**WSJT-X User’s Guide**

**Version 1.1**

**Joe Taylor, K1JT**

**Copyright © 2013**

**July 3, 2013**

Contents

[Introduction 3](#_Toc360603417)

[System Requirements 4](#_Toc360603418)

[Installation and Setup 4](#_Toc360603419)

[Basic Operating Tutorial 6](#_Toc360603420)

[Making QSOs 13](#_Toc360603421)

[On-Screen Controls 14](#_Toc360603422)

[Status Bar 19](#_Toc360603423)

[Menus 20](#_Toc360603424)

[Keyboard Shortcuts 22](#_Toc360603425)

[Under the Hood 23](#_Toc360603426)

[Acknowledgments 25](#_Toc360603427)

[Appendix A: Installed and Generated Files 25](#_Toc360603428)

[Appendix B: The JT9 Protocol 27](#_Toc360603429)

[Appendix C: Source Code 28](#_Toc360603430)

# Introduction

[*WSJT-X*](http://physics.princeton.edu/pulsar/K1JT/wsjt.html)is a computer program designed to facilitate basic amateur radio communication using very weak signals. The first four letters in the program name stand for “Weak Signal communication by K1JT”, and the “*-*X” suffix indicates that *WSJT-X* started as an extended (and experimental) branch of program *WSJT*.

Version 1.1 of *WSJT-X* offers two protocols or “modes,” JT9 and JT65. Both are designed for making reliable, confirmed QSOs under extreme weak-signal conditions. They use nearly identical message structure and source encoding. JT65 was designed for EME (“moonbounce”) on the VHF/UHF bands and has also proved very effective for worldwide QRP communication at HF. JT9 is optimized for the LF, MF, and HF bands. It is about 2 dB more sensitive than JT65 while using less than 10% of the bandwidth. Both modes use one-minute timed sequences of alternating transmission and reception[[1]](#footnote-1), so a minimal QSO takes 4 to 6 minutes — two or three transmissions by each station, one sending in odd minutes and the other even. World-wide QSOs are possible with power levels of a few watts and compromise antennas.

Starting with version 1.1, *WSJT-X* offers up to five times the displayed bandwidth and provides dual-mode reception. If your receiver has an SSB filter with at least 4 kHz bandwidth, you can set your HF dial frequency to one of the standard JT65 frequencies — for example, 14.076 MHz for 20 meters — and have the full range of JT65 and JT9 frequencies displayed simultaneously on the waterfall. You can make QSOs in both modes using nothing more than mouse clicks.

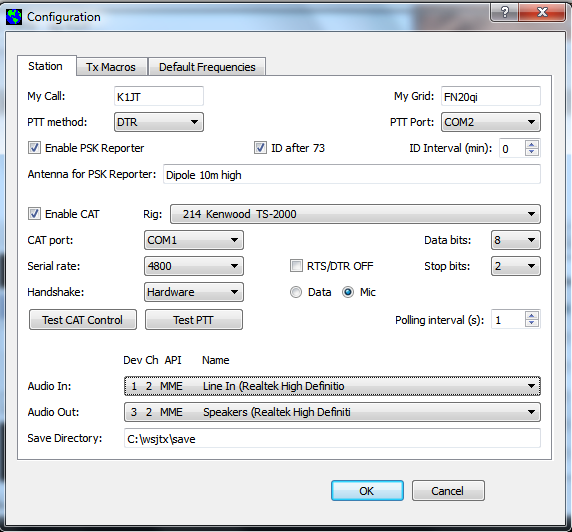
Plans for future program development call for *WSJT-X* and *WSJT* to merge together: *WSJT-X* will gradually acquire the additional modes JT4, FSK441, and ISCAT that are now supported in *WSJT*. The entire WSJT-related effort is an open-source project. If you have programming skills or would like to contribute to the project in other ways, please make your interests known to K1JT and the rest of the development team. The project’s source-code repository can be found at <http://developer.berlios.de/projects/wsjt/>, and communication among the developers takes place on the email reflector [wsjt-devel@lists.berlios.de](mailto:wsjt-devel@lists.berlios.de). You can subscribe to this list at <https://lists.berlios.de/mailman/listinfo/wsjt-devel>.

# System Requirements

* SSB transceiver and antenna
* Computer running Windows XP or later, Linux, or OS X (and probably FreeBSD and other unix-like operating systems. See Appendix C.)
* 1.5 GHz or faster CPU and 100 MB of available memory
* Monitor with at least 1024 x 780 resolution (more is better)
* Audio input and output devices supported by your operating system
* Computer-to-radio interface using a serial port to key your PTT line, or CAT control, or VOX for T/R switching.
* Audio or equivalent USB connections between transceiver and computer
* A means for synchronizing your computer clock to UTC. The built-in Windows facility is usually not adequate. I recommend *Meinberg NTP:* see <http://www.satsignal.eu/ntp/setup.html> for installation instructions.

# Installation and Setup

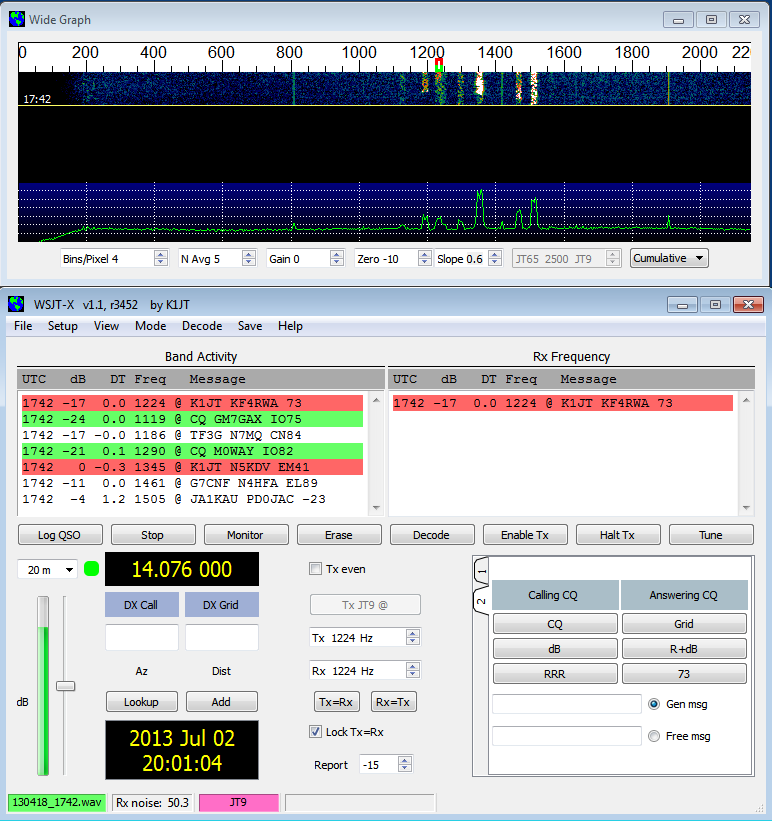
1. *WSJT-X* can be downloaded from the WSJT Home Page, <http://www.physics.princeton.edu/pulsar/K1JT/>. Click on the *WSJT-X* link at the left margin and then on the appropriate download link.
2. Under Windows, execute the downloaded file and follow its installation instructions. Install *WSJT-X* into its own directory (the suggested default is C:\WSJTX) rather than the conventional C:\Program Files\WSJTX. By default all files relating to *WSJT-X* will be stored in this directory and its subdirectories. You can uninstall *WSJT-X* simply by removing the installation directory and its contents. See Appendix C for other operating systems.
3. Start *WSJT-X* and Select **Configuration** from its **Setup** menu. Enter your callsign and Maidenhead grid locator as shown in the screen shot on the next page, and set the remaining parameters as required for your station. A simple setup might use **PTT method** = VOX and **PTT Port** = None. Stations already configured for other digital modes will typically use **PTT method** = DTR or RTS and a serial communication port such as COM1 for **PTT Port**.
4. Many users will have other software controlling their radios. *WSJT-X* does not implement full transceiver control, but it does provide a way to ensure that *WSJT-X* can know the radio’s dial frequency. If you want this capability check the box **Enable CAT**, select your radio type from a drop-down list, and select a port (not the same port selected for PTT control) and whatever serial communication parameters may be required by your radio. If using CAT control, most radios will allow you to set **PTT method** = CAT. Your radio may support two types of PTT assertion via CAT control: one takes audio input from the **Mic** connector, the other from a rear-panel **Data** connector. The simplest CAT configuration sets **Polling interval** to 0 (no polling the radio for dial frequency). *WSJT-X* is then able to set the radio’s frequency, but the program will be unaware of subsequent changes made using the radio’s panel controls. With most radios you can set **Polling interval** to a reasonable small number (say 1 – 3 s) and the program will then follow any frequency changes made at the radio. Note that you may not be able to simultaneously control your radio from *WSJT-X* and from another software program. Some experimentation may be required, and you may need to refer to the documentation for your radio. It is best to have the radio and any interface equipment turned on and connected before starting *WSJT-X,* and to exit the program before turning your equipment off.



1. If you use *Commander* or *Ham Radio Deluxe* to control your transceiver, you can configure *WSJT-X* to communicate with the radio through that program. Select “9998 Commander” or “9999 Ham Radio deluxe” from the drop-down list of rigs, and *Commander* or *HRD* will take care of the rest.
2. Try clicking the **Test CAT Control** and **Test PTT** buttons to see that you have established the desired control of station functions. Then click **OK** to dismiss the **Configuration** window.
3. *WSJT-X* expects your sound card to do its raw sampling at 48000 Hz. To ensure that this will be so when running under Windows, open the Sound control panel and select in turn the “Recording” and “Playback” devices you will use for audio input and output. Click on Properties, then Advanced, and select “16 bit, 48000 Hz (DVD Quality).” If you are using a sound card that is also the default device for Windows sounds, be sure to turn off all such sounds so they are not transmitted over the air.
4. To set the proper level of audio drive from *WSJT-X* to your radio, click the **Tune** button on the main screen. *WSJT-X* should set the radio into transmit mode and generate a steady audio tone at the same amplitude that will be used for a generated JT9 signal. Listen to the generated audio tone using your radio’s “Monitor” facility, or by another method. The tone should be perfectly smooth, with no clicks or glitches. Open the computer’s audio mixer controls for output (“playback”) devices and adjust the volume slider downward until the RF output from your transmitter falls by around 10%. This will be a good level for audio drive. Toggle the **Tune** button once more to stop your test transmission.

# Basic Operating Tutorial

1. Click the **Stop** button on the main window to halt any data acquisition. Select **JT9** from the **Mode** menu and **Deepest** from the **Decode** menu. Set the audio Tx and Rx frequencies to **Tx 1224 Hz** and **Rx 1224 Hz**. On the Wide Graph window set **Bins/Pixel** = 4, **N Avg** = 5, **Gain** = 0, **Zero** = –10, **Slope** = 0.6, and **Cumulative** (rather than **Current**) for data display. Select **Tab 2** (just below the **Erase** button on the main window) 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 130418\_1742.wav. You should see something like the screen shot on the next page.
2. Notice the green and red markers on the waterfall frequency scale. Decoding takes place at the end of a receive sequence and is organized in two stages. The first decodes take place at the selected Rx frequency, indicated by the green marker. Results appear in the both the left (“Band Activity”) and right (“Rx Frequency”) text windows on the main screen. The decoder then finds and decodes all signals in the selected mode and the displayed frequency range. The red marker indicates your Tx frequency.



1. Note that at least eight JT9 signals are present in the example file; all but one of them are decodable. When this file was recorded KF4RWA was finishing a QSO with K1JT. Since the green marker was placed at his audio frequency, 1224 Hz, his message “K1JT KF4RWA 73” appears in both decoded text windows. The “Band Activity” window shows this message as well as all the other decodes at nearby frequencies. The CQ lines are highlighted in green, and lines containing “My Call”, in this case K1JT, are highlighted in red. (For this step and the next one, you may want to pretend you are K1JT by entering that call temporarily as “My Call” on the **Configuration** screen. Your results should then be identical to those shown here.)
2. To gain some feeling for the controls you will use when making QSOs, try clicking with the mouse on the decoded text lines and on the waterfall spectral display. You should be able to confirm the following behavior:
   1. Double-click on either of the decoded lines highlighted in green. This action copies callsign and locator of a station calling CQ to the “DX Call” and “DX grid” entry fields. It also generates suitable messages for a minimal QSO and checks or clears the **Tx even** box so that you will transmit in the proper (odd or even) minutes. Rx and Tx frequency markers will be moved to the CQing station’s frequency, and the **Gen Msg** (“generated message”) radio button at bottom right of the main window will be selected. If you had checked “Double-click on call sets Tx Enable” on the **Setup** menu, **Enable Tx** would also be activated, and you would start to transmit automatically, at the appropriate time.
   2. Double-click on the decoded line with the message “K1JT N5KDV EM41”, highlighted in red. Results will be similar to (a), except the Tx frequency (red marker) is not moved. Such messages are usually in response to your own CQ, or from a tail-ender, and you probably want your Tx frequency to stay where it was. By holding down the Ctrl key when double-clicking on the decoded line (or checking **Lock Tx=Rx**) you can cause both Tx and Rx frequencies to be moved.
   3. Now double-click on the message from KF4RWA in either window. He is sending “73” to K1JT, signifying that the QSO is over. Most likely you want to send 73 to him, so the message “KF4RWA K1JT 73” is automatically generated and selected for your next transmission. (Alternatively, you might choose to send a free text message or to call CQ again.)
   4. Clicking on the waterfall moves the Rx frequency (green marker) to the selected frequency. Ctrl-click on waterfall moves both Rx and Tx frequencies.
   5. Double-click on the waterfall moves the Rx frequency and causes a narrow-band decode there at the new QSO frequency. Decoded text appears in the right window only. Ctrl-double-click moves both Rx and Tx frequencies and decodes at the new frequency.
   6. Clicking **Erase** clears the right window. Double-click on **Erase** to clear both text windows.
3. Now set **Bins/Pixel** = 7, **Zero** = -3, and **Slope** = 1.1. If necessary, adjust the width of the Wide Graph Window so that the upper frequency limit is 4000 Hz. Select **JT9+JT65** on the **Mode** menu, toggle the **Tx mode** button to read **Tx JT65**, and set the Tx and Rx frequencies to 1718 Hz. Double-click on **Erase** to clear both text windows. Then open the sample file 130610\_2343.wav. The waterfall and main window should look like the screen shot on the next page. This sample file contains 17 decodable signals — nine in JT65 mode (flagged with the character **#** in the decoded text windows), and eight in JT9 mode (flagged with **@**). Since the Tx mode was set to **Tx JT65**, signals in that mode were decoded first. If you had selected **Tx JT9**, JT9 signals would be decoded first.
4. Notice the blue marker on the waterfall scale, by default set at 2500 Hz. Its position is set by the spinner control **JT65 nnnn JT9**, where nnnn is a frequency in Hz. In **JT9+JT65** mode the program will decode JT65 signals below this frequency and JT9 signals above it.
5. Confirm that mouse-click behavior is similar to that described in the single-mode instructions (item 4, above). The program automatically determines the mode of each JT9 or JT65 signal. For example, double-click on the waterfall near 815 Hz: a signal originating from W7VP will be decoded, and the line

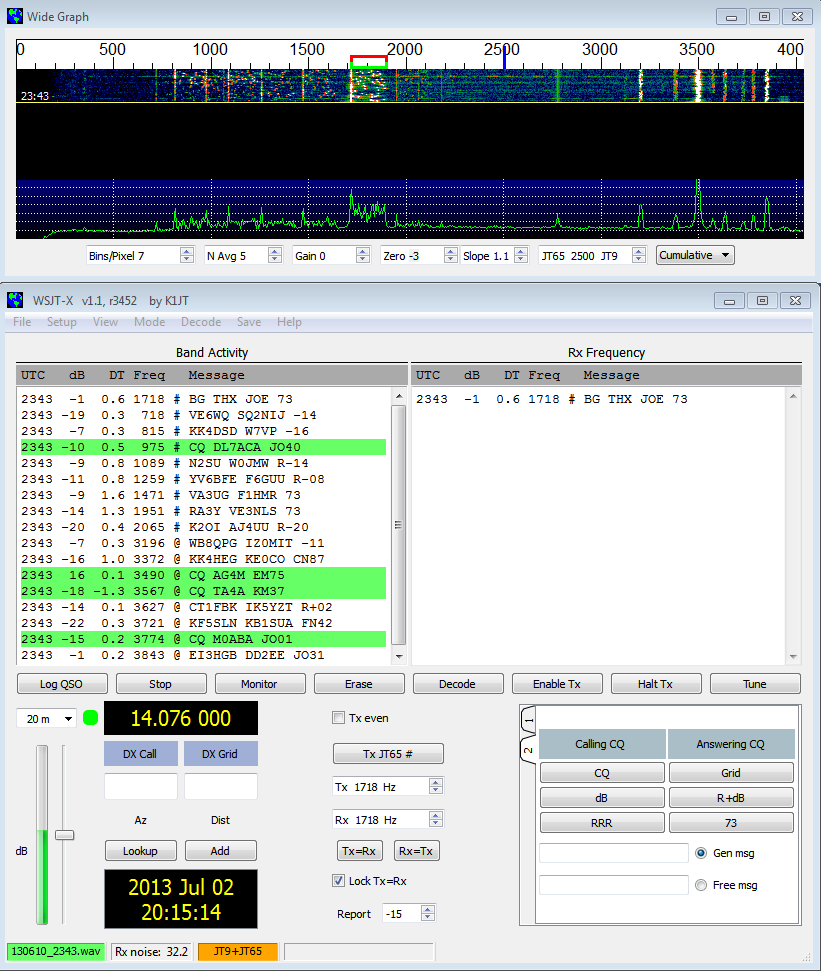
2343 -7 0.3 815 # KK4DSD W7VP -16

should appear in the Rx Frequency text box. Double-click on the waterfall at 3196 Hz and the program will decode a JT9 message from IZ0MIT:

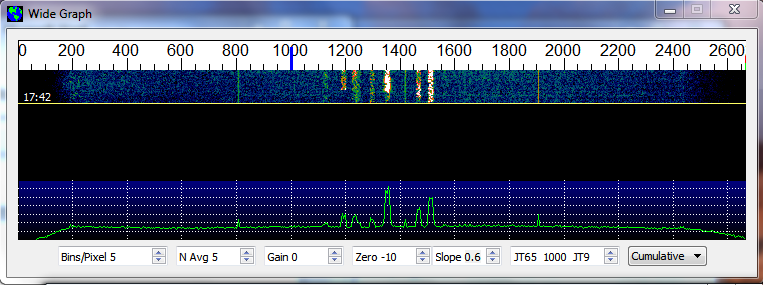
2343 -7 0.3 3196 @ WB8QPG IZ0MIT -11

Notice that when a signal is decoded in this way the Tx mode automatically switches to that of the decoded signal. The Rx and Tx frequency markers on the waterfall scale resize themselves accordingly.

1. Scroll back in the Band Activity window (if necessary) and double-click on the message CQ DL7ACA JO40. The program will set Tx mode to JT65 and Tx and Rx frequencies to that of DL7ACA, 975 Hz. If you had checked **Double-click on call sets Tx Enable** on the **Setup** menu, the program would prepare itself to start a QSO with DL7ACA.
2. Double-click on the decoded JT65 message CQ TA4A KM37. The program will set Tx mode to JT9 and both frequencies to 3567 Hz. You’re now configured properly for a JT9 QSO with TA4A.



1. Again open file 130418\_1742.wav in the Samples directory. You can immediately see that this data was recorded with a much narrower Rx bandwidth, roughly 200 to 2600 Hz. If you have no Rx filter wider than about 2.7 kHz, you will necessarily be using data similar to this sample. For best viewing of such data adjust **Bins/Pixel** and the width of the Wide Graph so that only the active part of the spectrum shows, say 0 to 2800 Hz. (Re-open the file after any change of Bins/Pixel or Wide Graph width, to refresh the waterfall.) The signals in this file are all JT9 signals. To decode them in **JT9+JT65** mode you’ll need to move the **JT65 nnnn JT9** delimiter down to 1000 Hz or less.
2. Now is a good time to experiment with the **Zero** and **Slope** parameters. **Zero** sets the baseline level for waterfall colors, while **Slope** provides a way to correct for non-flat spectral response in your radio. For the receiver setup of this file good values are around –10 and +0.6, respectively. If the **Cumulative** spectral baseline (green curve) slopes downward toward higher frequencies, increase the **Slope** setting. Re-open the wave file after each change, to see the new results. Here’s the resulting waterfall display with parameters set properly for my TS-2000 with 200-2600 Hz filter settings:



1. Don’t forget to re-enter your own callsign as “My Call”. Click the **Monitor** button to return to normal receive operation, and be sure that your transceiver is set to USB (or USB Data) mode. Using the receiver gain control(s) and/or the Windows mixer controls, set the background noise level to around 30 dB or mid-scale (lower left of main window). If necessary you can also use the slider next to the scale — but note that the overall dynamic range will be best with this slider close to its mid-point.
2. **Frequency Setting:** To take advantage of the wideband, dual-mode capability of *WSJT-X* v1.1, you’ll need a receiver bandwidth of at least 4 kHz (USB mode, of course). For example, on a Kenwood TS-2000 I set **Low Cut** to 200 and **High Cut** to 5000 Hz. Note that most SSB transceivers have a fixed filter in the Tx side that will not pass audio frequencies higher than about 2700 Hz. *WSJT-X* v1.1 takes care of this by using **Split** mode, receiving with VFO A and transmitting with VFO B. The Tx dial frequency (VFO B) is offset in 1000 Hz steps, and the generated audio frequency is adjusted so that it always falls in the range 1000 – 2000 Hz. With CAT enabled and your transceiver set to **Split** mode, frequency control will be handled automatically.
3. If you have only a standard SSB filter on the Rx side you won’t be able to use more than about 2.7 kHz bandwidth. You can still have all of the JT9 sub-band and part of the JT65 sub-band available, however. On 20m, say, set dial frequency (VFO A) to 14.0774 and the **JT9 | JT65** dividing line at 1600 Hz. JT9 signals in their conventional sub-band will then appear at 1600 – 2600 Hz, while JT65 signals will be below 1000 Hz. Of course, you might prefer to concentrate on one mode at a time, setting your dial frequency to (say) 14.076 for JT65 and 14.078 for JT9. In general, present conventions have the nominal JT9 dial frequency 2 kHz higher than the JT65 dial frequency.
4. You should now be ready to make QSOs using either JT9 or JT65.

# Making QSOs

By longstanding tradition, a minimal valid QSO requires the exchange of callsigns, a signal report or some other information, and acknowledgments. *WSJT-X* is designed to facilitate making such minimal QSOs using short, formatted messages. The process works best if you use these formats and follow standard operating practices. The recommended basic QSO goes something like this:

1. CQ K1ABC FN42

2. K1ABC G0XYZ IO91

3. G0XYZ K1ABC –19

4. K1ABC G0XYZ R–22

5. G0XYZ K1ABC RRR

6. K1ABC G0XYZ 73

These standard messages consist of two callsigns (or CQ, QRZ, or DE and one callsign) followed by the transmitting station’s grid locator, a signal report, or the acknowledgement “RRR” or “73”. These messages are compressed and encoded in a highly efficient and reliable way. Signal reports are given as signal-to-noise ratio in dB, using the standard reference noise bandwidth 2500 Hz. For JT65 they must lie in the range –30 to –1 dB; JT9 supports the extended range –50 to +49 dB.

In example message #3 K1ABC is telling G0XYZ that his signal is 19 dB below the noise power in bandwidth 2500 Hz. In message #4 G0XYZ acknowledges receipt of that report and responds with a –22 dB signal report. For operators with good hearing, signals start to become audible around –15 dB on this scale. Signals are visible on the waterfall down to about –26 dB. The JT65 decoder begins to fail at around –24 dB, JT9 around –26 dB.

Free-format messages such as “TNX JOE 73 GL” or “5W VERT 73 GL” can be sent up to a maximum of 13 characters. Users often add some friendly chit-chat as a final transmission, in place of the formatted “73” message. It should be obvious, however, that JT9 and JT65 are not suitable for extensive conversations or rag-chewing.

Compound callsigns such as PJ4/K1ABC or G0XYZ/P are handled in a slightly different way. The following formats are all valid:  
  
 CQ pfx/callsign grid   
 QRZ pfx/callsign grid   
 DE pfx/callsign grid   
 CQ callsign/sfx grid   
 QRZ callsign/sfx grid   
 DE callsign/sfx grid   
  
where “pfx” is a 1-4 character prefix, “callsign” is a standard callsign, “sfx” is a 1-3 character suffix, and “grid” is a 4-character Maidenhead locator, a signal report of the form “±nn” or “R±nn”, or the acknowledgment or signoff messages “RRR” or “73”. *WSJT-X* generates messages in these forms automatically, as required. A QSO between two stations using compound callsigns might look like this:

1. CQ KP4/K1ABC FK68

2. DE G0XYZ/P IO91

3. G0XYZ K1ABC –19

4. K1ABC G0XYZ R–22

5. G0XYZ K1ABC RRR

6. DE G0XYZ/P 73

Before attempting your first QSO with JT9 or JT65, be sure to go through the tutorial above as well as the following checklist:

* Your callsign and grid locator set to correct values
* PTT and CAT control (if used) properly configured and tested
* Computer clock properly synchronized with UTC to within ±1 s.
* Radio set to USB (upper sideband) mode
* Remember that JT9 and JT65 generally do not require high power. In most propagation conditions, QRP is the rule!

# On-Screen Controls

The following controls appear at the bottom of the Wide Graph window:



**FFT Bins/Pixel** controls the displayed frequency resolution. Set to 1 for the highest possible resolution, or to higher values to compress the spectral display. Normal operation with a convenient window size works well at 2 to 8 bins per pixel.

**N Avg** is the number of successive FFTs to be averaged before updating the spectral display. Values around 5 are suitable for normal JT9 and JT65 operation.

**Gain** and **Zero** control the scaling and reference level for waterfall colors. Values around 0 for both parameters are usually about right, depending on the input signal level and your own preferences.

**Slope** lets you make first-order corrections for a receiver spectral response that is not flat.

**JT65 nnnn JT9** sets the dividing point for wideband decoding of JT65 and JT9 signals in **JT9+JT65** mode. The decoder looks for JT65 signals below nnnn Hz and JT9 signals above.

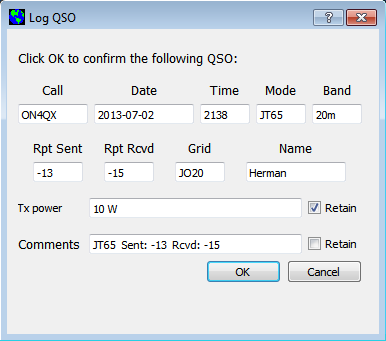
**Current / Cumulative / JT9 Sync** controls the graphical display in the bottom one-third of the Wide Graph window. **Current** is the average spectrum over the most recent **N Avg** FFT calculations. **Cumulative** is the average spectrum since the start of the current Rx sequence.

With the exception of **JT65 nnnn JT9**, controls on the Wide Graph window affect only the graphical displays — they have no effect on the decoding process.

The following buttons appear just under the decoded text windows on the main window:



**Log QSO** pops up a confirmation screen pre-filled with known information about a QSO you have nearly completed. You can edit or add to this information before clicking OK to log the QSO. If you select “Prompt me to log QSO” on the **Setup** menu, the program will pop up the confirmation screen automatically when you send a “73” or free-text message.



**Stop** will stop normal data acquisition in case you want to open and explore previously recorded audio files.

**Monitor** restarts normal program operation in receive mode. This button is highlighted in green when the program is receiving.

**Decode** tells the program to repeat the decoding procedure at the Rx frequency (green marker on waterfall), using the most recently completed sequence of Rx data.

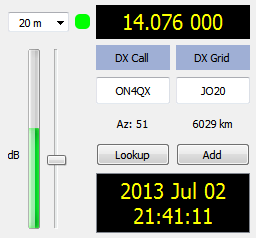
**Erase** clears the right (Rx frequency) window. Double-clicking clears both text windows.

**Tune** may be used to switch into Tx mode and generate an unmodulated carrier at the specified Tx frequency (red marker on waterfall). This process may be useful for adjusting an antenna tuner, for example. Toggle the button a second time to terminate the **Tune** process.

**Enable Tx** puts the program into automatic Rx/Tx sequencing mode and highlights the button in red. A transmission will start at the beginning of the selected (odd or even) sequence.

**Halt Tx** will terminate a transmission in progress and disable automatic Rx/Tx sequencing.

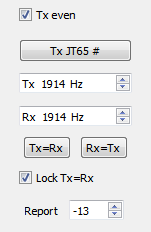
At lower left of the main window are controls and displays related to date and time, frequency, Rx audio level, and the station being worked.



The drop-down **Band** selector at upper left lets you select the operating band and sets dial frequency to a default value taken from the **Default Frequencies** tab on the **Setup | Configuration** screen. If you are using CAT control, a small colored square appears in green if the CAT control is two-way between *WSJT-X* and your radio, or orange if the control is only from program to radio. (You can request a one-time interrogation of the radio’s dial frequency by clicking on the orange square.) The square becomes red if you have requested CAT control but communication with the radio is lost.

If the **Dx Grid** is known, the great-circle azimuth and distance are given. The program can keep a database of callsigns and locators for future reference. Click **Add** to insert the present call and locator in the database; click **Lookup** to retrieve the locator for a previously stored callsign.

At center and right of the main window are a number of controls you will use when making QSOs. The following are near screen center:



Select **Tx even** to transmit in even-numbered UTC minutes. Uncheck this box to transmit in the odd intervals. This selection is made automatically when you double-click on a decoded text line as described in the Basic Operating Tutorial, pages 6–9 of this Guide. Your audio Tx and Rx frequencies are displayed and can be adjusted withspinner controls. These settings are normally handled automatically by the double-click procedure. The on-the-air frequency of your lowest JT9 or JT65 tone will be the sum of dial and audio frequencies. You can force Tx frequency to the current Rx frequency by clicking the **Tx=Rx** button, and vice-versa for **Rx=Tx**; you can check the **Lock Tx=Rx** box to make the frequencies always track one another. The **Report** control lets you change a signal report that may have been inserted automatically. Most reports will fall in the range –26 to +10 dB. When signals are close to or stronger than 0 dB, you and your QSO partner should probably reduce power. These are supposed to be *weak signal* modes!

Two configurations of controls are provided for generating and selecting Tx messages:



Traditional controls (carried over from program *WSJT*) provide six fields for message entry. Pre-formatted messages for the standard minimal QSO (see page 12) are generated when you click **Generate Std Msgs**, or when you double-click on an appropriate line of decoded text. Select the next message to be transmitted (at the start of your next Tx sequence) by clicking on the circle under **Next**. To change to a specified Tx message immediately, perhaps after a transmission has already started, click on a rectangular button (e.g., **Tx 3**) under the **Now** label. Changing Tx messages after a transmission has started reduces the probability of a correct decode by your QSO partner, but if you do it in the first 10 s or so of a Tx period, it will most likely succeed.

Right-clicking on the entry field for message #5 pops up a list of free text messages entered on the **Setup | Configuration | Tx Macros** dialog window, for example



You can select any of these pre-stored messages with the left mouse button.

The second configuration of message-selecting controls looks like this:



With this setup you will normally follow a top-to-bottom sequence of transmissions from the left column (if you are calling CQ) or the right column (if you are answering a CQ). Clicking a button puts the appropriate message in the **Gen Msg** box. If you are already transmitting, it changes the Tx message immediately. The actual message being transmitted always appears highlighted in yellow in the first box on the status bar, at the bottom left of the main screen. You can put anything you like (up to 13 characters) in the **Free Msg** box. Right-clicking on this entry field pops up your previously defined list of Tx Macros.

# Status Bar

A Status Bar at the bottom edge of the main window provides information about operating conditions.



Reading from left to right, these labels provide information about the current operating state (Receiving, Transmitting, Tune, or an opened file name), received noise level in dB, operating mode, and content of the most recent transmitted message.

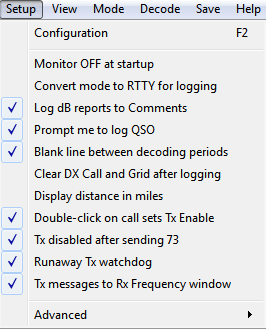
# Menus

Program menus offer many options for configuration and operation. Explore them and test the resulting program actions.

**File menu**



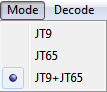
**Setup menu**



**View menu**



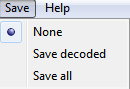
**Mode menu**



**Decode menu**



**Save menu**



**Help menu**



# Keyboard Shortcuts

**F1** Display online User's Guide in browser

**Ctrl+F1** About WSJT-X

**F2** Open the Setup | Configuration window

**F3** Display keyboard shortcuts

**F4** Clear Dx Call and Dx Grid entries

**Alt+F4** Exit program

**F5** Display special mouse commands

**F6** Open next file in directory

**Shift+F6** Decode all remaining files in directory

**F11** Move Rx frequency down 1 Hz

**Ctrl+F11** Move Rx and Tx frequencies down 1 Hz

**F12** Move Rx frequency up 1 Hz

**Ctrl+F12** Move Rx and Tx frequencies up 1 Hz

**Alt+1-6** Set next transmission to this number on Tab 1

**Alt+D** Decode again at Rx frequency

**Shift+D** Full decode (both windows)

**Alt+E** Erase

**Ctrl+F** Edit the free text message box

**Alt+G** Generate standard messages

**Alt+H** Halt Tx

**Ctrl+L** Lookup callsign in database, generate standard messages

**Alt M** Monitor

**Alt+N** Enable Tx

**Alt+Q** Log QSO

**Alt+S** Stop monitoring

**Alt+T** Tune

**Alt+V** Save the most recently completed \*.wav file

Special Mouse Commands

|  |  |
| --- | --- |
| **Mouse-click on** | **Action** |
| **Waterfall** | Set Rx frequency  Double-click to set Rx frequency and decode there  Ctrl-click to set Rx and Tx frequencies  Ctrl-double-click to set Rx and Tx frequencies and decode there |
| **Decoded text** | Double-click to copy second callsign to Dx Call and locator to Dx Grid; change Rx and Tx frequencies to decoded signal's frequency; generate standard messages. If first callsign is your own, Tx frequency is changed only if Ctrl is held down when double-clicking. |
| **Erase button** | Click to erase QSO window  Double-click to erase QSO and Band Activity windows |
| **Tx5 Entry** | Right-click to select a macro message |

# Background on the *WSJT-X* Modes

Since its origin in 2001, an important goal of the *WSJT* project has been education —bringing to Amateur Radio a better understanding of 21st century communication theory and its applications. I have acquired plenty of this education myself, while working on the project, and I think all of us contributing programming and other skills to the project feel that way. We hope that users of our software will, also. JT65 is a mature mode that has been described in detail before[[2]](#footnote-2). This brief section of the *User’s Guide* provides some technical details the JT9 protocol and how it is implemented in *WSJT-X*.

**Transmitting:** Immediately before the start of a transmission, *WSJT-X* encodes a message and computes the sequence of tones to be sent. A JT9 sequence consists of 85 tone intervals or “symbols,” each at one of 9 pre-defined frequencies. Tones are separated in frequency by 1.736 Hz — the reciprocal of the tone duration, which is 6912/12000 = 0.576 s. Tone differences are scarcely perceptible to the human ear, so a JT9 signal sounds almost like an unmodulated carrier.

Sixteen transmitted symbols — those at positions 1, 2, 5, 10, 16, 23, 33, 35, 51, 52, 55, 60, 66, 73, 83, and 85 in the sequence — are always transmitted at the lowest (“tone 0”) frequency. Their frequency and their pseudo-random spacings in time are used to establish time and frequency synchronization between transmitter and receiver. The remaining 69 intervals use tones 1 through 8, and they effectively send 207 information-carrying bits (three bits per tone) on their way into the ether. *WSJT-X* computes the transmitted audio waveform on-the-fly, calculating 16-bit integer samples at a 48000 Hz sample rate and maintaining phase continuity across symbol boundaries. The digital samples are converted to an analog waveform in the sound card or equivalent USB interface.

**Receiving:**  *WSJT-X* acquires 16-bit integer samples from the sound card at a 12000 Hz rate. Overlapping windowed spectra are computed for display and saved at intervals of half the symbol length. They are used to produce the **Current** and **Cumulative** curves that can be displayed below the waterfall in the Wide Graph window. As shown in the screenshot on page 7, a JT9 signal appears in the **Cumulative** spectrum as a nearly rectangular shape about 16 Hz wide. By convention, the nominal frequency of a JT9 signal is that of the sync tone at the left edge of the spectrum.

**Decoding:** At the end of a reception sequence, about 50 seconds into the UTC minute, the acquired samples are sent to the decoder. For operator convenience the decoder goes through its full procedure twice: first over a narrow range around the selected Rx frequency, and then in the full displayed frequency range (or in **JT9+JT65** mode, the displayed range above the blue **JT65 nnnn JT9** marker. Each pass can be described as a sequence of discrete steps. For those who may wish to study the program’s source code, perhaps with an eye toward making future improvements, the steps are presented here as blocks labeled with the names of functional procedures in the code.

**sync9:** Use sync symbols to find candidate JT9 signals

in the specified frequency range

Then, at the frequency of each plausible candidate:

**downsam9:** Mix, filter and downsample to 16 complex samples/symbol

**peakdt9:** Using sync symbols, time-align to start of JT9 symbol

sequence

**afc9:** Measure frequency offset and any possible drift

**twkfreq:** Remove frequency offset and drift

**symspec2:** Compute 8-bin spectra for 69 information-carrying

symbols, using the time- and frequency-aligned data;

transform to yield 206 single-bit soft symbols

**interleave9:** Remove single-bit symbol interleaving imposed at the

transmitter

**decode9:** Retrieve a 72-bit user message using the sequential

"Fano" algorithm for convolutional codes

**unpackmsg:** Unpack a human-readable message from the 72-bit

compressed format

It’s useful to know that when conditions necessary for decoding are marginal, the sequential Fano algorithm can have exponentially long times to completion. If the first step in the above sequence finds many seemingly worthy candidate signals, and if many of them turn out to be undecodable, the decoding loop can take a long time. For this reason the **decode9** step in the above list is programmed to time out and report failure if it is taking too long. The menu choice **Decode | Fast / Normal / Deepest** gives you three-step control of the timeout limit.

# Acknowledgments

Many users of *WSJT*, too numerous to mention here individually, have contributed suggestions and advice that have greatly aided the development of *WSJT-X* and its sister programs. Since 2005 the overall project (including *WSJT*, *MAP65*, *WSPR*, and *WSPR-X*) has been “open source”, all code being licensed under the [GNU Public License (GPL)](http://www.gnu.org/licenses/gpl.html). For *WSJT-X* in particular, I wish to acknowledge contributions from AE4JY, PY2SDR, VK4BDJ, AC6SL, G4KLA, and AF5X. Each has helped to bring the program’s design, code, and documentation to its present state.

# Appendix A: Installed and Generated Files

After installing *WSJT-X* as described in steps 1 and 2 on page 4, 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

hamlib-alinco.dll Hamlib libraries

hamlib-amsat.dll ...

hamlib-flexradio.dll

hamlib-icom.dll

hamlib-jrc.dll

hamlib-kachina.dll

hamlib-kenwood.dll

hamlib-kit.dll

hamlib-tapr.dll

hamlib-tentec.dll

hamlib-winradio.dll

hamlib-yaesu.dll

HRDInterface001.dll Ham Radio Deluxe interface library

jt9.exe Executable for JT9 decoder

jt9code.exe Test program to illustrate JT9 encoding

libfftw3f-3.dll Optimized FFT library

libgcc\_s\_dw2-1.dll gcc runtime

libhamlib-2.dll

libstdc++-6.dll standard C function library

libusb0.dll USB interface functions

mingwm10.dll MinGW library

mouse\_commands.txt Special mouse commands

palir-02.dll Linrad functions

PSKReporter.dll Library for PSK reporter

QtCore4.dll Qt libraries

QtGui4.dll ...

QtNetwork4.dll

QtSvg4.dll

qwt.dll Qwt library

save Directory for saved \*.wav files

shortcuts.txt Keyboard shortcuts

unins000.dat

unins000.exe Executable for uninstalling WSJT-X

wsjt.ico WSJT icon

wsjtx.exe Executable for WSJT-X

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

ALL.TXT Log of all received and transmitted messages

decoded.txt Decoded text from the most recent Rx interval

timer.out Diagnostic information for decoder optimization

wsjtx.ini Saved configuration parameters

wsjtx\_log.adi ADIF log

wsjtx\_status.txt Information sent to companion program JT-Alert

# Appendix B: The JT9 Protocol

JT9 is a mode designed for making QSOs at HF, MF, and LF. The mode uses essentially the same 72-bit structured messages as JT65. Error control coding (ECC) uses a strong convolutional code with constraint length K=32, rate r=1/2, and a zero tail, leading to an encoded message length of (72+31) × 2 = 206 information-carrying bits. Modulation is 9-FSK: 8 tones are used for data, one for synchronization. Sixteen symbol intervals are devoted to synchronization, so a transmission requires a total of 206 / 3 + 16 = 85 (rounded up) channel symbols. Symbol durations are approximately (TRperiod - 8) / 85, where TRperiod is the T/R sequence length in seconds. Exact symbol lengths are chosen so that nsps, the number of samples per symbol (at 12000 samples per second) is a number with no prime factor greater than 7. This choice makes for efficient FFTs. Tone spacing of the 9-FSK modulation is df = 1 / tsym = 12000 / nsps, equal to the keying rate. The total occupied bandwidth is 9 × df. The generated signal has continuous phase and constant amplitude, and there are no key clicks.

During a phase of experimentation, submodes of JT9 were defined with transmission lengths greater than one minute. Parameters of these submodes are summarized in the following table, along with approximate decoding thresholds measured by simulation on an additive white Gaussian noise (AWGN) channel. Numbers following “JT9-” in the submode names specify the T/R sequence length in minutes. (NB: When not otherwise specified, JT9 implies submode JT9-1. Submodes other than JT9-1 may not be available in a particular version of *WSJT-X*.)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Submode | nsps | Symbol Duration  (s) | Tone Spacing (Hz) | Signal Bandwidth (Hz) | S/N Threshold\* (dB) | QSO Time (minutes) |
| JT9-1 | 6912 | 0.58 | 1.736 | 15.6 | -27 | 6 |
| JT9-2 | 15360 | 1.28 | 0.781 | 7.0 | -30 | 12 |
| JT9-5 | 40960 | 3.41 | 0.293 | 2.6 | -34 | 30 |
| JT9-10 | 82944 | 6.91 | 0.145 | 1.3 | -37 | 60 |
| JT9-30 | 252000 | 21.00 | 0.048 | 0.4 | -42 | 180 |

\* Noise power measured in 2500 Hz bandwidth.

# Appendix C: Source Code

*WSJT-X* 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, g++, g95 or gfortran, portaudio, fftw3, and hamlib. For compiling in Windows I recommend installing the MinGW package.

The full source code for *WSJT-X* can be downloaded by using the command

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

1. Experimental versions of *WSJT\_X* have offered submodes JT9-2, JT9-5, JT9-10, and JT9-30 with longer sequences. They trade reduced throughput for smaller bandwidth and increased sensitivity. The slowest experimental submode, 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. These submodes require very stable oscillators in both transmitter and receiver, and are generally useful only at MF and LF. [↑](#footnote-ref-1)
2. Joe Taylor, K1JT: ["The JT65 Communications Protocol"](http://physics.princeton.edu/pulsar/K1JT/JT65.pdf) (QEX, September-October 2005). [↑](#footnote-ref-2)