**WSJT-X User’s Guide**

**Joe Taylor, K1JT**

Copyright © 2013

April 11, 2013

Contents

[Introduction 3](#_Toc332184432)

[Hardware and Software Requirements 3](#_Toc332184433)

[Installation and Setup 5](#_Toc332184434)

[Decoding Tutorial 11](#_Toc332184435)

[Additional Features and Options 14](#_Toc332184436)

[Appendix A: Installed and Generated Files 19](#_Toc332184437)

[Appendix B: The JT65 Protocol 20](#_Toc332184438)

[Appendix C: Astronomical Calculations 21](#_Toc332184439)

[Appendix D: Source Code 22](#_Toc332184440)

# Introduction

[*WSJT-X*](http://physics.princeton.edu/pulsar/K1JT/map65.html)is a computer program designed for amateur radio communication with very low power or when propagation conditions make signals weak. The first four letters stand for “Weak Signal communication by K1JT”, and the “*-X*” suffix indicates that *WSJT-X* is an experimental branch of the earlier program *WSJT*.

Version 1.0 of *WSJT-X* implements the new mode JT9, designed especially for use on the HF, MF, and LF bands. JT9 shares many characteristics with the modes JT65 and JT4 made popular in *WSJT*. All three are designed for making minimal QSOs under extreme weak-signal conditions. They use nearly identical message structure and source encoding. JT65 is used for EME on the VHF/UHF bands, and for worldwide QRP communication at HF. JT4 is used mainly on the microwave bands. In contrast, JT9 is optimized for HF and lower frequencies, and especially the 1.8 MHz, 472 kHz, and 137 kHz bands. JT9 is about 2 dB more sensitive than JT65A while using less than 10% of the bandwidth.

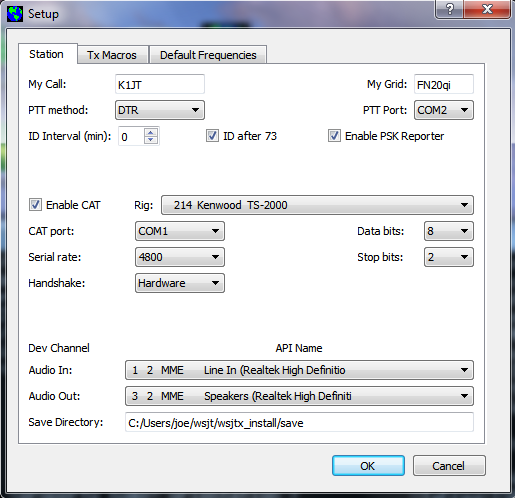
JT9 offers five choices for the duration of timed T/R sequences: submodes JT9-1, JT9-2, JT9-5, JT9-10, and JT9-30 use 1, 2, 5, 10, and 30 minutes, respectively. For obvious reasons JT9-1 is the preferred submode under most circumstances; the sub-modes with longer transmissions trade reduced throughput for smaller bandwidth and increased sensitivity. The slowest sub-mode, JT9-30, has total bandwidth 0.4 Hz and operates at signal-to-noise ratios as low as –40 dB measured in the standard 2.5 kHz reference bandwidth. JT9-1 is the recommended submode unless you really need the additional sensitivity of a slower mode.

# System Requirements

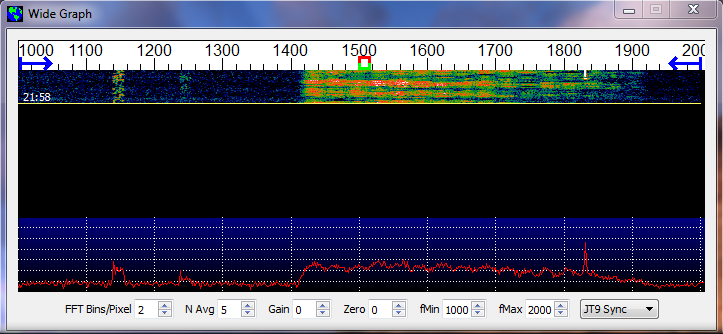
* SSB transceiver and antenna
* Computer running the Microsoft Windows/XP or later operating system. (*WSJT-X* also runs under Linux, but we do not yet provide a click-to-install package.)
* Pentium or faster CPU and 512 MB of available RAM
* Monitor with at least 800 x 600 resolution (more is better)
* Sound card supported by your operating system
* Computer-to-radio interface using a serial port to key your PTT line or for CAT control.
* Audio connections between transceiver and sound card
* A means for synchronizing your computer clock to UTC. I recommend is *Meinberg NTP*, see <http://www.satsignal.eu/ntp/setup.html> for installation instructions.

# Quick-Start Insllation and Setup

1. WSJT-X can be downloaded from the WSJT Home Page at <http://www.physics.princeton.edu/pulsar/K1JT/>. Click on the WSJT link at the left margin and then on the appropriate download link for WSJT-X.
2. Under Windows, execute the downloaded file and follow the installation instructions. Install WSJT-X into its own directory (the suggested default is C:\WSJTX) rather than the conventional C:\Program Files\WSJTX.
3. Start WSJT-X and Select **Options** from its **Setup** menu. Enter your callsign and Maidenhead locator, and set the remaining parameters as required for your station. Click **OK** to dismiss the **Options** window, and if necessary click the **Stop** button on the main window to halt any data acquisition.

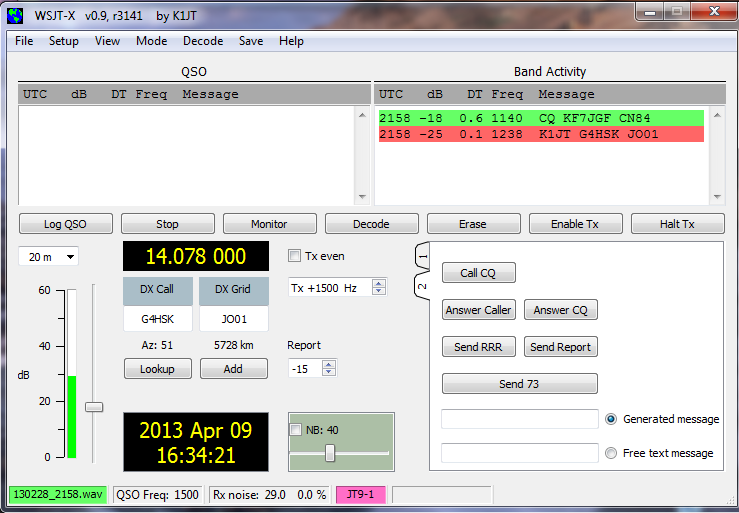


1. Select submode **JT9-1** from the **Mode** menu and **Normal** from the **Decode** menu. On the Wide Graph window select **JT9 Sync** (rather than **Current** or **Cumulative**) for data display. Select **Tab 2** (just below the **Erase** button) to choose the alternative set of controls for generating and selecting messages to be transmitted. Then select **File | Open**, navigate to directory …\Save\Samples under your WSJT-X installation directory, and open the example file 130228\_2158.wav. You should see something like the screen shots below and on the next page.

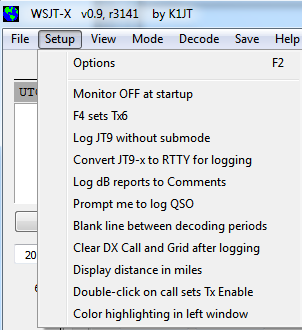


Notice on the waterfall frequency scale that the narrow-band Rx frequency is marked in green and the Tx frequency in red. The wideband decoding range is marked by blue arrows. The normal wideband range is 1000 – 2000 Hz, but you can move the limits using the **f Min** and **f Max** spinner controls.

Note that two JT9 signals have been decoded: KF7JGF is calling CQ, and G4HSK is responding to a CQ from K1JT. These signals are visible on the waterfall near audio frequencies 1140 and 1238 Hz, respectively. At SNR = -25 dB, the signal from G4HSK is close to the minimum for reliable decoding. The strong, wide signal between about 1400 and 1800 Hz is some other data mode; the JT9 decoder will spend some time trying to make sense of it, and then will eventually ignore it. The KF7JGF signal is highlighted in green because he is calling CQ. The signal from G4HSK is highlighted in red because it is directed to “MyCall”, in this case K1JT.



1. To gain some feeling for the controls you will use frequently when making QSOs, try clicking with the mouse on the decoded text lines and on the waterfall spectrum display. You should be able to confirm the following behavior:
   1. Double-click on decode line highlighted in green — copy callsign and locator of station calling CQ to “DX Call” and “DX grid”; generate suitable messages for minimal QSO with this station; moce Rx (green) and Tx (red) frequency markers on waterfall scale to match frequency of station calling CQ.
   2. Double-click on decode line highlighted in red — similar to (a) except Tx frequency is not moved.
   3. Click on waterfall — move Rx frequency (green marker) to selected frequency.
   4. CTRL-click on waterfall — move Rx and Tx frequencies.
   5. Double-click on waterfall — move Rx frequency and decode there. Notice that the decoded text now appears in the “QSO” window.
   6. CTRL-double-click — move Rx and Tx frequencies and decode there.
2. Examine the user options presented on the **Setup** menu. You may want to select some of these for your normal operation.



1. Click the **Monitor** button to return to normal receive operation and set the background noise level to around 30 dB on the thermometer scale at lower left of the main screen. With the slider at mid-scale, the dB scale is calibrated relative to the least significant bit of a 16-bit A/D converter in the soundcard. This setting is not critical.
2. You should now be ready to make QSOs with the JT9 modes in WSJT-X.

# Decoding Tutorial

The

# Additional Features and Options

The

# Appendix A: Installed and Generated Files

After installing *MAP65* as described in steps 1–3 on page 5, the following files will be present in the installation directory:

afmhot.dat Data for AFMHot palette

blue.dat Data for Blue palette

CALL3.TXT Callsign database

fftwf-wisdom.exe Program for FFT optimizations

fftwf\_wisdom.dat Results of running fftwf-wisdom.exe

kvasd.exe Koetter-Vardy decoder

m65.exe Slave program, controls all decoding steps

map65.exe Master MAP65 program

qthid.exe Slave program for configuring FUNcube

save Directory for saved files of wideband data

unins000.dat Data for uninstall utility

unins000.exe Default uninstall program

wisdom1.bat Batch file to run short FFT optimizations

wisdom2.bat Batch file to run long FFT optimizations

wsjt.ico WSJT icon

In addition, the following \*.dll support files will have been installed in your system directory, typically C:\Windows\System32 on a Windows system:

libfftw3f-3.dll

libgcc\_s\_dw2-1.dll

libstdc++-6.dll

libusb0.dll

mingwm10.dll

palir-02.dll

QtCore4.dll

QtGui4.dll

QtNetwork4.dll

QtSvg4.dll

qwt.dll

You might be curious about additional files that appear in the *MAP65* installation directory after using the program for a while. These include:

kvasd.dat Data for the Koetter-Vardy decoder

livecq.txt Information for web-based “LiveCQ” display

map65.ini Saved configuration parameters

map65.log Log file for decoder diagnostics, etc.

map65\_rx.log Log of all decoded messages

map65\_tx.log Log of all transmitted messages

prefixes.txt List of available add-on DXCC prefixes

timer.out Profile showing times in decoder routines

tmp26.txt Intermediate file used by decoder

# Appendix B: The JT65 Protocol

# Appendix D: Source Code

*MAP65* is an open-source program released under the [GNU General Public License](http://www.gnu.org/copyleft/gpl.html). Source code is available from the public repository at <http://developer.berlios.de/projects/wsjt/>. To compile the program you will need to install open source packages for Subversion, QtSDK, qwt, g++, g95 or gfortran, portaudio, and fftw3. For compiling in Windows I recommend installing the MinGW package.

The full source code for MAP65 may be downloaded by using the command

svn co svn://svn.berlios.de/wsjt/branches/map65 map65