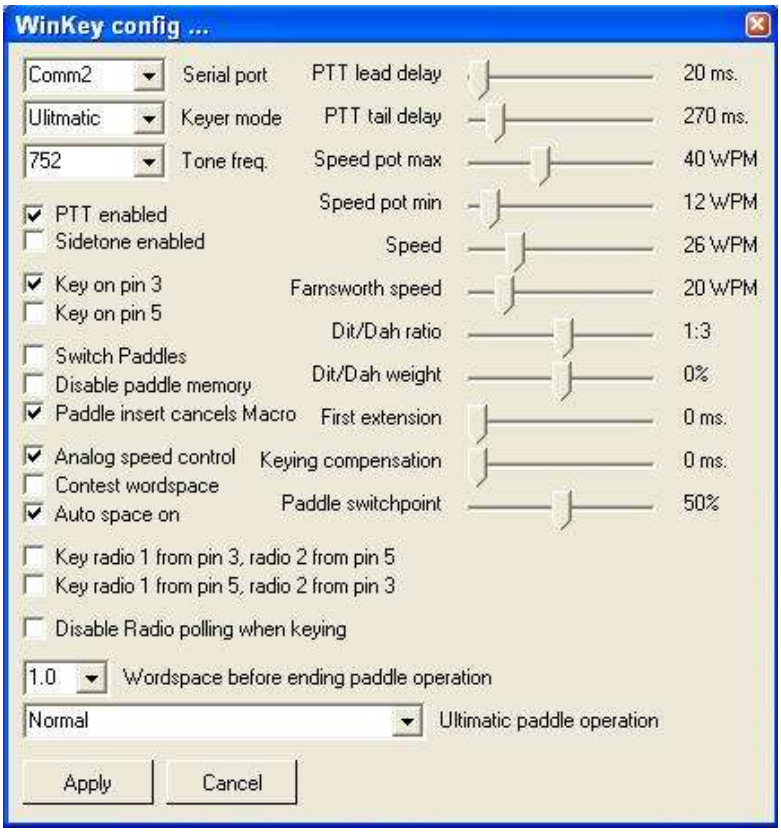
• Setup the CW Keyer: Logger32 supports WinKeyer and since the introduction of that support there have been two keyer setup windows.

◀The first, shown here, is for the original software keyer.

Not much more to do apart from setting the COM port to 6 and select <**Use dedicated serial port for CW**>.

Copyright © 2026 Robert C. Furzer K4CY

The second setup is for WinKeyer ▶



Information on all of these settings can be obtained from the WinKeyer manual and CW Machine chapter.

WinKeyer pin 5 is multi-functional pin. Logger32 can use this pin for various functions:

- None.
- PTT.
- Side tone.
- Second CW output.

MicroKEYER allows only the first two options, but Logger32 can use them all. Inside the microKEYER is a jumper marked SO1R/SO2R - a jumper for pin 5. If it is in the SO1R position, pin 5 is used for PTT. If it is in the SO2R position, pin 5 is disconnected from the PTT circuits but connects to the REMOTE mini DIN 6 connector (signal will be used in SO2R extension for microKEYER as second CW source).

When Logger32 opens the WinKeyer serial port, it takes full control of WinKeyer including PTT on pin 5. If you do not want PTT generation from WinKeyer when Logger32 is running, un-tick the PTT checkbox in Logger32.

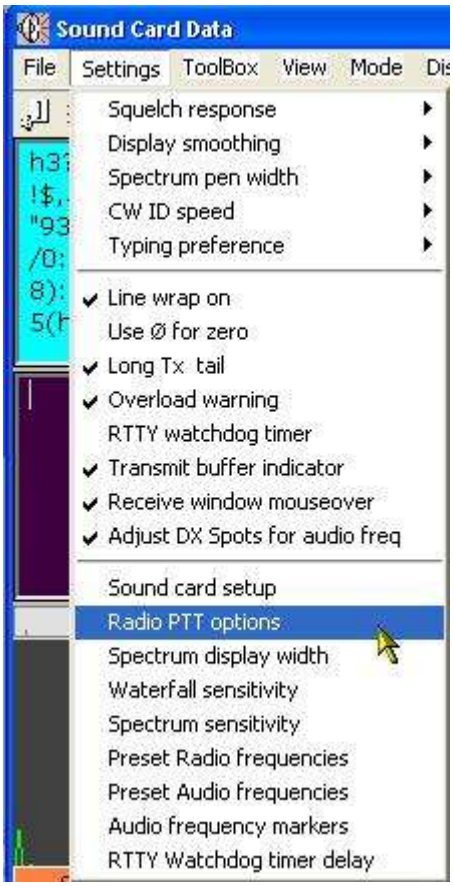
Have been using, enjoying and appreciating Logger32 as my regular logger for almost ten years. Not only are Logger32 users are extremely fortunate to have this software available, but also to have almost instantaneous availability of its developer, K4CY, in the event that help is required. In my experience, Bob is – by far – one of the most responsive and capable developers of ham radio software. Not only that, he is patient and will continue to hold your hand no matter how dense you may act. This I know from personal experience. Thank you, Bob!

Al K6RIM

Copyright © 2026 Robert C. Furzer K4CY

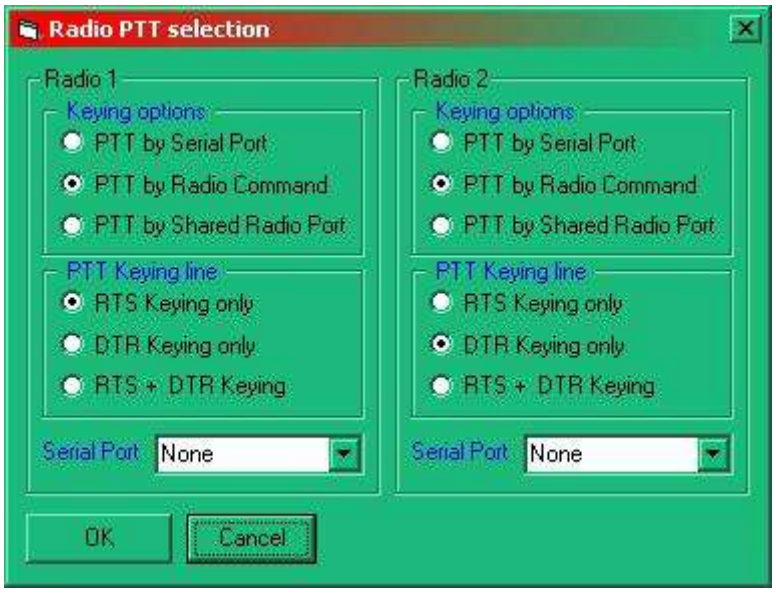
54.4.5 The sound card data window

For the Sound card data window there are two places that need to be set up.



First, select **Settings ⇨ Radio PTT options ⇨ PTT by serial port** otherwise FSK will not become available as described later. Select RTS keying and set the port to Com 7.

That sets up the AFSK side of Logger32 *i.e*. PSK31 and/or RTTY using audio tones.



If you are happy to use AFSK for RTTY, under the **RTTY Setup ⇨ Misc tab** (see below) the TX port has the "sound" radio button should be marked. To use FSK for RTTY, check the COM-TxD (FSK) radio button then click the USB port button and select "C". Changing between AFSK and FSK is as simple as that!



This completes the setup of the Logger32 software.

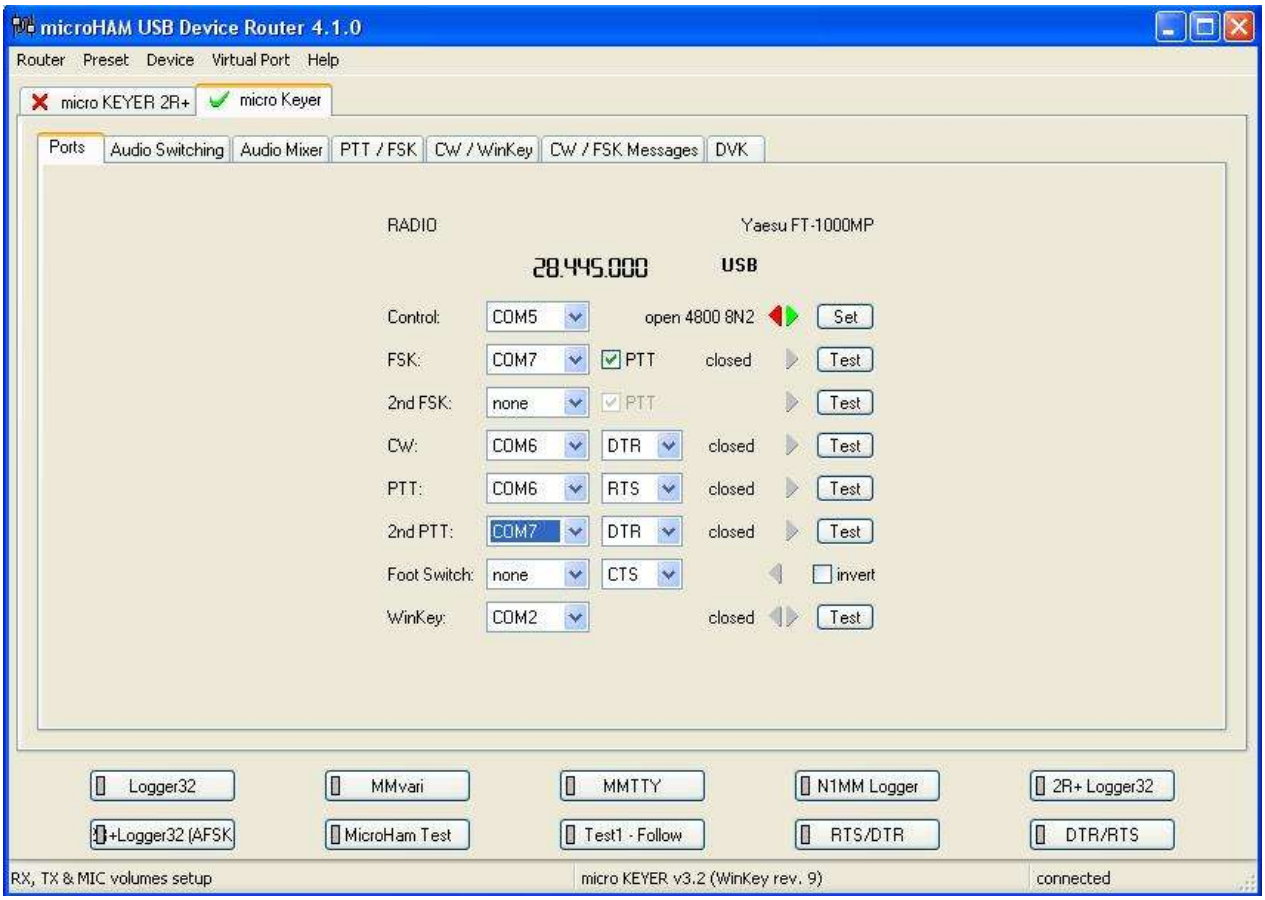
Copyright © 2026 Robert C. Furzer K4CY

Note: The Normal /Reverse button in the Communicating with RTTY section of Logger32 inverts the TX and RX data if you are using AFSK but not if using FSK.

54.4.6 USB software

Setup the *micro*HAM USB software as follows:

1. Click **Virtual ports ⇨ Create**
2. Select ports 2, 5, 6 and 7 in turn and <**OK**> them.
3. Open the <**Ports**> tab and configure the table as below.

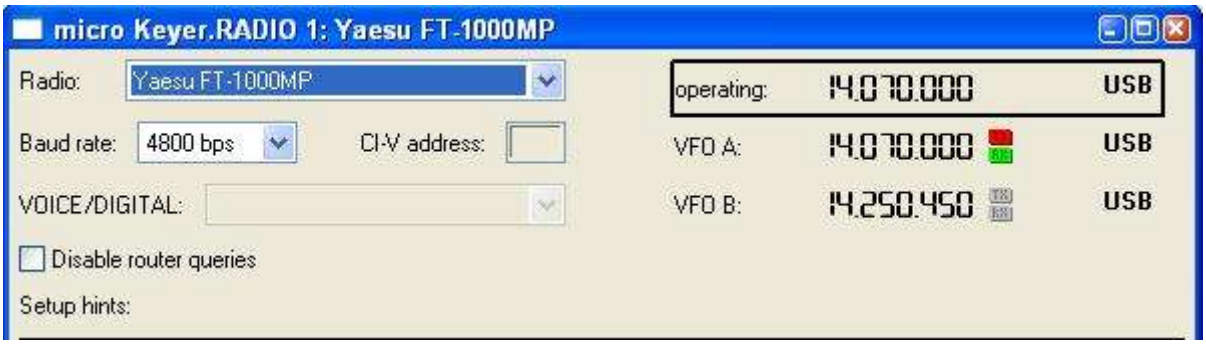


Notes:

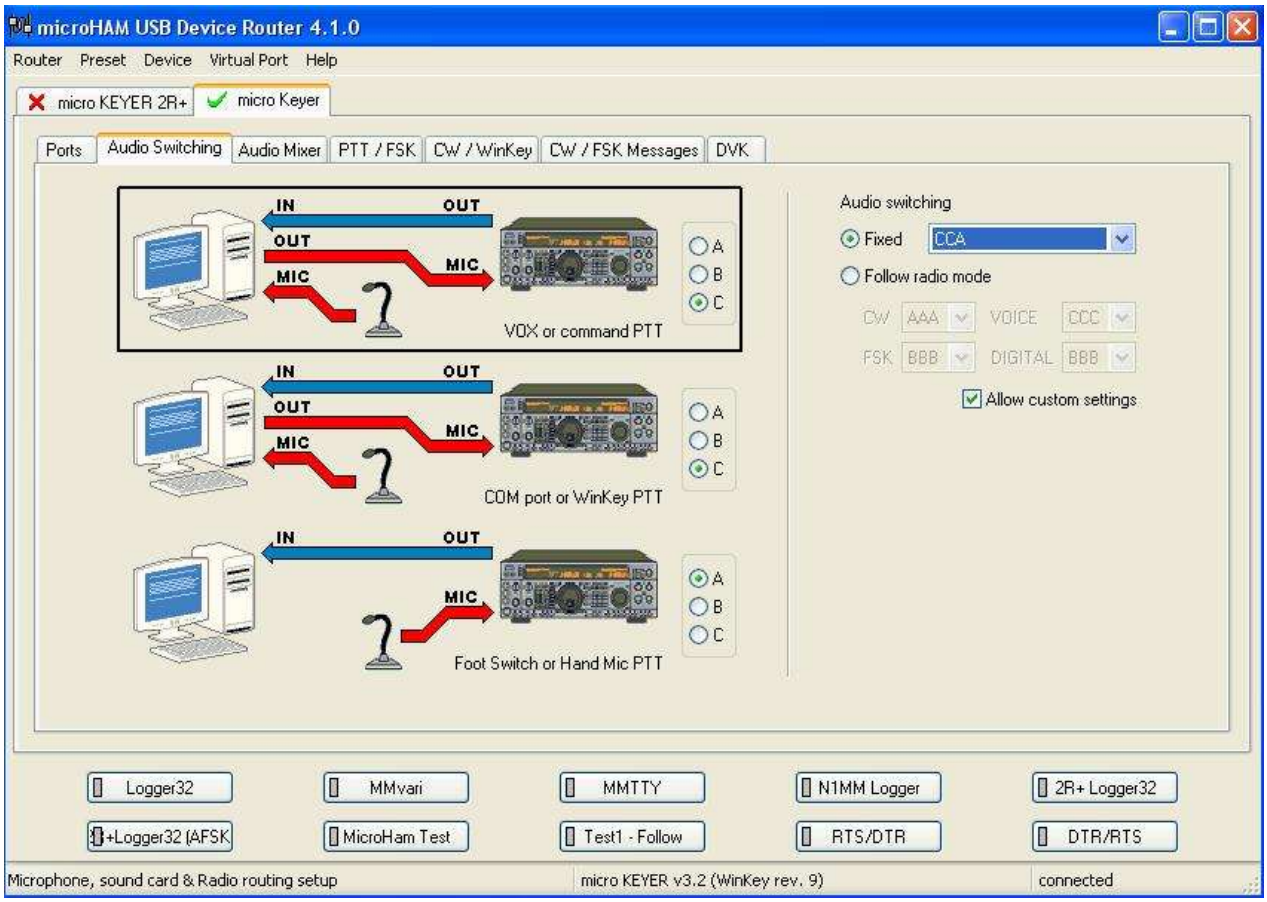
- If you have any software that can use WinKeyer, make sure it uses COM port 2.
- For Logger32 the PTT should use either RTS or CTS, but *not* RTS* or CTS*.
- If using FSK RTTY from MMTTY, the FSK port in the router must match that for the PTT.

Copyright © 2026 Robert C. Furzer K4CY

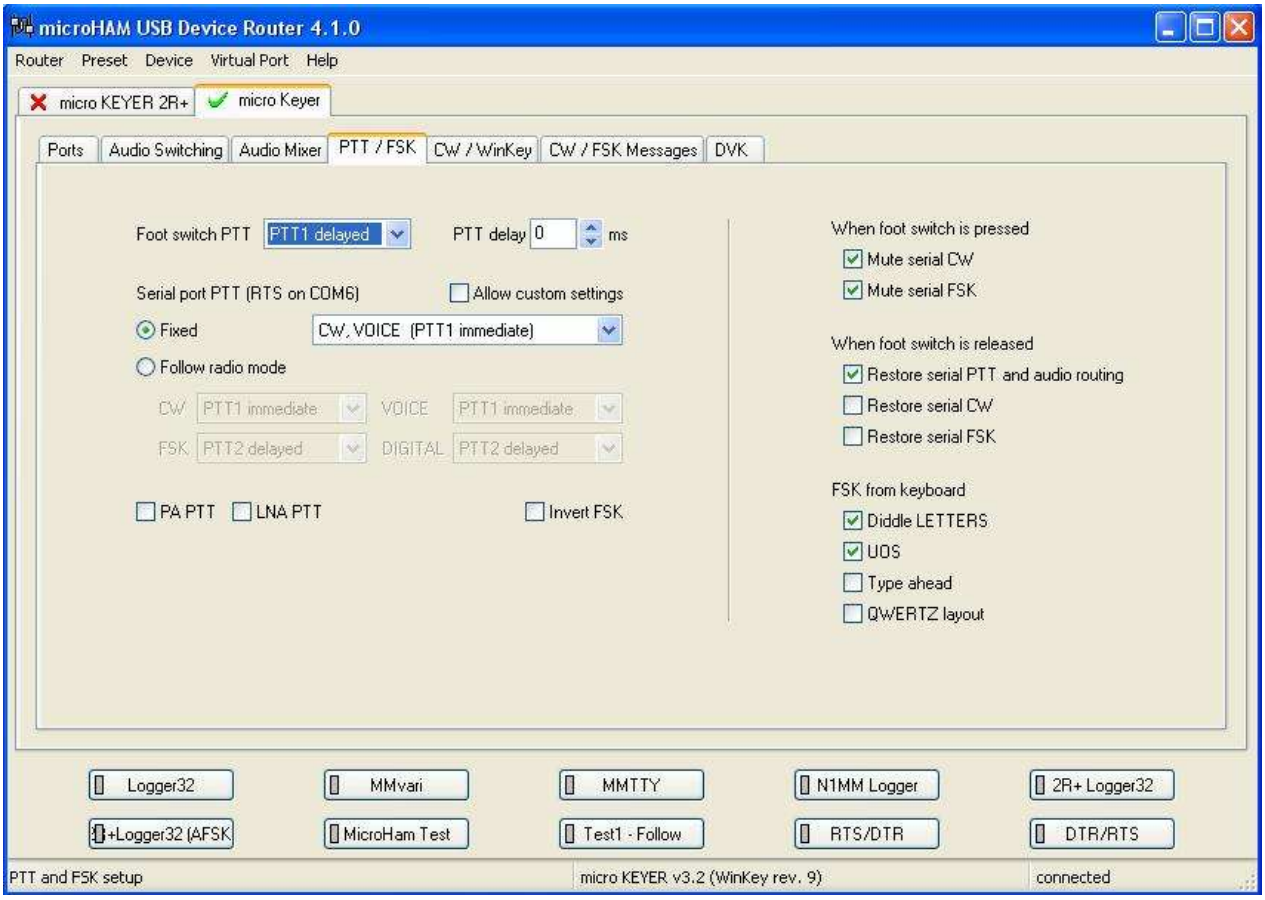
Make sure you have the correct radio set up in the router software. Click <**Set**> for Radio 1 control, to obtain the window shown below. Select the radio type, baud rate and if necessary the CI-V address.

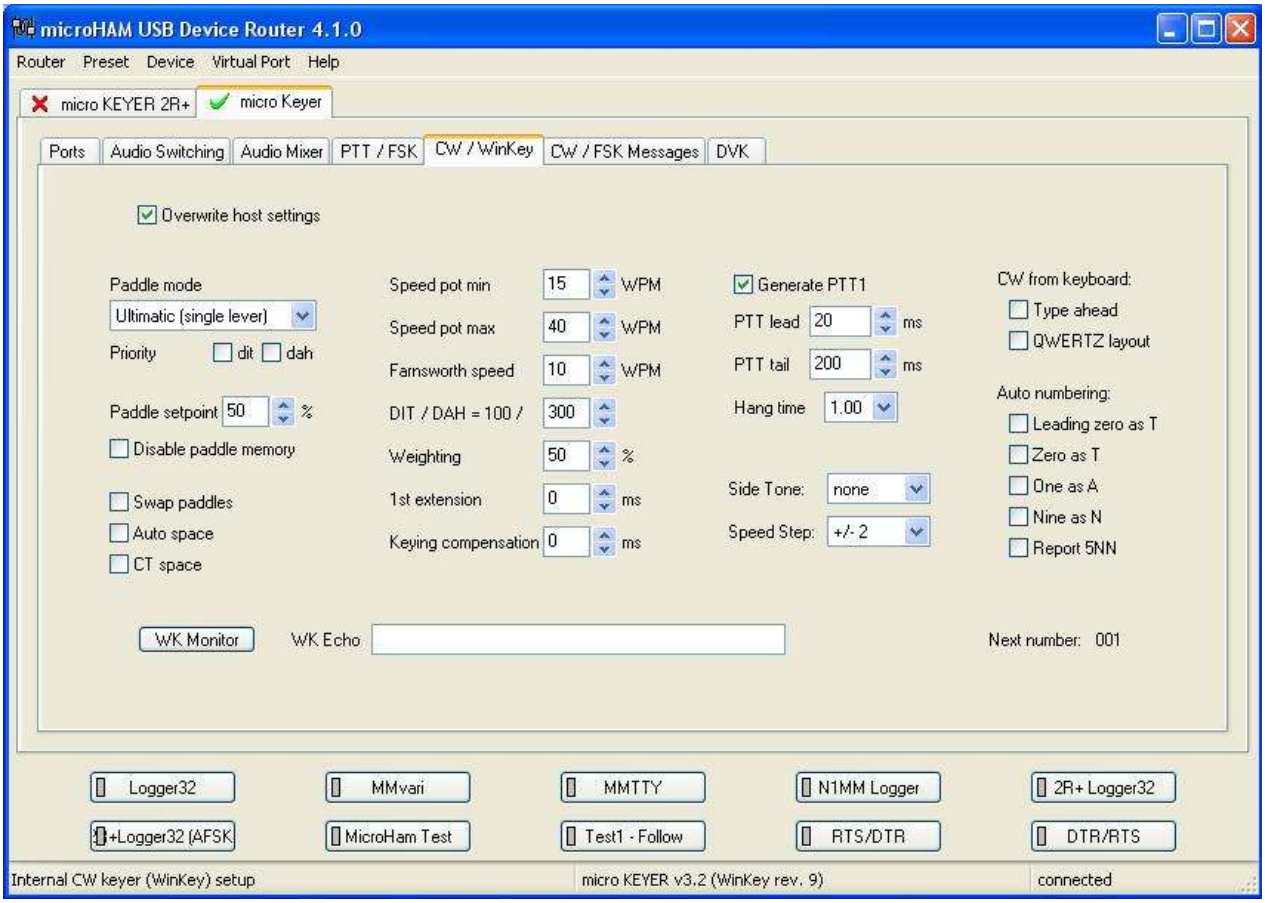


Check all the other tabs in turn. I show below my setup for the audio, PTT/FSK and CW/WinKeyer.



Copyright © 2026 Robert C. Furzer K4CY





Copyright © 2026 Robert C. Furzer K4CY

55 *Programmable* USB VFO knobs

> "I'm a knob person.
> I like twiddling knobs."
>
> *Zeena Parkins*



Several commercial suppliers offer external control knobs to adjust the PC's volume level, and switches for multimedia control. Most are simple USB 'plug and play' devices meaning that current/recent versions of Windows *should* recognize them – as volume controls – as soon as they are plugged in.

Some (but not all[481]) are **programmable**, which means we can link them via Logger32 to our CAT-connected radio rather than the Windows sound and multimedia functions that most non-hams evidently buy them for.



Here is a "Drok"-branded model on Amazon as this section is being written (end of 2025) ▶

Features:

- Rotary encoder with push switch
- Black or chrome knob
- Small or large size
- USB A connector
- ***PROGRAMMABLE!***
- ~US$35 from Amazon

When your Drok arrives, check that it works:

- Plug it in to a USB port.
- Turn the Drok knob clockwise: the PC's audio volume (as shown on the screen by the speaker icon and as heard by the Windows *ping* or the music playing) should increase.
- Turn the knob counterclockwise: the volume should decrease.
- Unplug the Drok ready to reprogram it for use with Logger32.

Hinson tip: if the knob won't even change the PC volume, there seems to be a problem with the device itself – maybe it has already been reprogrammed, strangely? Maybe it is dead-on-arrival.

[481] If yours is *not* programmable, you *may* be in luck: see the FAQ below. GL!

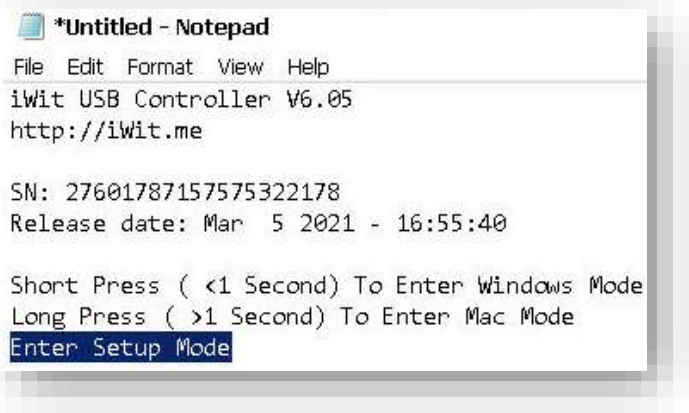
Copyright © 2026 Robert C. Furzer K4CY

55.1 Reprogramming the Drok

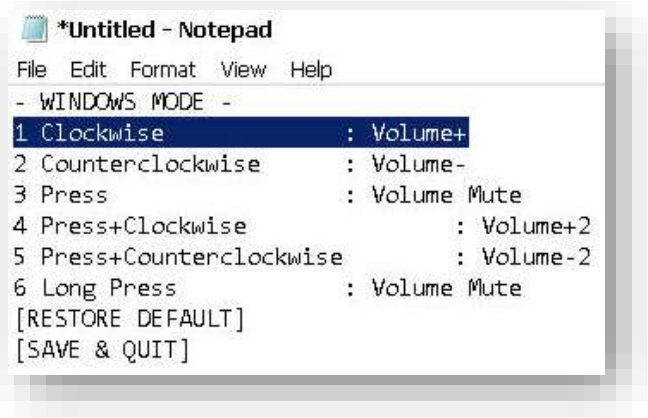
In order to tune the primary radio's VFO-A HF or LF via CAT using the keyboard, Logger32 expects <**Shift**+**Ctrl**+**Alt**+**L**> or <**Shift**+**Ctrl**+**Alt**+**R**> key sequences respectively. We therefore need to reprogram the Drok to send those obscure sequences[482] when the knob is rotated clockwise and counterclockwise, instead of the volume up/down commands it sends by default.

To put the Drok into Windows (programming) mode:

1. Launch a plain ASCII editor on your PC – preferably a modern replacement for MS Notepad such as TED Notepad or Mery[483].
2. With the focus on the empty editor window **press-and-hold the Drok wheel down as you plug it in**.
3. This text should magically appear in the editor window ▶



4. Like it says, "Short Press" (click) the knob to enter Windows [programming] Mode.



◀ You will now see something like this.

The Drok is now emulating the mouse, so you will be using the Drok - *not* the mouse - to navigate its programming options.

Hinson tip: from here on out don't touch the mouse, only the Drok knob. The focus must remain on the editor window throughout. If you nudge the mouse and move the focus by mistake, move the mouse pointer to the editor window, click it to regain focus … and stop fiddling with the mouse!

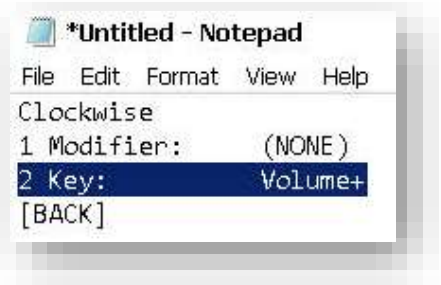
[482] These are *deliberately* obscure keyboard sequences to reduce the possibility of accidentally QSYing the radio when we are doing other things on the PC, such as using email … or editing the Logger32 User Manual in Word. Once configured correctly, these sequences will be sent by the Drok when the knob is turned and captured globally by Logger32, so whether Logger32 is running in the background or foreground, the radio should obediently QSY. We are unlikely to QSY the radio accidentally.

[483] Evidently, MS Notepad is a single-threaded app that struggles to receive *and* process the key sequences sent by the Drok, resulting in lost or garbled characters. Multi-threaded/higher-performing ASCII editors do better, apparently. You may need to experiment with different editors if this is a problem for you.

Copyright © 2026 Robert C. Furzer K4CY

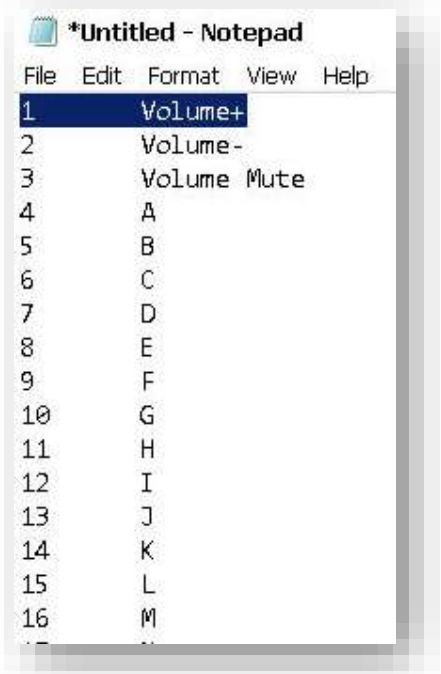
5. First, reprogram the clockwise knob rotation to send the <**Shift**+**Ctrl**+**Alt**+**L**> key sequence required to move VFO A HF.

   The Drok had already selected <**1 Clockwise : Volume+**> for us, so click the knob open this ▶



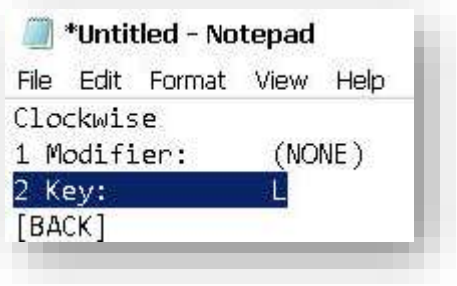
   Time to get the 'feel' of the knob. Rotate the knob slowly back and forth, clockwise and counterclockwise. You will feel light detents when turning. The highlight on the Notepad window jumps a line with each passing detent.

   OK, enough playing, let's get programming.



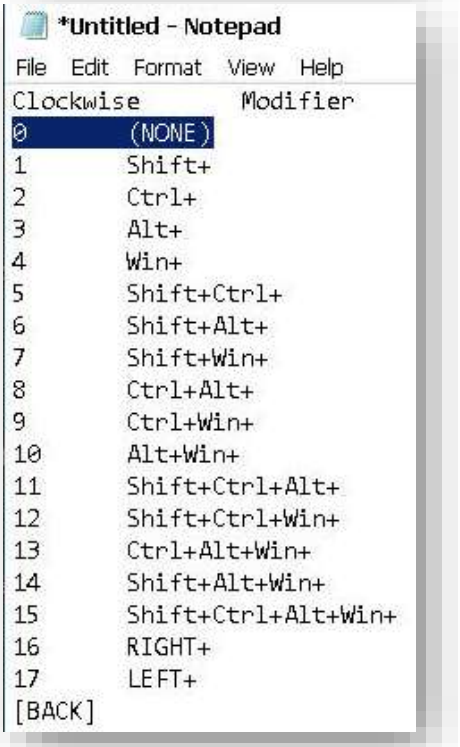
6. With <**2 Key: Volume+**> selected, click the knob ▶

7. Turn the knob past several detents to highlight the **L** on line 15, then click the knob to select it ▼

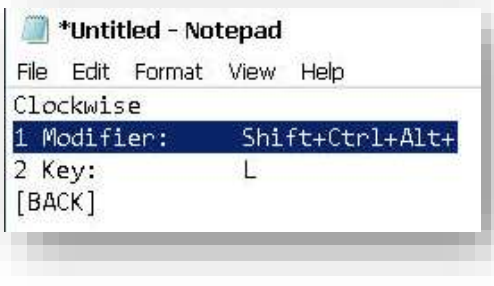


8. Turn the knob to highlight <**1 Modifier: (NONE)**> then click to change the rest of the key sequence.

   You now have a list of available modifiers ▶

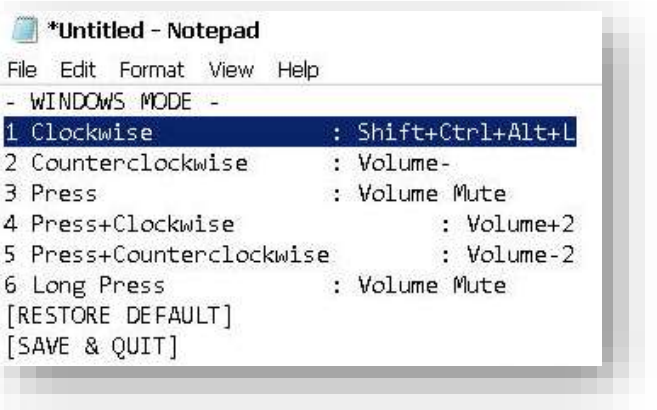


9. Turn the knob to highlight <**11 Shift**+**Ctrl**+**Alt**+> then click it, ending up here ▼

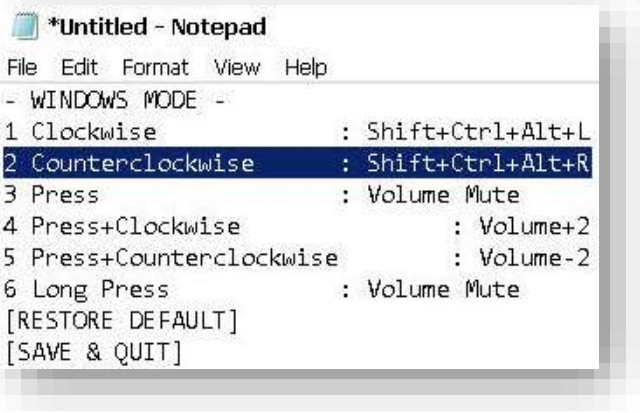


Copyright © 2026 Robert C. Furzer K4CY

Well done! You have just reprogrammed the Drok to send the <**Shift**+**Ctrl**+**Alt**+**L**> key sequence to the PC with each clockwise detent. On receiving that sequence, Logger32 will dutifully compose and send the CAT command to QSY your primary radio's VFO-A a little HF.



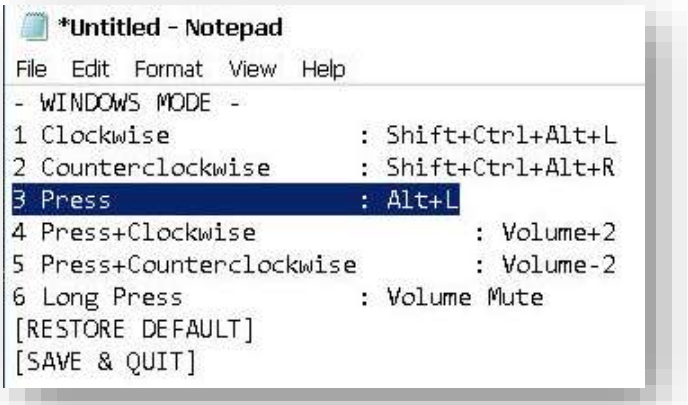
10. Turn the knob to select **[BACK]** then click it to get back to here ▶



11. You're flying solo now. In much the same way, select and change line 2 to send the QSY LF sequence <**Shift**+**Ctrl**+**Alt**+**R**> when turning the Drok counterclockwise … ending up like ◀ this

Almost done!

12. Select row <**3 Press: Volume Mute**> and reprogram it to send <**Alt**+**L**> which will lock and unlock the Drok. You have programmed lines 1, 2 and 3 in the Drok menu like this[484] ▶



13. Finally, select **[SAVE & QUIT]** and click the knob. You're done reprogramming the Drok.
14. Unplug the reprogrammed Drok.

[484] If you operate SO2V, you may also wish to reprogram the DROK to send <**Shift**+**Ctrl**+**Alt**+**U**> and <**Shift**+**Ctrl**+**Alt**+**D**> sequences in order to QSY VFO *B* when *pressing-and*-turning the Drok.

Similarly, maybe there is yet another Logger32 hotkey you'd like to trigger with a <**Long Press**> of the Drok, as opposed to <**Volume Mute**>? Be our guest. Knock yerself out.

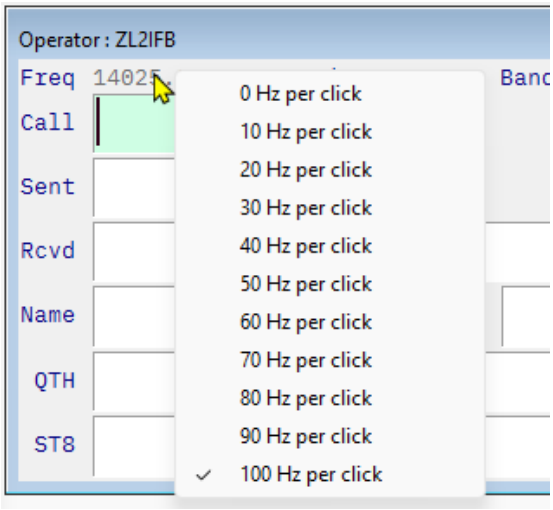
Copyright © 2026 Robert C. Furzer K4CY

55.2 Configure Logger32 to use the Drok

Using the mouse now …

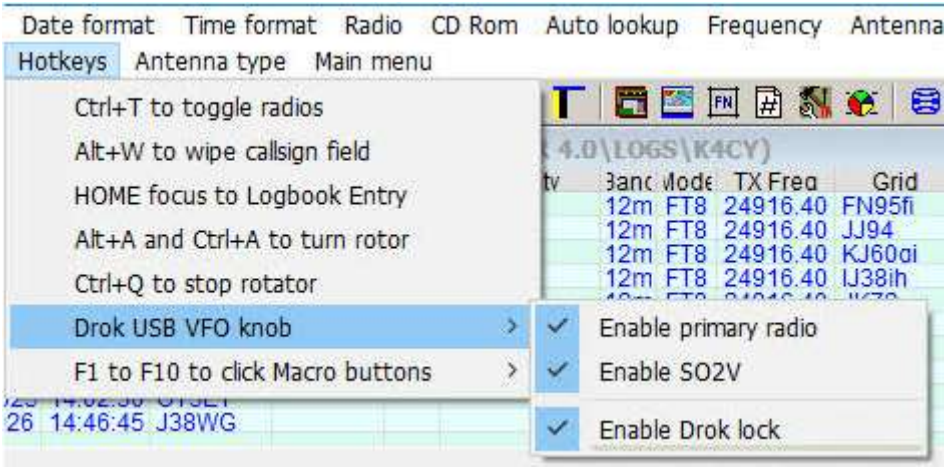
15. Launch Logger32 if it's not already running.
16. On the Log entry pane, right-click the frequency (the numbers, *not* the word "Freq") and click to tick your choice of VFO QSY rate.

When turning the Drok knob, each detent will QSY by, say, 100 Hz ▶



Hinson tip: try lower values for finer tuning. You can also hold down the left arrow key while turning the Drok to reduce the QSY rate a little, or the right arrow to speed it up a bit.

17. Finally, using **Setup ⇨ Hotkeys ⇨ Drok USB VFO knob**, click to tick all three options <**Enable primary radio**>, <**Enable SO2V**> and <**Enable Drok lock**>. ▼



55.3 Test your Drok programming skills

18. Plug the Drok back in to a USB port.
19. Click the knob. A message should appear near the PC clock at the bottom right of the screen, telling you the Drok is locked. As such, turning the knob will not send messages to the PC or via Logger32 and CAT to the radio.
20. Click the knob again and the message should say "Drok unlocked". Now, as the final test, does turning the knob tune the radio's VFO up and down the band? If so, pat yourself on the back.

Hinson tip: if not, what do you see in the radio debug window? Are the expected CAT QSY commands being sent … to the correct radio? Maybe unplug the Drok and go back to step 1 to check/correct its programming.

Copyright © 2026 Robert C. Furzer K4CY

55.4 USB VFO knob FAQ

Q. What other cool things can I do with my knob in Logger32?

A. Using the same approach, you can program the Drok to send other hotkey sequences to Logger32 *e.g.* <**Shift**+**Ctrl**+**Alt**+ **U** or **D**> will tune radio 1 VFO-**B** *N* Hz **U**p or **D**own, respectively.

Q. Can I still use the Drok to control the PC volume?

A. Yes. You won't even need to reprogram it or reset it to the defaults if you followed the instructions above. Notice in step 12 there are options to adjust the volume markedly by clicking-and-turning the knob. If you reprogram those from <**Volume+2**> and <**Volume-2**> to <**Volume+**> and <**Volume-**>, click-and-turn will adjust the volume less dramatically, like normal. Similarly, you can retain the default Drok click setting to mute and unmute the PC speaker as usual.

Q. Where are the drivers for these?

A. They are supposedly plus-and-play (self-installing) devices so you shouldn't need to install drivers, *provided* that (a) you are using a compatible version of Windows, preferably a current, supported version such as Windows 10 or 11. They will *probably* be OK on Windows 8 or 8.1, but *maybe not* on Windows 7, XP, Vista or other truly ancient version; *and* (b) they are well-engineered and manufactured. That's quite a lot to expect of a device costing $*notalot*.

Q. Can I make my own USB knob?

A. Be our guest! There are some intriguingly simple designs on the Web for USB-connected volume knobs if you are willing to have a go at making your own custom version to suit Logger32. Here's one that claims to be open and customizable, plus another on Instructables. They basically consist of a rotary encoder, Arduino-style microcontroller and maybe a few LEDs in a box … but programming the microcontroller to send the appropriate USB device ID and key sequences could be a challenge. *Please* let us know how you got on for the benefit of other DIY VFO knob-twiddlers out there.

Q. Will other similar USB volume knobs work?

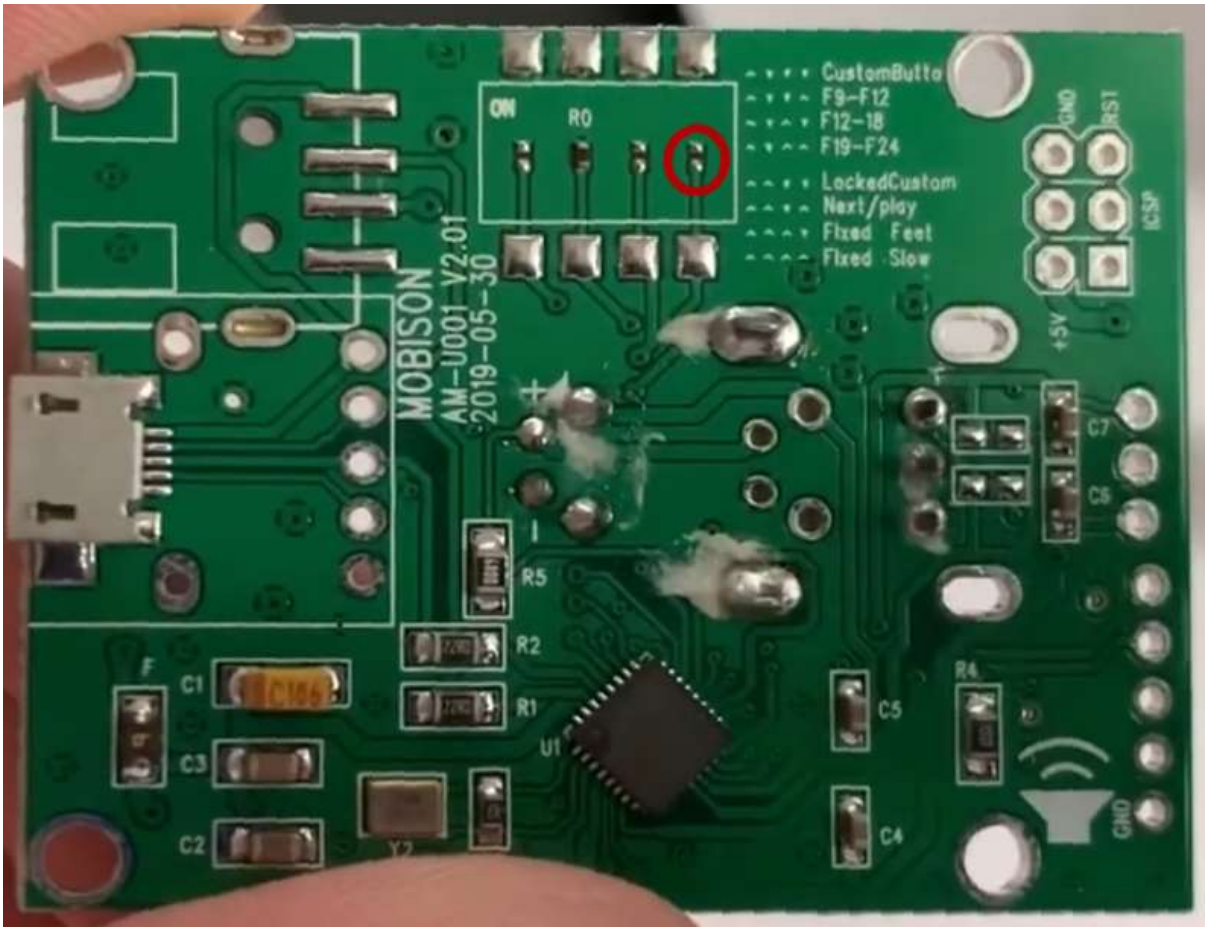
A. If they are ***programmable***, maybe – we don't know. Why not give it a go, following the knob's programming instructions? Whether you succeed or fail[485], *please* let us know what model of knob you used. We will update this manual accordingly.

[485] Failing to remote-control your VFO via Logger32 is not a total disaster: you *should* still have a functional PC volume control knob.

Copyright © 2026 Robert C. Furzer K4CY

Q. Can I use a *non*-programmable USB volume knob?

A. Maybe. Open the knob's case to take a look at the PCB. If you are lucky, your USB volume knob uses the standard microcontroller chip and PCB, in which case you can bridge the two solder pads outlined in red to enable it to be reprogrammed ▼



If your PCB or chip is different, you may be out of luck.

Copyright © 2026 Robert C. Furzer K4CY

56 Bits on the bottom

> "The bottom is a place of discovery and transformation"
>
> *Brendon Burchard*

This chapter is a motley collection of items lacking homes in earlier chapters.

56.1 The Logger32 team, a global community of volunteers

Logger32 simply would not exist without the generosity, skills and diligence of a substantial global team of selfless amateurs, contributing their efforts to the project. Here's a biased and somewhat jaundiced glimpse behind-the-scenes for a typical month:

"The latest release involved a making set of programming updates and tests last month. Bob sent us (the beta test crew) new beta releases to check roughly every day or two. Each one no doubt involved a couple of hours of work by Bob on average, ranging from maybe 20 minutes for a trivial bugfix up to several hours of thinking, intensive coding, alpha testing and further coding changes, then another few minutes to package up the release, briefly describe the change/fix and deliver it. Beta testing typically involves us downloading and installing the new beta, then using Logger32 normally and, usually, checking out the identified fixes/changes to confirm whether they work OK, giving feedback and discussing our findings as a team, sometimes leading to further program changes and betas – and so the cycle continues until we're all reasonably happy with the result (or exhausted!). The DXpedition function involved liaising with the crew behind DX-World's DXpedition timeline function, collaborating to pass the appropriate information to Logger32 users in the right format at the right time. Occasionally, changes don't work out and have to be reversed. Occasionally, technical limitations of the platform or database or whatever have to be investigated and worked-around (if feasible), necessitating further changes and testing … Anyway, as the changes stabilise and the monthly release deadline approaches, updating the manual takes me an hour or three – gathering screenshots, revising the text, adjusting the page layout and cross-references, updating the index etc. – plus maybe half an hour a day of general fiddling around with the wording based on using the software for DXing (I'm lucky enough to have time for that: Bob doesn't have the luxury). Meanwhile, we do our best to deal with requests, queries and complaints on the reflector, and spend the odd moment thinking up and discussing other possible changes including those suggested by users."

Gary ZL2iFB

Copyright © 2026 Robert C. Furzer K4CY

Developers: these are the coders slaving away on the programs running behind your screens …

JA1NLX Aki Yoshida
K4CY Bob Furzer (the boss!)
N2AMG Rick Ellison
VE7TK Rick Williams
VK3SU Mike Mace
W1MCP Charles Sutton
W2RU Bud Hippisley

Beta testers: teasing out, characterizing and reporting flaws, bugs and other issues is a necessary part of software development. These are Logger32's version 4 beaters, its crash test dummies (lots of testing, occasional crashes and no, these brave hams are *definitely* not "dummies"!) …

BD4EPV Joseph Jiang
DK7ZT Bernd Mueller
DL5VZ Alexander Schwinn
EA4AFP Jos Peña
EA7JXH Nick Sheard
EI9FV Gerry Lawlor
EW4DX Igor Artsyman
F5LMJ Alain Tuduri
F6EXV Paul Granger
G1XOW Steve Wragge
G4LMW Rob Thomson
G4RRM Pete Walker
G4UJS Rob Harrison
G7SSE John Munton
GI4DOH Richard White
GW4FRX John Nelson
GW8SZL Dave Phillips
HA1DCQ Feri Lengyel
I2SVA Alessandro Salvatico
I5FLN Luciano Fusari
IK2AGX Dave Lavelli
IK5PWC Luca Civinini
IN3ASW Gianni Fattore
JA1NLX Aki Yoshida
K2NF Neil Ferri
K4CY Bob Furzer (chief tester, analyst, breaker … and fixer-upper)
K5DHY Bill Engel
K8YYY Larry Fravel
LA8AW Odd-Egil Heradstveit
N2AMG Rick Ellison
N3ND Dan Atchison
N5KD Pete Hicks
N8DC David Colliau
OK2PAD Zdenek Habala
ON7NQ Danny Mees
PA4JJ Jan Buitenhuis
S51J Janez Majdic
S57U Polde Kobal
SP7DQR Marek Niedzielski
SV1VS Nick Plumidakis
TF3VS Villi Sigurjonsson
UA1AKE Constantin Semyonov
VA7RY Irek Kwapich
VE1ANU Mike Costelo
VE1RSM Bob Schofield
VE3EP Ian Morrison
VE6LB Gerry Hohn
VE7TK Rick Williams
VP8NO Mike Harris
VU2PTT Prasad Rajagopal
W1AJT Art Tolda
W2RU Bud Hippisley
W3GQ Paul Sturpe
W4/VP9KF Paul Evans
W4LDE Ron Walters
W4UCK Jim Altman
WA2UET Stan Engel
YT3W Dragan Acimovic
ZL2iFB Gary Hinson

Translators: making the program and this manual accessible to all involves a lot of hard graft interpreting and translating the English original, producing DLLs and documentation in a variety of languages. The following stars graciously assist/support their countrymen as well …

Chinese: BD4EPV Joseph Jiang
Czech: OK2PAD Zdenek Habala
Dutch: PA4JJ Jan Buitenhuis
French: F8ADY Franck Kerdoncuff
German: DL5VZ Alexander Schwinn
Hungarian: HA6FQ Papp Dénes
Icelandic: TF3VS Villi Sigurjonsson
Italian: I5FLN Luciano Fusari
Japanese: JA1NLX Aki Yoshida
Polish: SP7DQR Marek Niedzielski
Russian: UA1AKE Constantin Semyonov
Serbian: YT3W Dragan Acimovic
Slovak: OM3JH Jozef Hacaj
Slovenian: S57TUB Polde Kobal
Spanish: K8YC John Scott

Copyright © 2026 Robert C. Furzer K4CY

Country database maintenance: Bud Hippisley W2RU.

Group moderators and website: Prasad Rajagopal VU2PTT (Logger32.net – official download site) and Bud Hippisley W2RU (Logger32 reflector – which retains its original name "Hamlogger").

Logger32 documentation team:

4X1WQ Dr. Avi Rochman
CT1ILT Filipe Lopes
DL6KAI Randolf Schmitz
G0TSM Darren Collins
G0WZY Mike Davies
G3YYD David Wicks
G7SSE John Munton
IZ4AFW Fabio Mantovani
JA1NLX Aki Yoshida
K1DBO Don Beaudry
K3UK Andy O'Brien
K4CY Bob Furzer
K4ITO Curt Wann
K4JAF Jim Cox
K4SET Scott Thile
K8YC John Scott
KB1ZQ Hal Miller
KX2A Jan Ditzian
M0ZAK Jim Steel
N2AMG Rick Ellison
N2TU Lou Dietrich
N4GM Grant Mitchell
SP2EWQ Alec Otulak
SP7DQR Marek Niedzielski
VA3WI Barry Winch
VE6LB Gerry Hohn
VE6MRT John Torneby
VE7TK Rick Williams
VE9DX Andy McLellan
VK3SU Mike Mace
VU2PTT Prasad Rajagopal
W1MCP Charles Sutton
W2RU Bud Hippisley
W4LDE Ron Walters
W7TI William R. Turner
WA2UET Stan Engel
WA5UFH Randy Tipton
WB2RHM Ben Antanaitis
WB2SMK Mark Anderson
WB4MNK Art Petteway
WO4D Orrin Delany
ZL2iFB Gary Hinson (ed.)

Silent keys - former contributors: although now sadly QRT, we remain eternally grateful to the following for their generous legacy:

Antonio (Toni) Perez, EA1NK (SK 2006)
Jose (Joe) Valdes, YV5LIX (SK 2006)
Ray McClure, W8CNL (SK 2008)
Eike Barthels DM3ML (SK 2011)
Javier Bermejo EA1AUS (SK 2017)
Geoff Anderson G3NPA (SK 2018)
Jorge Enrique Knull LU4YAO (SK 2019)
Arne Andersson SM5CNQ (SK 2019)
Jim Hargrave W5IFP (SK 2022)
Jim Pickett K5LAD (SK 2024)
Hew Lines VA7HU (SK 2024)

"The truth is, gaining wisdom is kinda like climbing a mountain. You keep climbing the front side and you're surprised at the minimal incline angle. Suddenly you reach a peak and as you start down, it's almost a sheer cliff. I just wish I could have taken a mental picture back when I had made it to the peak."

Jim K5LAD (SK)

Copyright © 2020 Bob Furzer K4CY

56.2 Acknowledgments

We extend our sincere thanks to:

- Makoto Mori **JE3HHT** for permission to incorporate the MMTTY and MMVARI digimode engines (soft modems).
- Aki Yoshida **JA1NLX** and Hal Kojima **JE4IVN** for their considerable assistance during the development of the radio control software for use under the Japanese versions of Microsoft Windows.
- Alex Shovkoplyas **VE3NEA** (Afreet Software Inc.) for supporting the integration of his Ham CAP propagation prediction freeware.
- Roger Hedin **SM3GSJ** for his invaluable assistance with the azimuthal map projection code.
- The RSGB IOTA committee (now IOTA Ltd.) for supporting the downloading of their "activity" XML files for use with Logger32.
- 425 DX News for allowing the use of their DX News and Calendar.
- DX-World, Col MM0NDX and Max IK8LOV for integrating their DXpeditions Timeline.
- Robert Chalmas **HB9BZA** for supporting Logger32 in allowing us to download and use his LoTW user database.
- Rene Schmidt **DF9GR** and Richard Desaulniers **VE2DX** for their assistance and testing during the integration of the support for the GS232B Az/Az rotator protocol.
- David Wicks **G3YYD** thanks Peter Martinez **G3PLX** for assistance with Windows and sound card programming, Rick Ellison **N2AMG** for help overcome a problem on the N1MM DI/MMTTY interface and Kok Chen **W7AY** for help on extended Nyquist filtering.
- Michael Wells **G7VJR** and the Club Log support team for integrating Club Log's DXCC prefix exceptions function.
- … and all of you who provided creative suggestions, corrections, amplifications, valuable feedback, tips and information on specific topics throughout the project. Long may it continue!

56.3 Glossary and abbreviations

- **ACK**: short for **ack**nowledge, the response flag telling the sender of a TCP message that it has arrived. The received message sequence number is incremented by one and sent back.
- **ADI** or **ADIF**: **A**mateur **D**ata **I**nterchange **F**ormat (or **F**ile) – an XML standard for exchanging QSO data between systems.
- **AFC**: **A**utomatic **F**requency **C**ontrol.
- **AFSK**: **A**udio **F**requency-**S**hift **K**eying.
- **AGC**: **A**utomatic **G**ain **C**ontrol.
- **A-index**: a geophysical measure of geomagnetic conditions.
- **ALC**: **A**utomatic **L**evel **C**ontrol.
- **AM**: **A**mplitude **M**odulation, also known jokingly as Ancient Modulation, this "legacy mode" reached its peak of popularity several decades ago. The audio signal from a microphone is amplified and used to modulate (vary) the strength of a carrier wave.
- **AMTOR**: **AM**ateur **T**eleprinting **O**ver **R**adio, a digital communications protocol using time diversity (essentially, signals are repeated for more reliable reception).

Copyright © 2026 Robert C. Furzer K4CY

- **AOS**: **A**cquisition **O**f **S**ignal – the timepoint at which a satellite appears over a station's horizon and hence is potentially visible and contactable by the direct line-of-sight radio path at that station.
- **App**: a computer **app**lication – a program such as Logger32 or Excel.
- **ARQ**: **A**utomatic **R**epeat Re**Q**uest detects (some) data errors and asks for them to be sent again.
- **ASCII**: **A**merican National **S**tandard **C**ode for **I**nformation **I**nterchange, the character set used in most IT systems. It uses seven digital bits to represent 127 letters, figures and control characters, or eight digital bits for 256 in 'extended ASCII'.
- **AX.25**: amateur radio version of the pro X.25 datacomms standard, known as packet radio.
- **Baud**: the number of signal changes (*e.g*. stepping between FSK frequencies) per second on a serial connection. With simple coding schemes such as RTTY, each signal change represents a single bit of information so baud = bps, whereas more sophisticated schemes such as FT# can communicate more than one bit per signal change so baud < bps.
- **BAD**: **B**roken **A**s **D**esigned – for practical reasons, a few Logger32 functions are imperfect.
- **BCD**: **B**inary-**C**oded **D**ecimal – uses 4-bit words to represent the 10 decimal digits. So, a decimal number such as 73 would be BCD coded as 0111 0011, not 1001001 which is actual binary.
- **Bit**: **B**inary digi**t** – represented as a one or a zero, on or off, up or down, heads or tails …
- **BPF**: **B**and-**P**ass **F**ilter – attenuates frequencies above and below the passband.
- **bps**: the maximum number of **b**its that can be transferred **p**er **s**econd on a serial link.
- **BPSK31/63/125**: **B**inary **P**hase-**S**hift **K**eying at different speeds.
- **Byte**: a sequence of 8 bits, usually, known also as an octet.
- **CAT**: **C**omputer **A**ssisted [or **A**ided] **T**ransceiver – a way to computer-control a radio via a data link such as a serial or Ethernet connection. Think small pause rather than small paws.
- **CI-V**: ICOM's fifth **C**omputer **I**nterface standard.
- **CNTY**: county such as Los Angeles, California or Berkshire, England.
- **COM**: serial data**COM**munications port.
- **CQZ**: **CQ z**one number – 1 of 40 areas of the world as originally determined by CQ Magazine.
- **CSV**: **C**omma **S**eparated **V**ariable – a file format where data fields in a record are separated by commas or sometimes tabs, semicolons or other designated characters.
- **CTS**: **C**lear **T**o **S**end – a hardware handshaking signal line in the RS232-C standard.
- **CW**: **C**ontinuous **W**ave – Morse code.
- **CW-R**: CW using the 'reverse' sideband, the opposite to normal (whichever that is!).
- **DATA-LSB** or **DATA-USB**: data transmission using LSB or DSB, respectively.
- **db**: decibel – a tenth of a Bel, which is a ratio of two quantities (usually power).
- **DB-9, DB-25, Centronics and USB ports**: different physical and electrical standards for serial or parallel data communications connectors.
- **Declination**: the variation in azimuth headings between true North (the precise spot on the ground in the Arctic which is centered directly over Earth's spinning axis), and magnetic North (the ground spot where the Earth's magnetic field lines converge, several hundred kilometers away from the true pole and wandering around by about 50km per year).
- **DIG, digi**: digital – typically denoted by either of two states *e.g*. +5 volts or ground.
- **DLL**: **D**ynamically **L**inked **L**ibrary – stored common functions used by programs as they run.

Copyright © 2026 Robert C. Furzer K4CY

- **DOK**: a German administrative district+chapter code as defined by DARC *e.g*. P03 is the DOK for Friedrichshafen. Reminiscent of an English parish or a US county.
- **DSP**: **D**igital **S**ignal **P**rocessing – analog signals are converted to/from digital for math magic.
- **DSR**: **D**ata **S**et **R**eady – a control line used for handshaking (flow control) on a serial connection such as an RS232 line. A slow device can pause the flow of data while it does things, slowly.
- **DST**: **D**aylight **S**avings **T**ime – local clocks in some countries are typically set forward by an hour in summer to "save" daylight. Presumably there is a big stash of it somewhere. UTC remains UTC forever … well except for the occasional leap-second adjustment anyway to correct for Earth's uneven spin.
- **DTR**: **D**ata **T**erminal **R**eady – an RS232 hardware handshaking signal for the other end of a comms connection to a device using DSR.
- **Dupe**: duplicate QSO *i.e*. one adjacent to another in the log, less than 60 seconds apart, with the same callsign, band, mode, operator, state, county, comment, name and gridsquare.
- **DVK**: **D**igital **V**oice **K**eyer – like a parrot that learns to squawk "Pieces of eight!" on command.
- **DX**: ??? – there is no standard or universally-accepted definition of this term! It is a shortened form of "distance", but how distant is distant? Not nearby, that's for sure, but beyond that?
- **DXCC**: **DX C**entury **C**lub – a prestigious, longstanding ARRL award scheme for making confirmed contacts with at least 100 'entities' (countries) on the DXCC list.
- **eQSL**: **e**lectronic **QSL** – an Internet-based system to check, match and hopefully confirm the details of QSOs that have been logged correctly and claimed by both parties.
- **FEC**: **F**orward **E**rror **C**orrection – a mathematical encoding/decoding scheme that compensates for various bit errors on a noisy or unreliable communications link using redundancy.
- **FFT**: **F**ast **F**ourier **T**ransform – a mathematical algorithm used to extract the frequencies of component signals within a compound signal (demodulation) or to compose a compound signal from the component parts (modulation).
- **FIR**: **F**inite **I**mpulse **R**esponse filter – a basic DSP method periodically samples the input data stream, multiplies the samples by specific coefficients, and sums the results to generate the output. Got it?
- **Flag**: binary value – can be raised or set to 1 or true *e.g*. to denote "Send a QSL", or lowered or *un*set/*re*set to zero or false when said QSL has been sent or is not required.
- **FM**: **F**requency **M**odulation – the frequency of a carrier is continuously modulated within a range to carry analog information.
- **FSK**: **F**requency-**S**hift **K**eying – the carrier frequency shifts (steps) between two discrete values (known as the 'mark' and 'shift' frequencies) to carry digital information.
- **FT#**: FSK modes invented by Steve **F**ranke K9AN and Joe **T**aylor K1JT to communicate short messages between amateurs, using clever coding, compression and modulation techniques to compensate for low power and marginal HF paths. FT8, FT4 and FT2 are presently popular.
- **Geomag**: **Geomag**netic – relating to the Earth's magnetic field.
- **GMSK**: **G**aussian **M**inimum-**S**hift **K**eying – a type of continuous-phase frequency-shift digital modulation. Gaussian shaping of the waveform reduces the bandwidth by smoothing out otherwise sharp amplitude transitions.
- **GMT**: **G**reenwich **M**ean **T**ime – an obsolete time reference linked to the Earth's rotation, superseded by UTC. Named from the royal observatory at Greenwich in East London where the

Copyright © 2026 Robert C. Furzer K4CY

Prime Meridian, a line connecting the poles at 0° longitude, is still physically marked on the ground by a brass strip. Go see!

- **Greenstamp**: a US dollar bill or some other banknote of similar value, although not necessarily green *e.g*. a Euro or pound note.
- **Grid square, Maidenhead locator**: a scheme agreed at a meeting in Maidenhead, England, to specify locations on the Earth's surface according to uniquely identified 'squares'[486]. The first two letters identify a 20˚ by 10˚ "field", followed by two digits designating a 2˚ by 1˚ "square" within that field. Further character pairs can be added for even greater precision.
- **G-TOR**: **G**olay **T**eletype **O**ver **R**adio – a digital communications protocol that provides an improved communication capability for the HF bands and increased throughput, compared to legacy protocols such as **TOR** (**T**eleprinter **O**ver **R**adio).
- **HELL**: **Hell**schreiber mode – a raster communication mode invented by Dr. Rudolf Hell in 1929. Also the place where pirates ultimately end up.
- **IMD, intermod**: **I**nter**M**odulation **D**istortion – a measure of the quality of a radio signal.
- **IOTA**: **I**slands **O**n **T**he **A**ir – an award scheme from the **R**adio **S**ociety of **G**reat **B**ritain that recognizes confirmed amateur contacts with numbered islands meeting specific criteria.
- **ITUZ**: **ITU z**one number (between 1 and 90 inclusive). These are *similar* but not precisely the same as the ITU's CIRAF zones, used for managing professional broadcasting arrangements.
- **JCC**: **J**apan **C**entury **C**ities – a JA award scheme.
- **JCG**: **J**apan **C**entury **G**unn – another JA award scheme.
- **JSON**: JavaScript Object Notation – an open standard data exchange format using attribute–value pairs and arrays.
- **Keps**: **Kep**lerian element**s** – a set of numbers mathematically defining the orbital paths of satellites. Named after the famous astronomer Johannes **Kep**ler.
- **kHz**: **kilohertz** – thousands of cycles per second. The lower-case 'k' for kilo meaning thousand also appears in other common units such as kV (kilo Volts) and kW (kilo Watts).
- **K-index**: an indicator of geomagnetic activity during the previous three hours on a scale of 0 to 9. A K-index of 0 indicates very calm geomagnetic conditions, while disturbed/storm levels begin at a K-index level of 4. A K-index of 9 is both highly disturbed and somewhat disturbing.
- **km**: kilometre – a thousand SI metres, or 'meters' if you prefer. Five furlongs in old money.
- **LCID**: **L**o**C**ale or **L**anguage **C**ode **Id**entifier – Microsoft's numeric code for the PC's language.
- **Localhost**: hostname for the computer on which the network app is running with the IPv4 address 127.0.0.1 or the IPv6 address ::1. Also known as the **loopback** address.
- **LOS**: **L**oss **O**f **S**ignal – the timepoint at which a satellite drops below your horizon and hence is no longer visible or contactable by the direct line-of-sight radio path from your station.
- **LoTW**: **L**ogbook **o**f **T**he **W**orld – ARRL's online electronic QSO matching and verification system. Free QSLs! See https://lotw.arrl.org/
- **LSB**: **L**ower **S**ide**B**and with suppressed carrier (and suppressed upper sideband!).
- **Macro**: programmable series of command instructions in the form of a script.

[486] Due to the Earth's shape, 'grid squares' are not *precisely* square. Despite certain claims to the contrary, the Earth is *not*, in fact, a flat rectangle. It has a *nearly* spherical surface but not an edge.

Copyright © 2026 Robert C. Furzer K4CY

- **MB**: **M**ega**B**yte – a million (10^6) bytes. The "M" abbreviation for mega meaning million is always a capital (upper-case) in SI units, whereas a lower-case "m" for milli means one thousandth.
- **Mbps**: **M**ega**b**its **p**er **s**econd – data transfer rate in millions of bits per second.
- **MBCS**: **M**ulti**B**yte **C**haracter **S**et, an alternative to Unicode, ASCII *etc*. for Japanese, Chinese or other languages with more characters than can be represented by a single 8-bit byte.
- **MDI**: **M**ultiple-**D**ocument **I**nterface, a Microsoft facility for applications to display multiple documents (outputs) at the same time in separate windows.
- **MFSK**: **M**ultiple **F**requency-**S**hift **K**eying stepping between more than FSK's two frequencies.
- **MHz**: **M**ega**H**er**z**: millions of cycles per second. That's a lot of cyclists.
- **MMTTY**: a modem program written by Makoto Mori JH3HHT that supports RTTY.
- **MMVARI**: a modem program written by Makoto Mori JH3HHT that supports RTTY and PSK.
- **ms**: millisecond (thousandth of a second), an SI unit of measure for time.
- **MT63**: a **M**ulti-**T**one frequency-shift keying mode developed by SP9VRC for two or more hams to converse. It uses 64 tones and FEC.
- **MUF**: **M**aximum **U**sable **F**requency. Over a given DX path, signals above this frequency are not effectively reflected by the ionosphere.
- **NCDXF**: **N**orthern **C**alifornia **DX F**oundation – see www.ncdxf.org
- **OS**: **O**perating **S**ystem – such as Windows, Android or Linux.
- **OQRS**: Club Log's **O**nline **Q**SL **R**equest **S**ervice – see clublog.org/oqrs.php
- **OTRSP**: **O**pen **T**wo **R**adio **S**witching **P**rotocol – a protocol for controlling SO2R devices.
- **PACTOR**: a commercial digital communication mode that dynamically adapts to the path or link quality. If a PACTOR system receives a series of imperfect or incomplete data segments, it 'intelligently' attempts to reassemble them into a solid group.
- **PAD**: **P**acket **A**ssembler/**D**isassembler – device converts between streaming data and the discrete packets for X.25 or AX.25/packet radio, and manages the link.
- **PCB**: **P**rinted **C**ircuit **B**oard – uses copper traces on/in a phenolic, fiberglass or plastic board/sheet to connect electronic components soldered to pads, forming a circuit.
- **Pfx**: the **p**re**fix** part of a callsign, usually indicating the entity or country.
- **PKT**: packet radio *i.e*. AX.25.
- **PROP_MODE**: VHF propagation mode as defined in the ADIF standard.
- **PSK31, PSK63, PSK125**: digital communication modes using **P**hase-**S**hift **K**eying at different rates.
- **PTT**: **P**ush-**T**o-**T**alk – ground this control line to start your transmitter transmitting.
- **QPSK**: **Q**uadrature **P**hase-**S**hift **K**eying – a digital communications mode.
- **Reflector**: service receives emails from members of a group and automatically sends 'reflected' copies back out to all other members *e.g*. the Logger32 reflector.
- **RS232**: the common name of the EIA-232-D standard for serial data communications, usually meaning the third and final revision C (officially, "RS232C") with ±12-volt signal levels.
- **RTTY: R**adio**T**ele**TY**pe – an early digital communications FSK mode that alternates between two frequencies (the "mark" and "space") separated by a few tens or hundreds of Hertz.
- **RTTY-L**: **R**adio**T**ele**TY**pe using **L**ower sideband.
- **RTTY-U**: **R**adio**T**ele**TY**pe using **U**pper sideband.

Copyright © 2026 Robert C. Furzer K4CY

- **RX**: receive or receiver.
- **SDR**: **S**oftware-**D**efined **R**adio – uses reprogrammable DSP for signal processing.
- **Sequence number**: a serial number in TCP packets allows the recipient to combine them in the correct sequence, allowing for unpredictable network delays on successive packets.
- **SI**: **S**ystème **I**nternational d'unités – the 'metric' system based conveniently on the number ten, the preferred number of fingers on a human. Or toes for that matter.
- **SO2R**: **S**ingle **O**perator **T**wo **R**adios, a contesting/DXing technique where the operator transmits on one frequency while listening on another, alternating between two radios and (usually) two bands to maximize QSO rate and monitor for band openings.
- **Socket**: the combination of a host IP address and TCP port number necessary to route TCP network traffic to its intended destination.
- **Solar flux**: a measure of electromagnetic activity at the surface of the sun. The **S**olar **F**lux **I**ndex (SFI) is normally measured in **S**olar **F**lux **U**nits (SFUs) at a wavelength of 10.7 cm.
- **SSB**: **S**ingle **S**ide**B**and, a voice and data communications mode using either the upper (HF) or lower (LF) sideband of an AM signal, with the other sideband and the carrier both suppressed by filtering in the transmitter.
- **SSID**: **S**econdary **S**tation **Id**entifier. Appending a hyphen and number or letter to a callsign turns creates unique SSIDs, allowing multiple simultaneous DX cluster connections for the same station *e.g*. ZL2iFB-**1**, ZL2iFB-**2** and so on. By convention, numbers below 16 indicate the type of station, while 16 and up are free-choice. Letters are used by D-star.
- **SSTV**: **S**low **S**can **T**ele**V**ision, a raster communications mode to encode and send static images line-by-line as audio tones over a narrow-bandwidth HF link.
- **Straight cable**: short for straight-through cable *i.e*. pin 1 at one end connects to pin 1 at the other end, and so on for all the pins – or at least all those that are used.
- **Sub-receiver**: some radios have a second, subsidiary receiver in addition to the main one – either a complete second receiver from antenna input to audio output, or one that shares some circuits with the main receiver (*e.g.* its RF band-pass filters).
- **SYN**: short for **syn**chronize, a flag set on the first message in TCP's three-way handshake link establishment.
- **TCP**: **T**ransmission **C**ontrol **P**rotocol – actually a suite of networking protocols collectively known as TCP/IP. TCP is connection oriented (connections are made, persist while passing messages and then closed) and reliable (provides basic error detection and correction).
- **Terminator**: the boundary between day and night. Also an Arnold Schwarzenegger character.
- **TLE**: **T**wo-**L**ine **E**lement – a succinct Keplerian data format for satellites.
- **TNC**: **T**erminal **N**ode **C**ontroller combining an AX.25 modem with a PAD.
- **TTL**: **T**ransistor-**T**ransistor **L**ogic – uses 0 to 0.8 volts for a logical zero and 2 to 5 volts for a logical one, with intermediate "limbo" values being callously ignored.
- **TX**: transmit or transmitter. Or **T**e**X**as. Or treatment/therapy. Or **T**ELE**X**. Context-dependent.
- **UAC**: **U**ser **A**ccount **C**ontrol – a Windows security feature that warns you when an app asks for administrator rights to perform a privileged system function such as resetting the clock or meddling with important Windows folders and files.
- **UDP**: **U**ser **D**atagram **P**rotocol – a networking protocol allowing messages to be broadcast onto the network, without having to establish connections and acknowledge receipt like TCP. Low

Copyright © 2026 Robert C. Furzer K4CY

overheads and efficient transport make UDP useful for real-time audio-visual streaming that can tolerate a proportion of dropped or corrupted packets.

- **UOS**: **U**nshift **O**n **S**pace – a RTTY technique to revert to the *letters* character set automatically whenever a space is received even if the anticipated unshift command gets lost in the ether, resulting in lots of numbers where there should be letters.
- **UQF**: **U**niversal **Q**SL **F**ormat – a curious and now obsolete logging format. It remains available as a log export option in Logger32 for reasons lost to K4CY Bob's Famous Grouse fumes.
- **URL**: **U**niversal **R**esource **L**ocator – a system providing addresses for IT systems, devices, files, web pages and other resources on the global Internet, and perhaps elsewhere in the universe.
- **USB**: **U**niversal **S**erial **B**us – a family of serial datacommunications standards using low voltage signal levels *e.g*. 0/2.8V for USB 1 and 0/400mV for USB 2 or 3, plus power at +5V normally but negotiable up to +48V at higher current for fast-charging of some devices.
- **USB**: **U**pper **S**ide**B**and with suppressed carrier (and suppressed lower sideband!).
- **UTC**: Coordinated Universal Time – a globally-recognized and synchronized standard date and time system, maintained by reference to *extremely* accurate atomic clocks.
- **VFO**: **V**ariable **F**requency **O**scillator – one whose frequency can be varied intentionally.
- **WAD**: **W**orking **A**s **D**esigned – Logger32 does a lot, but not everything. There are limits.
- **WPM**: **W**ords **P**er **M**inute – a measure of the speed of a CW signal, determined by the number of standard words (by convention, "PARIS") that can be sent in one minute.
- **WPX**: the *Worked All Prefixes* award scheme, originally run by CQ Magazine.
- **XML**: e**X**tensible **M**arkup **L**anguage – a data format that data types and field lengths to be denoted *within* a data stream or file. It is extensible because new data types (*e.g*. modes and submodes in ADIF) can be defined by agreement between the communicating parties.
- **YMMV**: **Y**our **M**ileage|**M**etrics|**M**orse **M**ay **V**ary.
- **Zakanaka**: one of the first PSK programs written decades ago by Bob K4CY, way back in the days when Creed 444 teleprinters were all the rage and shacks smelt of warm oil, pipe smoke and flux.

56.4 Additional information online

Looking for even more information, definitive sources and additional help? Check out our favorite resources …

56.4.1 Logger32-specific resources

- The *official,* definitive Logger32 website is www.logger32.net. Go there for the installation files and the latest updates, for all manner of information about Logger32, and to find out about joining the supportive global community of users, beta testers and fans of Logger32.
- The Logger32 email reflector (still named "Hamlogger") lets you ask questions, report bugs or issues, and participate in the friendly global community of Logger32 users via email or through the Groups.IO web interface. Individual emails sent to the reflector address Hamlogger@Groups.IO by members of the group are broadcast out to all 3,000+ members of

Copyright © 2026 Robert C. Furzer K4CY

the group whereupon, after a brief pause to think, someone usually responds to the whole group in the same way[487].

- Aki JA1NLX has written several Logger32 utilities, describes updates to Logger32 in Japanese on his blog, occasionally translates this manual into Japanese, and generally does a *fantastic* job of supporting fellow JA users of Logger32. [Aki is one of a pool of hams generously supporting 'the locals' to get the most out of the program, in their own languages.]
- Rick N2AMG offers invaluable Logger32 add-on utilities as well, such as L32 LogSync.
- The Files area of the Hamlogger group ▶ at groups.io has numerous supplementary files created by Logger32 users over the years, such as secondary administration data files. With some notable exceptions, the files are generally unmaintained (and some are seriously out of date now) so you will probably need to update them, but even so they can save you a metric tonne of work and set you off to a flying start.



56.4.2 General amateur radio resources

- The **declination** (offset between true and compass North) for any QTH on Earth can be determined using the calculator at www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml
- For information on time zones, visit www.timeanddate.com/worldclock
- The US **N**ational **I**nstitute of **S**tandards and **T**echnology has useful information about using NTP and runs several US-based time servers.
- The NTP Pool project maintains redirect addresses for public NTP servers around the world.
- Google's public NTP time servers may be handy if you have a Google data center nearby (and who doesn't?).
- Check the accuracy of your PC clock very easily and simply by browsing to time.is

[487] If you intend to respond privately just to the originator of a comment or query on the reflector, use their personal email address, *not* the Logger32 reflector address Hamlogger@Groups.IO – unless you have something highly embarrassing and personal to reveal, in which case go ahead and tell us all. We'll probably enjoy a good laugh at your expense. Belated excuses and apologies are accepted. Also, please be considerate and patient: membership of the group is a privilege, not a right. Our members are dotted around the globe like "DX" spots during CQ WorldWide contest, while the Logger32 gurus are spread thin and far between like genuine top-ten DX spots.

Copyright © 2026 Robert C. Furzer K4CY

- Club Log acts as a clearing house for logs from DXers and DXpeditioners, with callsign lookup and propagation prediction tools, DXCC tracking, club league tables and a single-screen dashboard *e.g*. https://clublog.org/dx-dash/ZL2IFB.
- CwGet is a CW decoder for the hard-of-Morse.
- **I**slands **O**n **T**he **A**ir is an award scheme for contacting stations on various islands and island groups around the world.
- **N**orthern **C**alifornia **DX F**oundation runs the **I**nternational **B**eacon **P**roject.
- Ham CAP propagation prediction software.
- VOACAP lets us make our own propagation predictions including short- and long-paths to selected 'DX sites' (*e.g.* DXpeditions).
- Contest info from Contesting.com.
- WA7BNM's diary of upcoming contests is also available as a convenient Google calendar.

56.4.3 Electronic logging resources

- Find out about the **A**mateur **D**ata **I**nterchange **F**ormat for exporting and importing log-related data between and among ADIF-compliant logging programs, such as Logger32
- For *free* online callsign lookups, HamQTH is the option built-in to Logger32. Is there an up-to-date record for *your* callsign? That could be useful info for the people you contact.
- Information on ARRL's **L**ogbook **o**f **T**he **W**orld – login to LoTW here and download TQSL here
- If you don't need Gary ZL2iFB's LoTW new user guide, Dave AA6YQ's succinct LoTW quick start guide for Windows gets straight down to business, no messin'

56.4.4 Satellite resources

- **T**wo-**L**ine **E**lement files from the PE0SAT and AMSAT websites are suitable for the Tracking window's satellite map.
- AMSAT is the global organization for all the **am**ateur **sat**ellites

56.4.5 DX cluster resources

- Lee VE7CC's CC User software makes connecting to and configuring DX clusters easy – basically just point and click. For the 'official' download, the bcdx.org website has fallen off the web but you might find copies of the installer at www.NE6I.com/CC-User-Ver3.0271.zip or k2dbk.com/CCuser/ or elsewhere.
- DXCluster.info has an extensive list of Telnet-accessible DX clusters
- Paul G4BKI, W4/VP9KF *etc*., occasionally updates his Telnet DX cluster info

56.4.6 Digital mode resources

- FT8 Operating Guide by Gary ZL2iFB – no longer maintained/updated I'm afraid
- Getting Started on RTTY by Don AA5AU
- MMTTY classic RTTY engine, and MMVARI multimode engine

56.4.7 Interfacing and radio control resources

- Info on interfacing between sound card and transceiver from Ernie WM2U

Copyright © 2026 Robert C. Furzer K4CY

- WinKeyer documentation by Steve K1EL

56.5 Logger32's history

Logger32 is the product of an ongoing experiment. Way back when Marconi was a bambino and Hertz was cycling to school, Bob Furzer K4CY (formerly N6BFM) wrote one of the first experimental digimode programs **Zakanaka** - an African word meaning 'excellent' or 'heavenly'. Needing some way to log the QSOs, he developed a simple **DOSlogger** and then a QRO version **Logger807**.

As interest grew and the experiment proved valuable, Bob enlisted the assistance of other amateurs in creating the **HamLogger** package. Work started in July 2001 with a team of volunteers coordinated via the Interweb to help with the specification, development, testing and support of the program, drafting and reviewing the documentation and so forth.

The Hamlogger name persists to this day in the online support forum (Logger32 reflector).

When HamLogger was updated from a 16- to a 32-bit database engine, its name morphed into **Logger32**. Logger32 has been continually maintained and improved ever since, reaching version 3.5 in 2019 where it stabilized for a bit pending development of the next major version …

Logger32 version 4 was developed and thoroughly beta-tested during 2020, a welcome diversion from COVID-19 (remember that?). It included another database update, new serial port drivers and various functional enhancements. A pilot study at the end of 2020 led to a full public version 4 release in January 2021. By April 2023, Logger32's source code had exceeded 141,000 lines.

New numbered subversions of version 4 are released as auto-updates as soon as they have been tested – usually at the start of the month. Development, testing, bug-fixing and documentation happens continually behind the scenes: Bob and the intrepid "beater-testers" slave away over one or more beta versions most days, thrashing them in order to discover what still works and what doesn't. Your scribe does his level best to understand and explain new/changed features, frequently tweaking the wording and updating images in this User Manual the better to describe stuff. At least, I *think* I'm making it better, but patently I'm biased. In summary, Logger32 and this very manual are being *actively* improved and updated. Well, updated anyway. They are *definitely* updated, for sure, no doubt about that. No siree.

56.6 Logger32 copyright, licensing, terms and conditions

Logger32 and this manual are copyright © 2026 Robert C. Furzer.

Authorization to use the software is limited to radio amateurs who agree to comply strictly with the following operating practices:

- Any questions or comments pertaining to Logger32 that are not covered by this manual should be addressed to the Logger32 reflector and ***NOT*** to Bob Furzer directly.
- Know the callsign of the station you are calling *before* you call.
- *Always* use your full callsign when CQing or calling another station.
- Pointedly ignore callers who do not use their full callsigns.
- Do not s-p-e-l-l o-u-t your name or QTH u n n e c e s s a r i l y.
- Do not call the operator of the station you are in contact with 'Sir', 'Your Highness', 'Doctor', 'Monk' or any other title (real or imaginary).
- Do not begin your transmissions with 'QSL', 'Roger', or 'Roger Roger'.

- Do not end your transmissions with 'QSL', 'Roger', 'Over', or 'Over over'.
- Do not operate on DX list or DX net[488] operations (nor condone those that do).
- Do not transmit SSB with filters more than 2.8 kHz wide.
- Do not use devices deliberately but unnecessarily to distort your audio, beyond what the radio technology and ionosphere already do.
- Do not precede an exchange of signal reports with "Please copy ...," "When last heard...," or any other superfluous colloquialism. We eschew superfluous colloquialisms.

In general, Logger32 is free for use in amateur radio, but is not licensed for reproduction on CDs or websites other than those selected and explicitly authorized by Bob Furzer. **Reproduction of this software on other CDs, elsewhere on the Internet or in any other form or media, is expressly prohibited.** Please don't waste Bob's time or yours by asking: the answer is a resounding *NO!* This is necessary to maintain a single, definitive and up-to-date source at Logger32.net (well, two actually: the entire website is replicated across two servers for resilience and performance).

Do not print out this manual! Printing is expressly prohibited. It is specifically designed to be used on-screen as an electronic document, updated every month in line with the software. Listen up, you rogue, you arboreal anarchist! You *dare* consider printing this digital masterpiece? Have you no heart, no soul, no shred of concern for the fluffy polar bears clinging desperately to their shrinking ice floes? Consider this your official warning: printing this document even double-sided, requires *at least a ream of paper*, an act of environmental terrorism, punishable by a lifetime subscription to Greenpeace emails and forced viewings of Al Gore and Greta Thunberg documentaries. Every sheet you print is a personal attack on the adorable lungs of wide-eyed toddlers and the vibrant foliage of the Amazon rainforest (soon to be the Amazon dustbowl, thanks to you). Can you live with the bloodcurdling whimpers of overheated puppies echoing in your dreams? Can you stomach the haunting image of the equatorial regions transformed into desolate deserts, desperately devoid of DX? Think again, paper fiend! This manual was lovingly crafted for the digital realm, where trees frolic freely and ink cartridges don't weep. Do yourself, the planet and all its oxygen-sucking inhabitants a favor: **embrace the digital revolution, you eco-traitor!**

> *"I took the time and effort to write the software. Many others have taken the time and effort to debug the software and write the documentation. At the very least, you can make the effort to comply with the terms of the use of this software. If you choose not to operate in accordance with the requirements listed above, please do not use this software."*
>
> *Bob K4CY*

[488] To be clear, this is nothing to do with setting Net **On** in MMTTY or MMVARI. That's OK. So is *the* net, meaning the Internet, to some extent anyway. Please stick to RF and the ether for ham contacts.

Copyright © 2026 Robert C. Furzer K4CY

Unfortunately, the *official* IOTA database cannot be incorporated into Logger32 for copyright reasons. The *unofficial* IOTA information within Logger32 is derived from public (non-copyright/public domain) sources, and extensive contributions from individual amateurs. The accuracy of Logger32's IOTA information is uncertain as it has not been officially validated by IOTA officials in the IOTA office. You are free to modify Logger32's IOTA information as you see fit, or indeed to validate and update it for us.

Much the same applies to other data within Logger32. We have done our level best to ensure it is accurate and up-to-date but we do not guarantee that. If you spot errors and omissions (including problems with this User Manual), *please* let us know. Do us all a favor.

Elecraft holds the copyright on some of the images in the Elecraft chapter which are used with permission. Other images, text snippets and quotations in this manual are included under the fair use provisions of copyright law, are in the public domain (including the one at the bitter end ▼▼) or were prepared specifically for this document (such as the numerous screenshots from the authors' Logger32 systems) and as such are covered under the same copyright terms as the manual as a whole. We gladly acknowledge our sources. Thanks all!

56.7 Logger32 guarantee and loyalty scheme

In the highly likely event that Logger32 does *not* fully and totally satisfy your every whim, operate to your complete satisfaction and impress you with its sheer unbridled brilliance in every conceivable respect, you are free to uninstall Logger32 from your system or systems, then request and wait patiently by your postbox for your no-questions-asked money-back cast-iron guarantee payout. Simply furnish your original receipt and your Logger32 license fee will be refunded forthwith, less a nominal re-stocking fee.

Conversely, in the unlikely event that Logger32 meets and perhaps even *exceeds* your wildest dreams, you are of course free to describe and recommend it enthusiastically to your peers (*e.g*. on eHam.net), for which you will earn Logger32 loyalty points. Collect a sufficiency of points to qualify for the opportunity to apply for the dubious honor of taking over the maintenance of this manual.

Please.

Copyright © 2026 Robert C. Furzer K4CY

Appendix: Logger32 color config cheat sheet

> "Life is about using
> the whole box of crayons"
>
> *RuPaul*

We can choose most of Logger32's colors. When (not if!) we forget how to change the color of something, this cheat sheet leads us directly to the corresponding instructions:

- Award tables *e.g*. banded shading of alternate rows; colored backgrounds for deleted entities
- Background colors for forms, scrollbars, progress bars, status bar and data fields, plus various "grids" (tables) such as the logbook and DX Spots pane.
- BandMap zoom color to distinguish zoomed-in from zoomed-out
- BandMap line colors connecting spotted callsigns to the frequency scale
- BandMap DX spots have the same text colors as DX spots in the DX Spots pane
- Beacon marker colors on the NCDXF map – also the beacon table text
- Blobs denoting LoTW/OQRS users on Log entry, DX Cluster/DX Spot panes and BandMaps
- Callsign CD lookup colors
- CW Machine colors
- DX cluster pane colors for Talk and Announce messages, tabs and boilerplate
- DX spot & BandMap colors including: highlighting of various 'new ones', specified callsigns, current DXpeditions, LoTW and OQRS users, and stations with unknown DXCC countries; DX spots received through each of the DX cluster pane's tabs; and the **X**'s for logged stations
- DX spot and home location marker colors on the Tracking window's maps
- DX spot tooltip popup colors (text and background)
- DXCC wanted highlight colors for DX spots of interest (for a selected DXCC award)
- Floating callsign colors, fonts and transparency
- Generic QSO table callsign-only/whole-row coloring
- Gridlines and row/column headings in the logbook and other tables – previous QSOs, worked/confirmed *etc*.
- Gridsquare colors on the UDP BandMap (highlighting wanted grids)
- GRITTY colors
- Highlighting colors (various) *e.g*. **W**orked, **C**onfirmed, **S**ubmitted and **G**ranted
- IOTA map colors
- JTDX Control Panel button colors
- Log entry pane colors including Dark Mode
- Logbook text and background color for the field currently being edited
- Logbook, previous QSOs and generic QSO table (grid) colors

Copyright © 2026 Robert C. Furzer K4CY

- Macro button colors for the JTDX Control Panel, MMTTY, MMVARI, RCP, DVK, CW Machine …
- Manual calling and cherry picker event viewer colors for the UDP BandMap
- Manual calling box color on the UDP BandMap
- Map (tracking window) tab colors plus sea, land, background and compass colors on the maps
- Map overlay contrasting shades *e.g*. nighttime, grayline dawn/dusk and daytime.
- Map spot colors – either the same highlighting as for the BandMaps or all one color
- MMTTY|MMVARI colors for the XY scope, waterfall, spectrum, signal and squelch levels
- MMTTY|MMVARI – color the type-ahead text about to be or already transmitted
- MMVARI colors for the Main and Aux windows *etc*.
- MMVARI MultiRX colors for callsigns, CQs, reports *etc*. on the multichannel decoder
- My location (station) marker color on the maps, also its size
- NCDXF beacon monitor colors for dead, dormant and transmitting beacons
- Pseudo spot text color for bookmarked callsigns
- QSO background colors in your logbook
- **R**adio **C**ontrol **P**anel – adjust the frequency & mode text colors
- RCP waterfall/spectrum colors
- Rotator direction indicator and wedge colors
- RTTY text highlighting when mousing-over received text
- Satellite path and footprint colors on the tracking window, plus the contrast levels
- Scratchpad colors for rows selected or being edited
- Selected or edited field colors in the log entry pane, scratchpad *etc.*
- Selected line in the logbook
- Split indicator colors (text and background) plus the font, text size, borders and positioning
- Telnet pane data entry panels for sending cluster commands and talk messages
- Worked, confirmed, submitted and granted lines on award reports plus the **W**orked, **C**onfirmed, **S**ubmitted and **G**ranted blobs in the award tables
- Worked/confirmed table band+mode text colors in the 'complex' view.

Hinson tip: Logger32's color settings are saved in the *.INI* files. You *can* edit them directly if you bravely claim to understand the parameters and decimal color values such as:

```
New Country=12434431
New Band/Mode Country=16509525
Need Confirmation This Band=16381875
Need Confirmation This Mode=16381875
Need Confirmation This Band/Mode=16381875
```

It is foolish to meddle in the *.INIs* without first making backups, however, so when you next have a bad day, you can easily put things back as they were *before* you started 'improving' them.

Hinson tip: Windows – not Logger32 – controls the appearance (*e.g*. colors, fonts, transparency, borders) of the window captions (title bars) plus the mouse pointer (*e.g*. its shape, size, color, shadow). To change them, right-click the Windows desktop then click **personalize** and fiddle around with the colors, themes, fonts *etc*. or ask Google.

Copyright © 2026 Robert C. Furzer K4CY



With apologies to Mel Brooks' Blazing Saddles
and thanks to Google Gemini/Nano Banana AI

Copyright © 2026 Robert C. Furzer K4CY