FLDIGI Users Manual

3.21

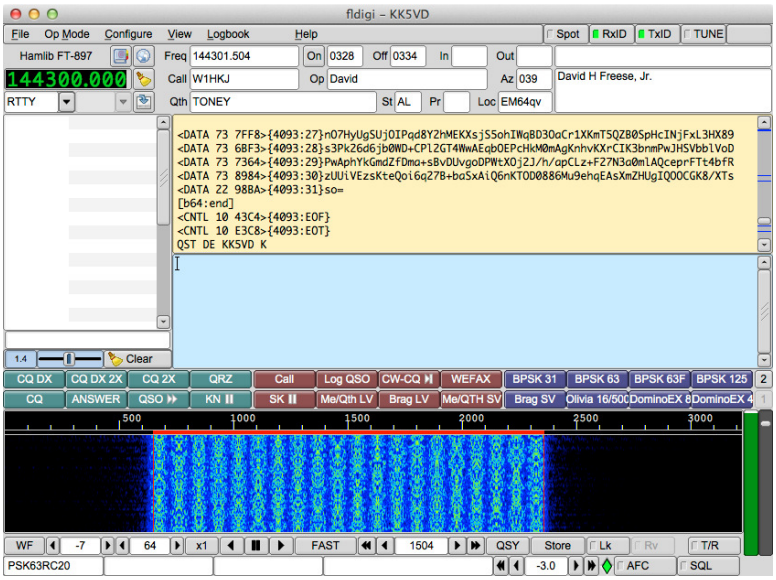
Generated by Doxygen 1.8.6

Tue Apr 29 2014 15:10:39

Contents

Chapter 1

FLDIGI Users Manual - Version 3.21



1.1 Fldigi Configuration and Operational Instructions

- Configuration
- Modems
- Operating
- Logging
- Macros
- Developers
- License(s)
- Recognitions

Return to Top of Page

Chapter 2

Configuration

The first time you execute fldigi you should resize the main window to suit your screen dimensions. Then adjust the divider line between the Rx and Tx text widgets..

Fldigi contains many configurable items, to specify operator data, user interface, and modem characteristics. The application also saves many state variables between executions. It will start up in the state that it was last used.

2.1 User Interface configuration

You should initially configure the following:

- Configure Operator
- Sound Card Configuration
- Rig Control
- New Installation

2.2 Windows Specific Install / Config

- Installing Fldigi on Windows
- New Install Wizard

2.3 Other Configuration options

When the program receives and transmits digital signals and your rig control is satisfactory then you can continue configuring other aspects of the program:

- Callsign DB Configuration
- Colors and Fonts
- PSKmail Configuration
- User Interface Configuration - Browser
- User Interface Configuration - Contest
- User Interface Configuration - General

- User Interface Configuration - Macros
- User Interface Configuration - WF Controls
- Waterfall Configuration
- Working Logs

2.4 Command Line Switches

Additional configurational items are available from the command line.

See Command Line Switches for details.

2.5 Modem Configuration Options

You can configure each modem type to suit your particular operating needs, but the defaults should be satisfactory for most users.

- Contestia Configuration
- CW Configuration
- DominoEX Configuration
- Feld Hell Configuration
- ID Configuration
- Miscellaneous Configuration
- MT63 Configuration
- Olivia Configuration
- PSK Configuration
- RTTY / FSK Configuration
- Thor Configuration

When you have completed the configuration go to the **Configure** menu and select **Save config** or press the "Save Config" button on the configure dialog box. The program will write the file ~/.fldigi/fldigi_def.xml.

Return to Top of Page

Return to Main Page

2.6 Configure Operator

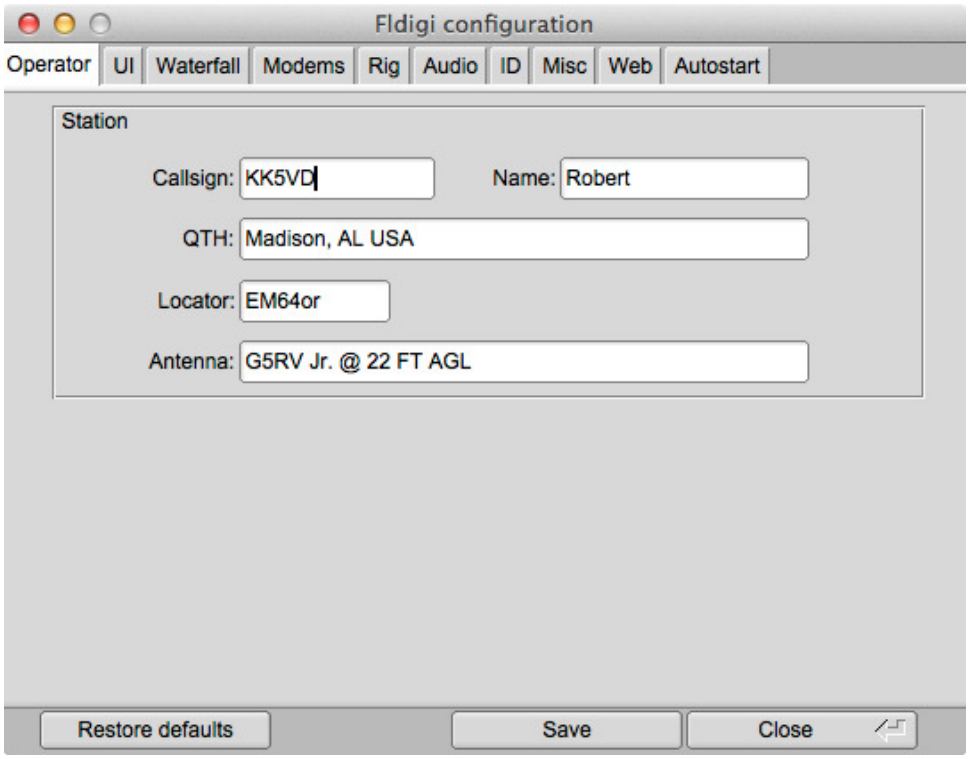


Figure 2.1: Operator

Enter your personal information on the **Operator** tab of the configuration dialog. This information is used by some of the macro expanders.

The antenna information is required if you elect to report to the spotting web site, PSK reporter.

Your locator data is also used for automatically computing Azimuth to a remote locator when that is available from an on-line database Call query.

Return to Top of Page

Return to Main Page

2.7 Sound Card Configuration

A few words about sound I/O on the PC. "You are in a maze of twisty little passages, all alike".

PortAudio, PulseAudio and OSS are different ways in which fldigi can access your sound card through the various sound systems.

OSS was the first audio backend in fldigi. It works with the Linux sound system of the same name, which has now been replaced by ALSA but is still supported via an emulation layer. Its only advantage, as an audio backend, is that it's simple and doesn't require any external libraries.

The PortAudio backend was written subsequently to support OSS on Linux and FreeBSD, ALSA and JACK on Linux, CoreAudio on OS X, and also the various sound APIs on Windows – all through the same PortAudio library.

PulseAudio is more than an audio hardware access layer; refer to its website for a summary of what it does. Fldigi supports it mainly because many Linux distributions are now integrating it with their desktops, but also because it has a few interesting features:

- it can take care of the resampling and volume control for us,
- it can stream audio over the network, and
- it makes it easier to run multiple fldigi instances (all accessing the same sound card).
- it provides mixer controls for input and output audio streams
- it remembers which hardware is used for each application it serves, and it remembers the mixer levels associated with that application

In the future it might be possible to replace all of these with a single backend, without any loss of functionality, performance, sound system or platform support. That'll be the day! Until then:

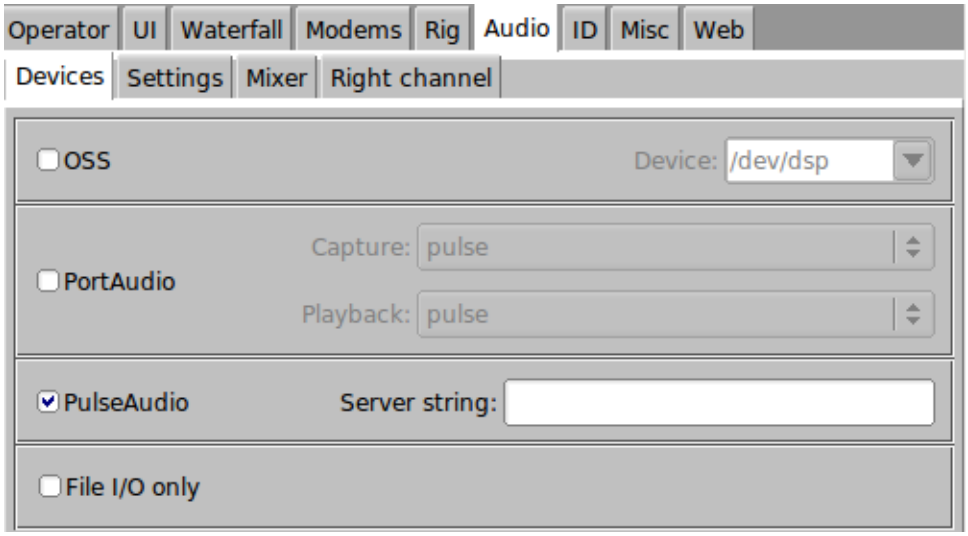


Figure 2.2: Audio Devices

On Linux:

- Use PulseAudio if your Linux distro ships it, and you already have the pulseaudio daemon running (this is the case with Fedora 8/9 and Ubuntu 8.04, probably also with openSUSE 11.0). Or if you want networked audio, etc. etc.
- Otherwise, use PortAudio and select a device from the list(s). PortAudio is also the best way to access JACK, through which you can use other programs as audio sources/sinks – particularly useful with SDR software. As with PulseAudio, you can select different capture and playback audio devices.
- The OSS backend should be used only as a last resort. Note that it has not been updated to support user-configurable sample rates.

On Windows:

Use the PortAudio and select the device from the list(s).

Select the *SndCrd* tab on the configuration dialog.

On Linux Fldigi can interface to the sound card using either the OSS, the Portaudio, or the PulseAudio. Each of the appropriate libraries must be present on the computer to use that particular sound i/o.

On Windows Fldigi uses the Portaudio sound driver only.

It is also possible to configure Fldigi with File I/O only, which is useful for testing the application without an interface to the sound card. In the File I/O only configuration you can record and playback audio files in a number of different formats including the "wav" format associated with the Windows operating system.

The program will find all active sound cards and the associated drivers for both. Select the sound card and driver type that will be used with the program. I recommend using the Pulseudio device driver if that is available on your Linux distribution.

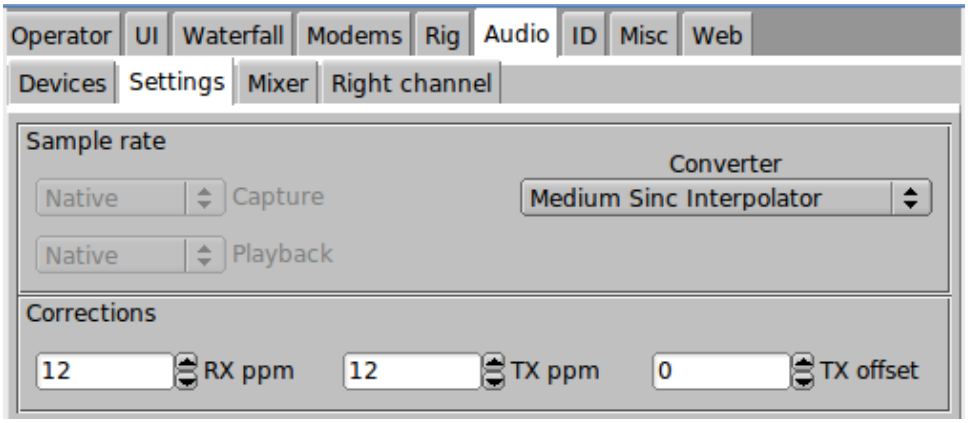


Figure 2.3: Config Audio Settings

If Pulse or Port audio is selected then you can either allow the program to use the auto detect to determine the best sound card sampling rate, or you can pick from the drop down list. If you know your RX and TX sound card oscillator correction factors you can enter them now. If not you can determine the RX rate correction using a special WWV modem built into *Fldigi*. The decoder and encoder logic for each of the various modems require a specific sound card sample rate which may not be the the actual sound card sample rate. The conversion between the modem sample rate and the sound card sample rate is accomplished by one of a set of sample rate converters.

Sound card oscillators may have a slight error in frequency that causes their sampling rate to not be the value specified. This error is usually small enough to be measured in a parts per million. Fldigi uses a technique called rate conversion to correct the sampled waveform for this error. The error can be measured and the correction factor determined by using the WWV calibration modem. The supporting library used for the converter provides several different levels of conversion, Best, Medium, Fastest and Linear Interpolator. The default, Medium interpolator, will be satisfactory for most sound cards. If you are running fldigi on a computer with limited cpu power you might find it necessary to select one of the more cpu efficient converters, either Fastest or Linear. Each gives progressively poorer performance but use fewer cpu cycles to perform the frequency conversion. You should also be sure that the cpu type is set to Slow cpu on the miscellaneous configuration tab.

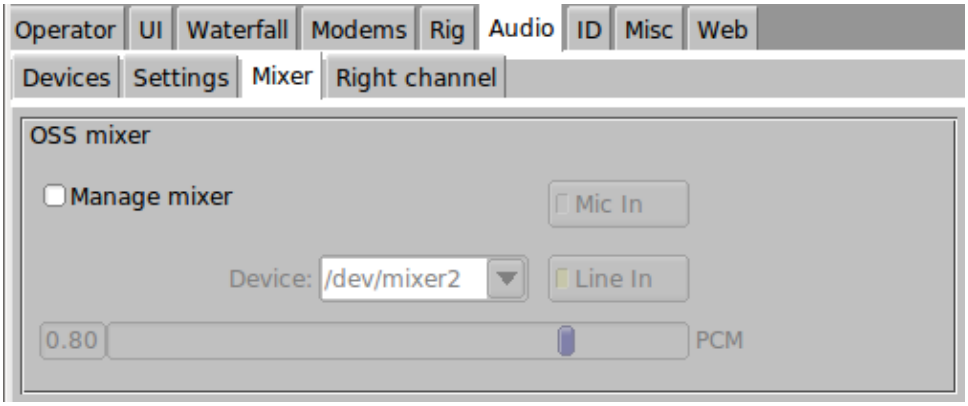


Figure 2.4: Audio Mixer

Mixer controls are only active on Linux using OSS, ALSA backends and if the distribution provides application layer mixer support. Select whether you will be using Line-In or Mic-In for the audio connection from the receiver output. Fldigi ALWAYS expects to use the Line-Out for driving the transmitter audio. Set the PCM level for your sound card. If you check "Manage mixer" then the Tx and Rx "volume" controls on the main fldigi dialog will be active.

2.7.1 Right Channel Audio Output

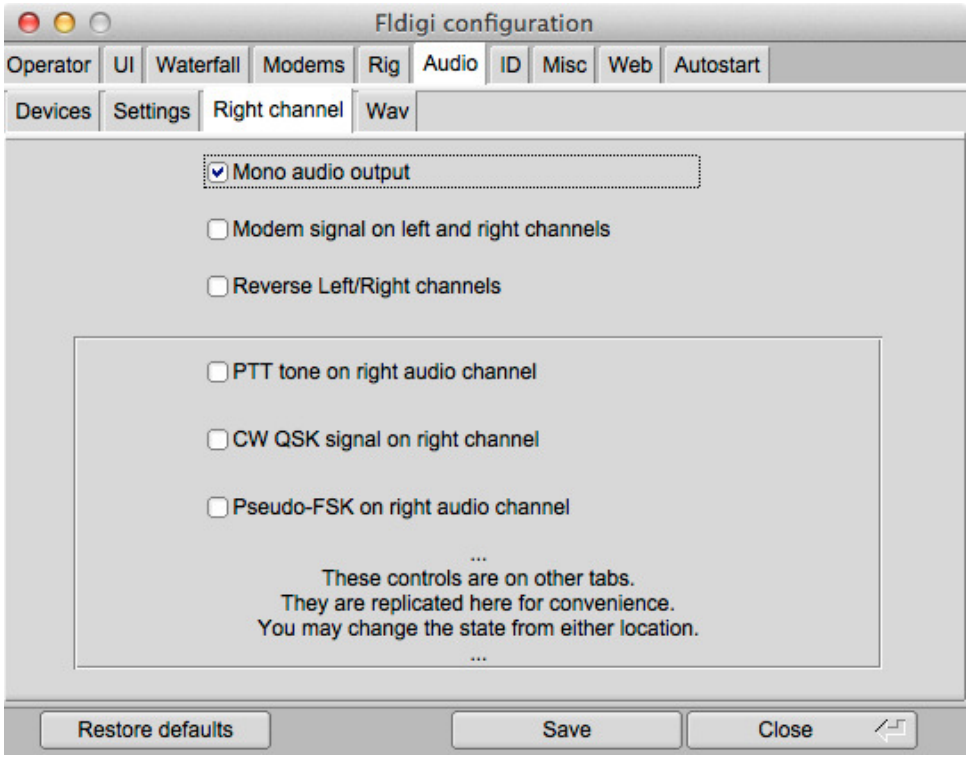


Figure 2.5: Right Audio Channel

You may elect to make the right and left channels both contain the modem signal. Or, you may reverse the right and left channel audio signals. These two controls only change the function of the audio output.

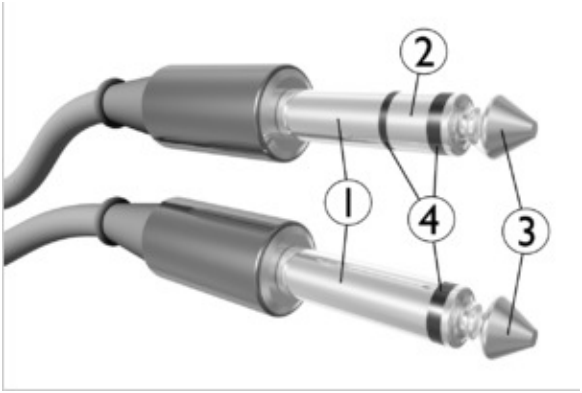


Figure 2.6: Stereo Plug

1. Sleeve: usually ground
2. Ring: Right-hand channel for stereo signals, negative polarity for balanced mono signals. Fldigi uses this channel for special controls signals.
3. Tip: Left-hand channel for stereo signals, positive polarity for balanced mono signals. Fldigi uses this channel for Rx and Tx audio.

The PTT, CW QSK and Pseudo-FSK items are found on their respective configuration tabs. They are replicated here for your convenience and these controls may be changed on this or the other tabs.

2.7.2 WAV File Sample Rate

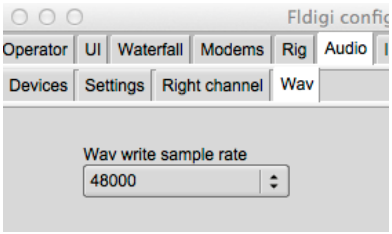


Figure 2.7: Sample Rate of Saved WAV Files

User selectable WAV file sample rate.

2.7.3 Multiple sound cards

In systems with multiple sound cards they will not always be in the correct sort order on boot. This may cause problems not only with fldigi, but other apps that depend on a certain sound card. The work around is not that difficult and will reliably place the preferred sound card in the correct slot. Sound cards are numbered from 0 on to however many cards you have in your computer. Usually its only 2.

The 1st step is to determine the correct id of the cards in your system. Open a terminal and issue the following :: aplay -l

Here is an example of what you may see.

```
**** List of PLAYBACK Hardware Devices ****
card 0: ICH6 [Intel ICH6], device 0: Intel ICH [Intel ICH6]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: ICH6 [Intel ICH6], device 4: Intel ICH - IEC958 [Intel ICH6 -
IEC958]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: AudioPCI [Ensoniq AudioPCI], device 0: ES1371/1 [ES1371
DAC2/ADC]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: AudioPCI [Ensoniq AudioPCI], device 1: ES1371/2 [ES1371 DAC1]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

If it does not already exist you will need to create the following file:

`/etc/modprobe.d/alsa-options`

You will have to root privileges to create the file, so a system such as Ubuntu that depends on sudo you would enter

`$ sudo touch /etc/modprobe.d/alsa-options`

Using the above example, you edit that file so that it's contents contain

Set preferred order of the sound cards

`options snd-ICH6 index=0`

`options snd-AudioPCI index=1`

The simplest terminal editor that is on just about every distribution is

"nano" so to edit the file you just created you can again use sudo

`$ sudo nano /etc/modprobe.d/alsa-options`

If you want to test this new file, reboot and open a terminal and again issue the command :: aplay -l

You will normally find that your internet browser and/or Flash want to use card 0, so you then may want to consider using card 1 for fldigi

Here is another example with 3 sound systems; mother board ALC1200, a thumb-drive audio codec, and a Signa-Link USB:

```
**** List of PLAYBACK Hardware Devices ****
card 0: NVidia [HDA NVidia], device 0: ALC1200 Analog [ALC1200 Analog]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: Set [USB Headphone Set], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 2: default [USB Audio CODEC ], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

The /etc/modprobe.d/alsa-options contains these lines

```
options snd-NVidia index=0
options snd-Set index=1
options snd-CODEC index=2
```

Notice that each line is uniquely related to the aplay -l report

Return to Top of Page

Return to Main Page

2.8 Rig Control

Figure 2.8: Initial Rig Setup

CAT not enabled

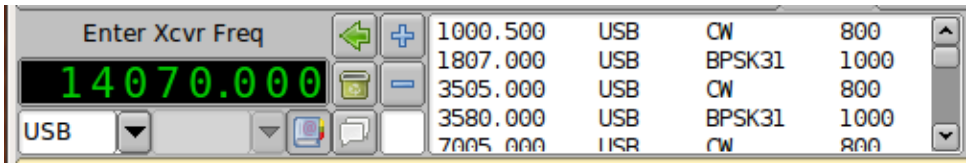


Figure 2.9: Docked Rig Control

CAT not enabled, Manual Entry of transceiver frequency

Note: The same control is also used for both manual entry of the transceiver frequency or with full CAT control. When no CAT is available the control is simply a convenient way of keeping track of the transceiver USB/LSB suppressed carrier frequency, the mode and the audio tracking point. If fldigi is being used with an FM transceiver you probably should enter the simplex frequency or the input frequency of a repeater being used. This frequency value is used with the waterfall audio frequency to compute the logged frequency. The logged frequency value will only be correct for LSB and USB operation.

The frequency/mode pick list is displayed when the book button is pressed. Pressing the book button a second time will restore the original logging panel.

The pick list buttons control selecting, adding and deleting entries in the frequency/mode list.

- add the current frequency / mode / audio track point to the list
- select the current list entry
- delete the highlighted entry from the list
- delete all entries from the list (a warning prompt will appear)
- show active frequencies based on either the entry field to the right or the stations locator, see pskreporter/spotter.
- entry field for active frequencies search, for example "EM."

The browser list contains frequency, sideband, modem type and audio frequency. The list is saved when fldigi is shut down.

The combo box on the left will allow the selection and control of the operating mode of the transceiver.

The combo box on the right will allow the selection and control of the transceiver bandwidth.

The frequency display is in fact a set of special buttons. Each digit may be left-clicked to increment in frequency by that digit value, or right clicked to decrement by that digit value. The leading digits will follow suit if a decade rollover occurs. You can also place the mouse cursor on a digit and then use the mouse wheel to roll the frequency up and down.

Manual entry of frequency can be accomplished by clicking on any digit and then entering the numeric value in KHz. Don't forget the decimal point if you are entering a fractional KHz value.

The mode combobox, the bandwidth combobox and the frequency display also annunciate the current transceiver status. If you change operating mode on the transceiver, that will be annunciated in the respective combobox and fldigi will adjust any internal parameters accordingly. Fldigi queries the transceiver 10 times per second to maintain a lock step with the transceiver.

2.8.1 Rig Configuration

Hardware PTT control

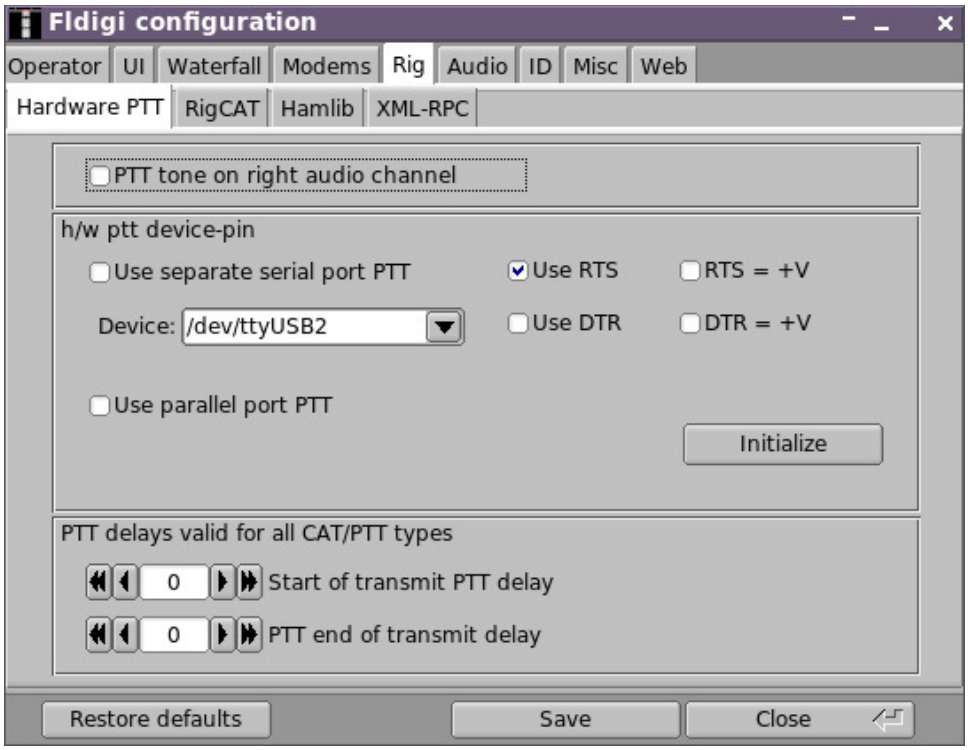


Figure 2.10: Hardware PTT control

Right Channel VOX Signal

Fldigi can generate a 1000 Hz tone for the duration of the PTT keydown period. A simple tone detector/filter and transistor switch can be used to generate a PTT signal from this sound card output. Jim, W5ZIT, has provided details on building an interface for this type of PTT control.

Serial Port using DTR or RTS

The simplest rig control is just being able to control the push to talk via an external transistor switch. You set this type of control on the first configuration tab for rig control.

You select this operation by checking the "Use serial port PTT". Select the serial port from the list (fldigi will have searched for available ports). Then specify whether the h/w uses RTS or DTR and whether a + or - voltage is required to toggle PTT on.

You can use a serial port for control with the RTS and DTR pins configured for you particular interface. The program allows you to use RTS, DTR or BOTH for the PTT signal. Press the Initialize button to start the serial port.

Parallel Port (Linux and Free BSD only)

Fldigi sets and clears the parallel port pin, PARPORT_CONTROL_INIT, pin 16 on the 25 pin parallel port connector. Keydown sets Pin 16 to +5 volts and keyup sets the voltage to zero.

μH Router (MacOS X)

Simular functionality can be achieved on the Macintosh operating system using 'μH Router' by Kok Chen, W7-AY. See μH Router Website for specific details and requirements. A selectable (check box) option will be available on the Rig->Hardware PTT Configuration panel.

PTT delays

You can accommodate delays in transceiver switching between receive and transmit by adjusting the PTT delays. The control values are in milliseconds. These controls have no effect on external PTT circuits such as those implemented in the SignaLink interfaces. They rely on detecting the audio data stream. You can use a combination of macro tags in a macro key definition to achieve a resolution. For example try a macro definition similar to this to insure that the RSID is sent via a slow FM xcvr (or via a VHF repeater)

```
<TX><MODEM:NULL><IDLE:2.5>
<!MODEM:MT63-500>
<TXRSID:on>
```

Change the idle time value (in fractional seconds) to suit your needs.

2.8.2 RigCAT control

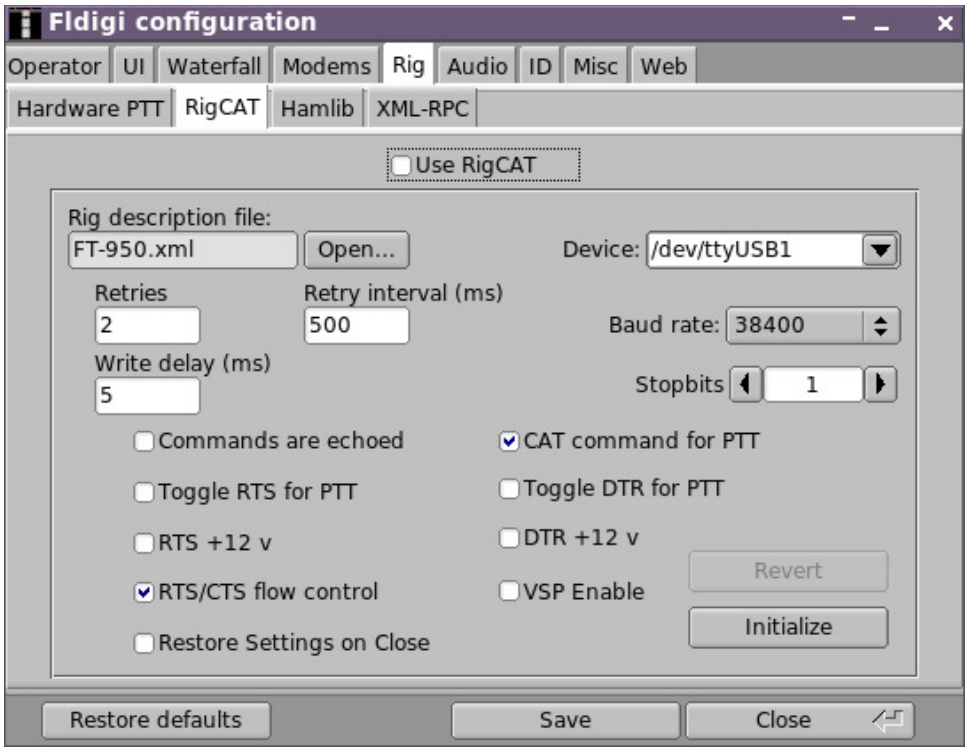


Figure 2.11: RigCAT control

RigCAT is a rig control system similar to hamlib that was developed specifically for fldigi. It uses command / response definitions that are found in various rig.xml files. You can use a rig.xml file specific for your transceiver or write and test one yourself. The easiest way is to adapt an existing rig xml file for a rig that is similar to your own. ICOM almost identical command/response strings for all of its transceiver line. Yaesu rigs have nearly all used unique command/response structures until just recently. The TS-450, TS-950 and others share a similar set of commands and responses.

RigCAT commands and responses are defined in a rig specific xml file which contains all of the required queries and responses in extended markup language format. Please read the specification document rigxml to learn more about this new way of building generic rig interface definitions and how they are used with fldigi. fldigi will look for a file in the $HOME/.fldigi/rigs directory for all files with extension ".xml". These contain definitions for the transceiver indicated by the file name, ie: FT-450.xml, IC-756PRO.xml, etc. You can download the appropriate xml files from the resource directory tree `http://www.w1hkj.com/xmls` or from the archives `web page`. Place the file in your rigs directory and fldigi will find it.

You will need to specify how your PTT will be triggered. This can be using a CAT command, the RTS or DTR pins or none. None would be appropriate if you are using the rig's VOX or an outboard sound card interface such as the SignalLink SL-1+ which produces its own VOX type of PTT. In that case simply leave all of the PTT options unselected.

If you are using a transceiver or a rig interface such as CI-V that echos all serial data you check off the "Commands are echoed" box. That will suppress fldigi trying to respond to a command it just sent to the transceiver.

You may need to try various values of retries, retry interval, and command interval to achieve consistent rigcat control.

Press the Initialize button after setting all of the parameters. If the settings are all correct fldigi should start receiving frequency information from the rig and annunciating them on the rig control frequency display.

2.8.3 Hamlib CAT control

Hamlib is a set of standard libraries for interfacing to a large number of transceivers. The hamlib library system consists of a front end which acts on behalf of all rigs and backends which are specific to each rig.

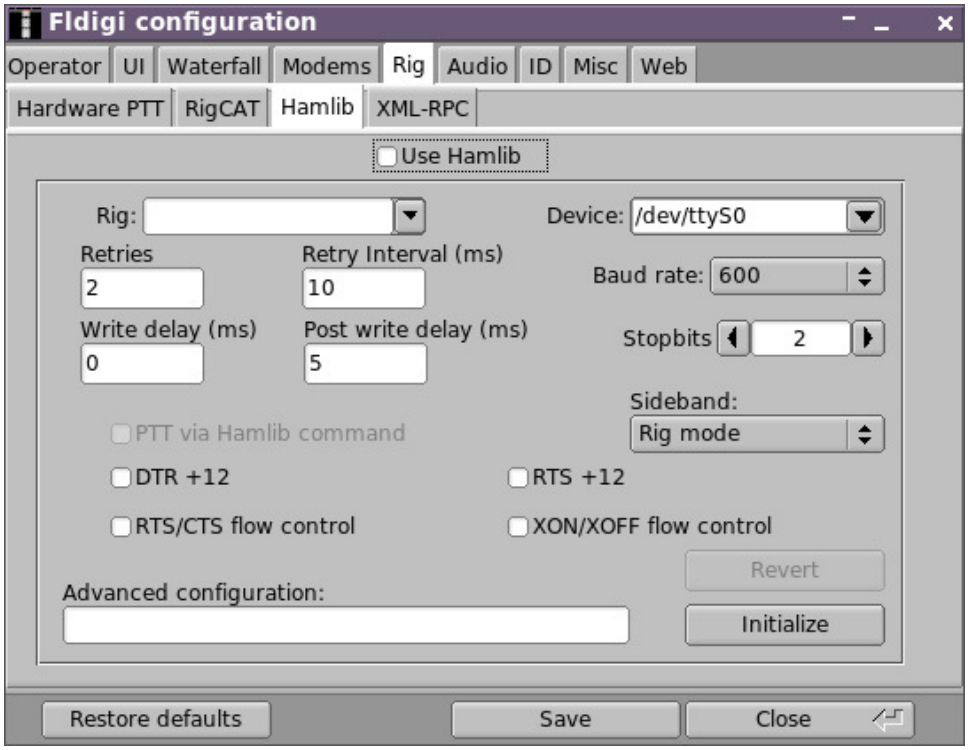


Figure 2.12: Hamlib CAT Control

Select your transceiver from the list of supported units. Then select the serial port and baud rate. If you are familiar with the hamlib library you can send various startup sequences to the rig using the advanced configuration. PTT control can be achieved using CAT commands or via DTR / RTS on the same port as the control comms. You might also need to specifiy whether RTS/CTS flow control is uses (Kenwood rigs use this quite often) or if Xon/Xoff flow control is used.

You may need to try various values of retries, retry interval, and command interval to achieve consistent hamlib control.

Press the Initialize button after setting all of the parameters. If the settings are all correct fldigi should start receiving frequency information from the rig and annunciating them on the rig control frequency display.

2.8.4 Xml-Rpc CAT

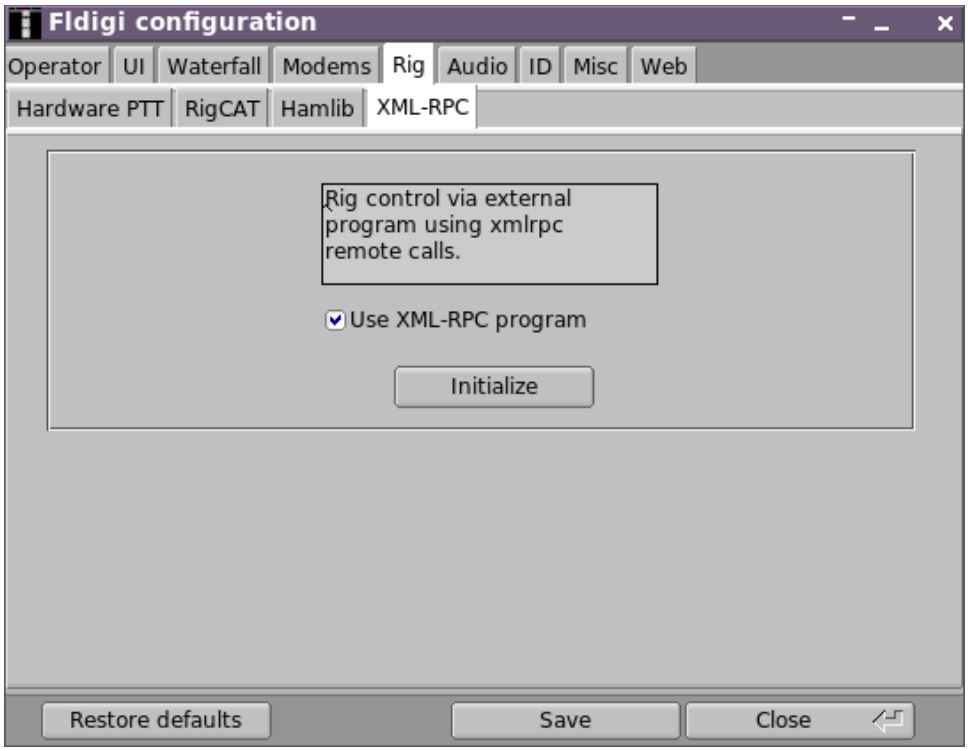


Figure 2.13: Xml-Rpc CAT Control

Xml-Rpc allows third party software to control various aspects of fldigi operation including but not limited to rig control. This is the data interface that is also used by the program **flrig**, a fldigi companion transceiver control program.

If you are using a third party interface such as DxKeeper Bridge you might be instructed to select this method of CAT.

Return to Top of Page

Return to Main Page

2.9 New Installation

fldigi's opening screen looks like the following when starting fldigi for the first time or when setting up a second or subsequent instance using the –config-dir command line switch. The Wizard has been completed and the callsign, W1HKJ, entered.

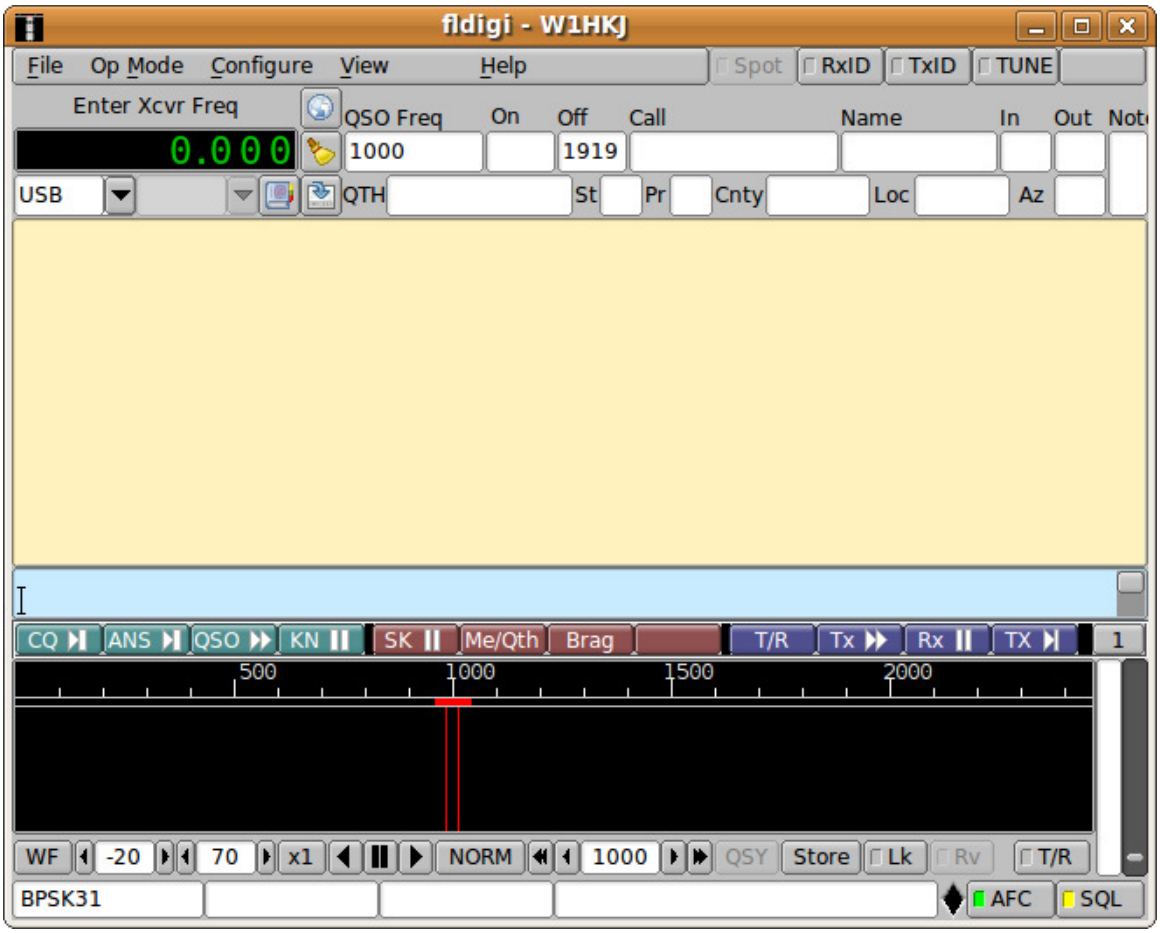


Figure 2.14: Main Dialog New Install

Fldigi will create a working files folder, multiple sub folders and also populate them with a set of default files. The working files folder is different on the different OS.

Operating System	Folder/Directory
Windows	C:\Documents and Settings\<username>\fldigi.files
Vista	C:\Users\<username>\fldigi.files
Linux	/home/<username>/.fldigi
Macintosh	/Users/User_Login_Name/.fldigi

After closing the application the working folder will contain the following folders and files:

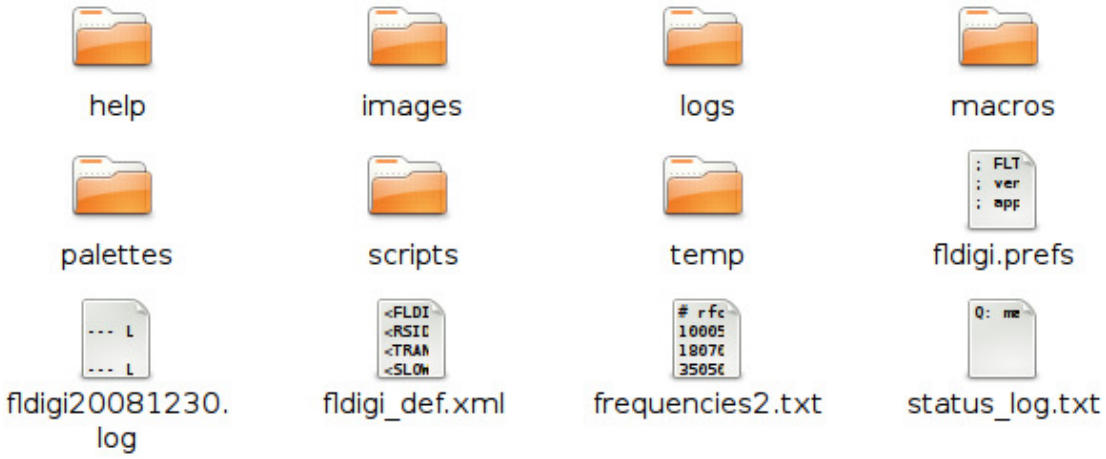


Figure 2.15: Working Folder

The help, images, logs, scripts and temp folders will be empty. They will contain program created files as you use the program or you may post files in those folders for use by fldigi. Images to be sent with the MFSK pic mode should be placed in images. Your logbook database will appear in logs. If you are running on Linux then you can use various scripts to enhance the macro language that fldigi supports. The temp directory holds files that are transitory and you can safely delete those files between sessions. The 5 files that appear initially are:

fldigi.prefs	contains variables that describe the status of fldigi when last used. This is an ASCII text file that you can safely read. You should not edit or change this file.
fldigiYYYYMMDD.log	this is an historical log of all the received and transmitted text during the day for which the log refers
fldigi_def.xml	contains variables that relate to all of fldigi's configurable items. This is an ASCII text file that conforms with the XML specification. You can safely read this file but should not edit or change it.
frequencies2.txt	an ASCII text file that contains the default (and / or modified) entries for fldigi's rig control process
status_log.txt	a log of events for the most current fldigi execution. This file will contain information relative to any errors that my occur and is important for debugging purposes.

The macros folder contains a single file: macros.mdf. This is an ASCII text file that contains the default macro definitions. After running fldigi for a while and creating your own sets of macro definitions there will be additional $*$.mdf files located here.

The palettes folder contains the following files:

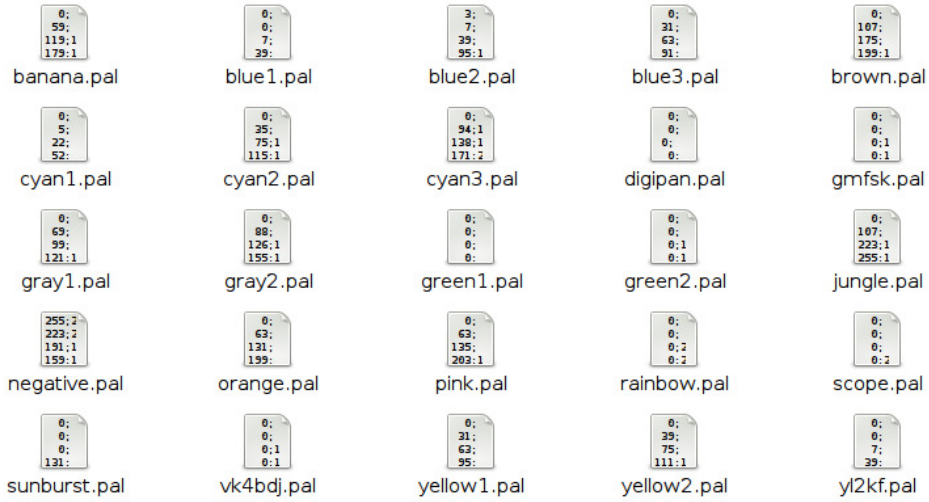


Figure 2.16: Folder Palettes

Each of these is a palette definition file that is used to modify the appearance of the waterfall. Fldigi has a palette editor that enables you to modify these default files or to create your own. The file format of these files is identical to the palette files used by DigiPan. The final color rendition might be a little different as a result of using different painting functions. The file digipan.pal contains:

```
0; 0; 0
0; 0; 62
0; 0;126
0; 0;214
145;142; 96
181;184; 48
223;226;105
254;254; 4
```

Don't bother trying to modify these using an editor. The palette editor is much easier to use and will keep you from wrecking havoc with the program.

The easiest way to find the working files folder is to start fldigi and then select the menu item File/Show config.

Return to Top of Page

Return to Main Page

2.10 Installing Fldigi on Windows

The port of fldigi to the Windows operating system is built using cross-compilers on Linux. The cross-compilation environment is created using mingw32.

Installing fldigi on windows is very simple. Simply execute the installer program and both fldigi and flarq will be installed in the default programs directory structure for the specific Microsoft version that is being targeted. Desktop icons and desktop menu items will be created. An uninstaller link will be created on the desktop menu.

Click on the desktop icon to start the application. If this is a new installation you will be guided through the inital configuration with a new install wizard.

Resize the main dialog to suit your screen. Adjust the Rx/Tx divider to your liking. Then set up the operator and sound card configuration items; configuring fldigi. When you have fldigi receiving and decoding signals you can exit the application which will allow you to save your configuration settings.

Now open up the following folder using your windows file explorer if you do not have a login name and password:

On XP	C:\"Documents and Settings\<urlogin>\fldigi.files"
On W2K	C:\"Documents and Settings\<urlogin>\fldigi.files"
On Vista/Win7	C:\User\<urlogin>\fldigi.files</td>

where <urlogin> is the name with which you log onto the computer.

All of these files were generated by fldigi when it first started. The files with the extension pal are palette definition files. The file "macros.mdf" contains the macro definitions which you can change using the macro editor. fldigi.prefs and fldigi_def.xml are used for storing the application state and configuration items respectively. With the exception of the location of this folder the operation of fldigi on windows is identical to linux. In all instances where the help files make reference to $HOME/.fldigi you should be substitute the appropriate directory for XP or Vista/Win7.

Please take the time to read and reread the on-line help file. Better yet download the Adobe Reader file so that you can view the help locally without needing access to the internet. Fldigi is a large complex program with many ways for the user to customize its operation to his or her hardware environment.

2.10.1 Special Note for Vista/Win7 from user!

I am trying to install the latest and greatest version of FLDIGI and FLARQ on a new Toshiba laptop computer that runs Vista/Win7 / Win7. The main screen of FLDIGI comes up ok and it says that no call sign has been set and down in the bottom center right there is an **error message about there not being the sound card that it thinks it should see**. I understand the messages as the program needs to be configured. The problem is that when I click on 'Configuration' the main screen grays out and it seems to go off into never never land and not comeback or put up the configuration window. At that point the only thing that works on the main screen is the close button. What am I missing?

I solved the mystery! Cockpit error, kind of! I tried installing MULTIPSK and got the "no sound card error" also. Went into the Control Panel to see what was going on. When I looked at sound input it said there was NO microphone plugged in. Then the light came on and I remembered that you MUST have a microphone plugged in when using the Vista OS or it doesn't think there is a sound card in the computer!!! Well this is my first encounter with Vista and how am I suppose to know or remember that little quark on an OS I've never used before??

Once I plugged a mic into the sound card input both programs worked just fine.

Vista/Win7 requires either a **microphone** or a **line-in** device actually plugged into the 8 mm audio jack before the sound driver reports that there is an audio capture device.

2.10.2 Installing fldigi programs on thumb drive

Create a folder on a USB drive (any removable read/writable media). Suggest using the folder name NBEMSapps. Then copy the following files to that folder.

- fldigi.exe
- flarq.exe
- flmsg.exe
- flamp.exe

Optional installs are the remaining suite of FLDIGI programs.

Note: Dynamic libraries (mingwm10.dll, pthreadGC2.dll) are no longer required on current versions of the FLDIGI suite of programs.

Create a file in that folder named NBEMS.DIR.

Then start fldigi etc, from within that folder. Double clicking on the executable from the File Explorer is probably the easiest way to do that. The standard data folders for each application will be created and used from within this folder, for example if the thumb drive were mounted as drive E:

```
E:\NBEMSapps\fldigi.files
```

```
E:\NBEMSapps\NBEMS.files
```

Return to Top of Page

Return to Main Page

2.11 New Install Wizard

New Install Wizard

The new installation wizard borrows from the normal configuration dialogs. You will find information on setting each of the wizard dialog pages on the associated configuration link.

Figure 2.17: Wizard Configuration

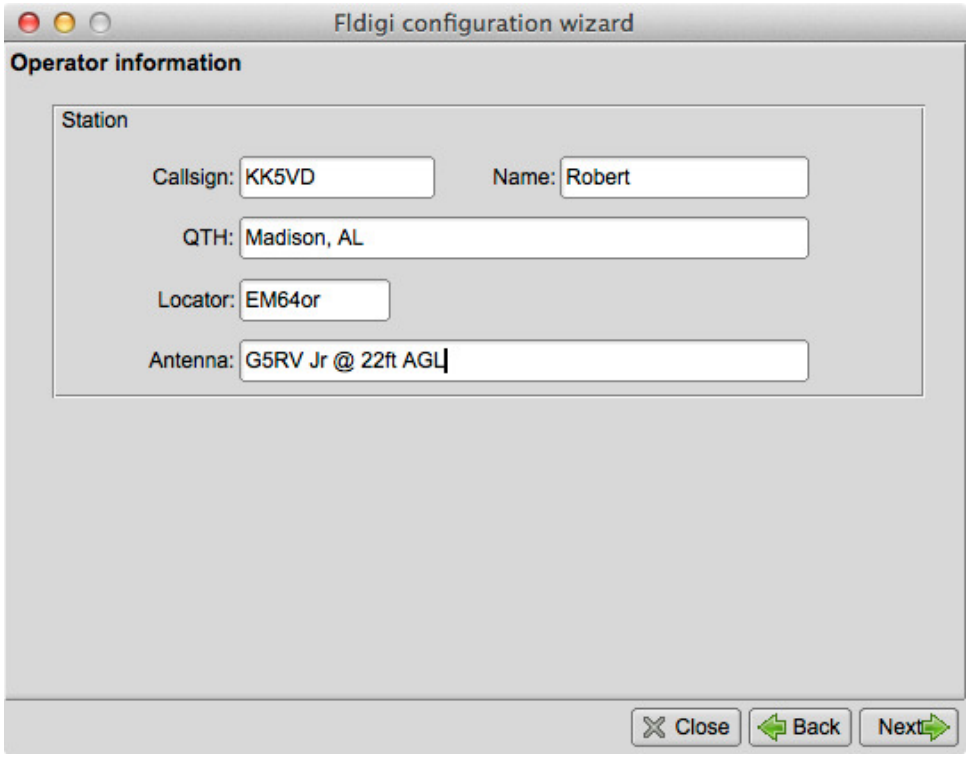


Figure 2.18: Operator Configuration

See Operator Configuration

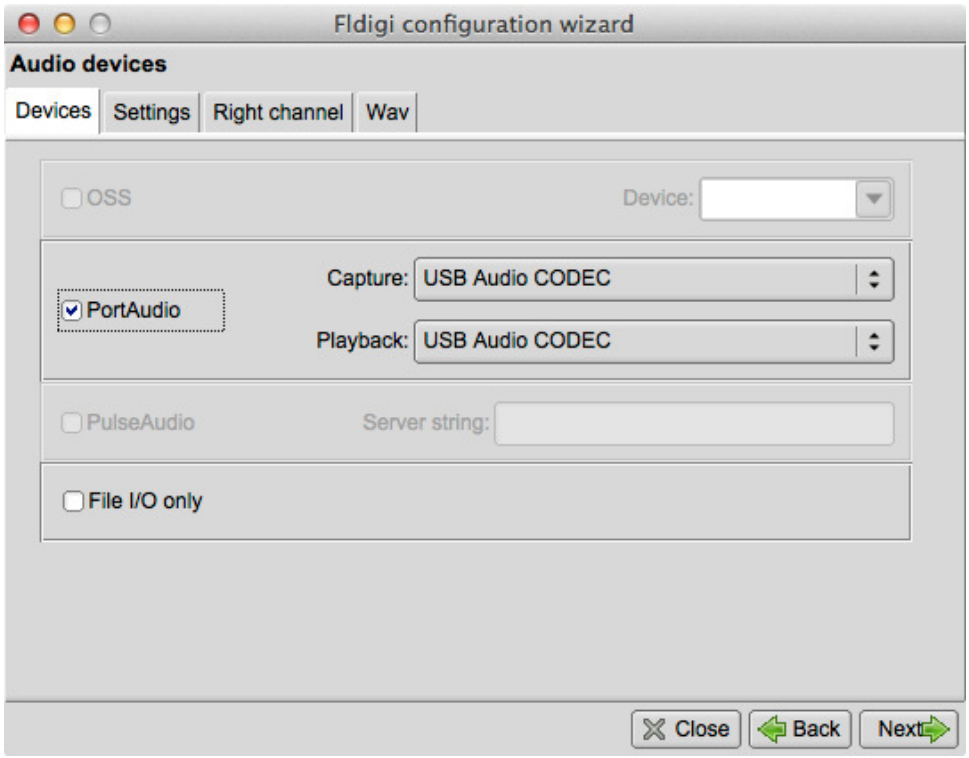


Figure 2.19: Sound Card Configuration

See Sound Card Configuration

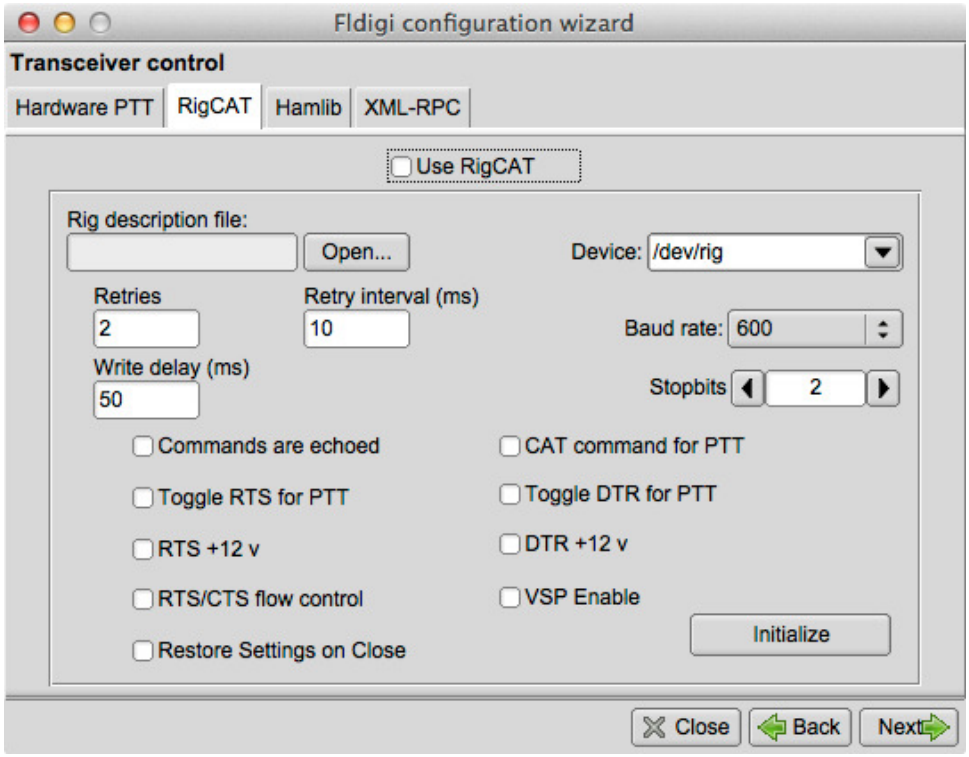


Figure 2.20: Transceiver Configuration

See Transceiver Configuration

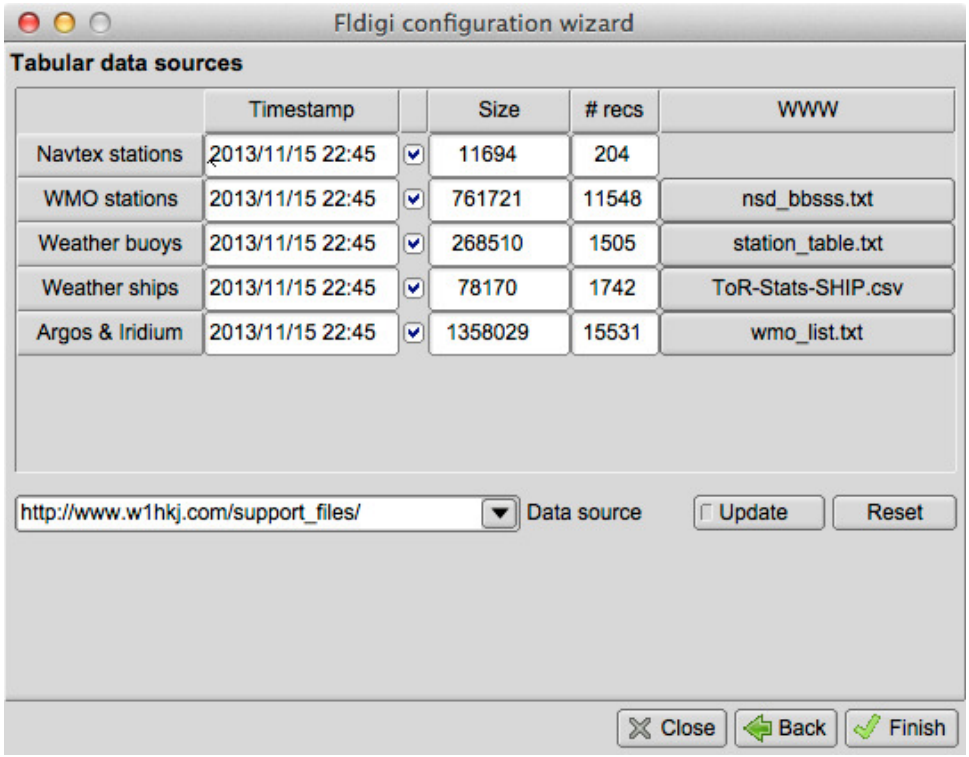


Figure 2.21: Tabular Configuration

See Tabular Data

Return to Top of Page

Return to Main Page

2.12 Callsign DB Configuration

2.12.1 QRZ

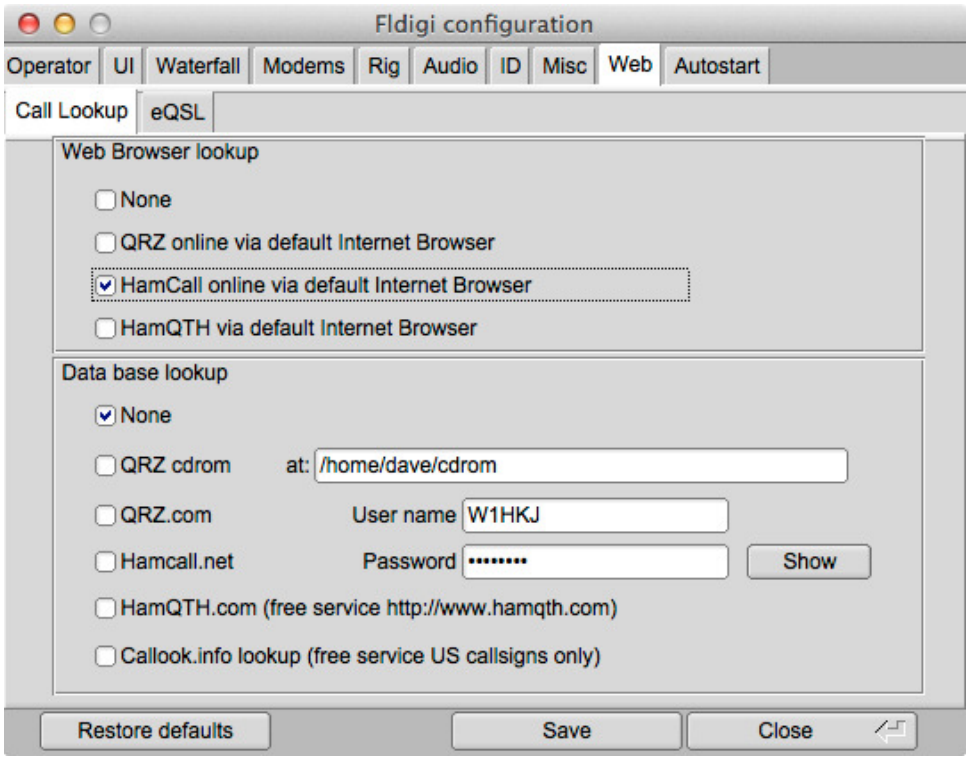


Figure 2.22: QRZ

Fldigi will open a web browser to either QRZ.com or Hamcall.net with the contents of the QSO Call field used as a query string to the on line service. You may find that your default browser needs to be triggered twice on the first such query. That behavior seems to be associated with IE7 but not IE6 for example.

If you have a CD with the QRZ database installed you can use that CD or its' stored contents on a hard drive. Simply specify where the CALLBK directory can be found and enable the QRZ radio button.

If you are a paid subscriber to either QRZ or Hamcall xml database service then you can specify that fldigi use that service for all Callsign data base queries.

OK2CQR provides a very nice callsign lookup service on his hamqth.com web site. There is no subscriber fee to use this service. Simply register and then use your registered user name and password for access. You might want to provide monetary support to Petr if this service meets your needs.

2.12.2 EQSL

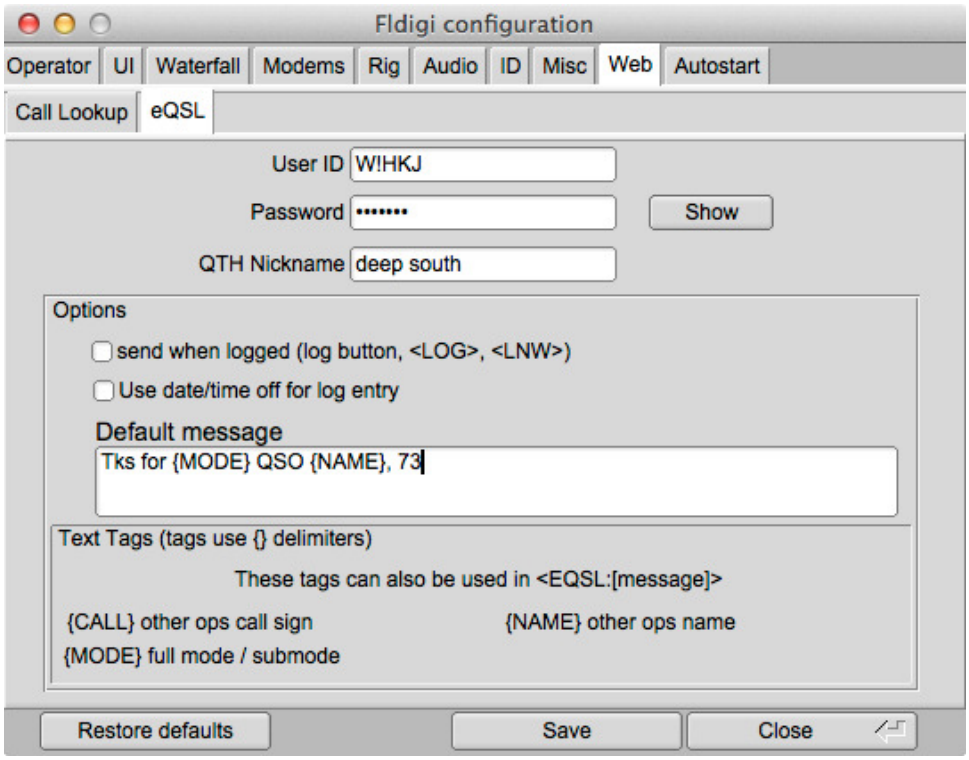


Figure 2.23: EQSL

Configure the eQSL log entry upload with your user ID, password, and QTH nickname. See Macros for additional information.

2.12.3 FLdigi Logging

Figure 2.24: Log It

If you check the "send when logged" button then the logged record will be sent to eQSL when either the log it button is pressed, or when the macro tag <LOG> or <LNW> is executed.

You may elect to use the default of date-time ON or date-time OFF for the eQSL submission.

You may send a message with the log record. If you fill in the default message text then it will be used for that purpose. You might want to use something like:

Tks for {MODE} qso {NAME}. 73, pse eQSL

Return to Top of Page

Return to Main Page

2.13 Colors and Fonts

"System colors" are set by command line switches. The default is black on a white background.

From the Menu **Configure/Defaults** select the menu item **Colors and Fonts** and then select one of the following tabs.

2.13.1 Text Ctrls

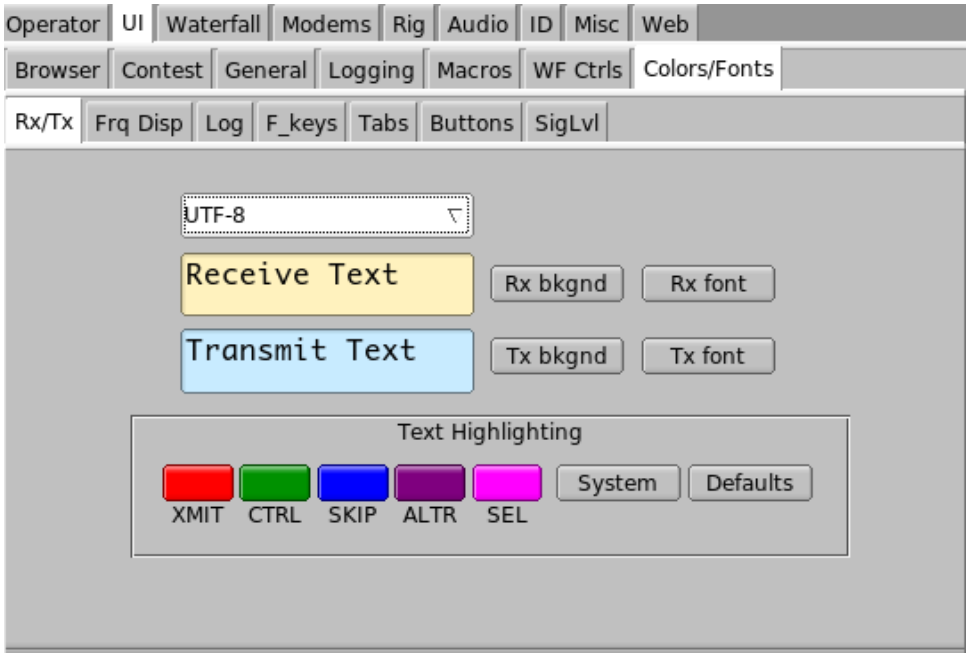


Figure 2.25: Colors Fonts RX / TX

The initial color, font and font-size for the Rx and Tx panel are the default values. You can always return to these by pressing the Defaults button. The background color, font and font-size are independently selectable. The Rx panel displays text in one of 5 colors:

- normal received text - "Rx font" button
- transmitted text - XMIT button
- control characters - CTRL button
- skipped characters (Tx ON/OFF in Tx pane) - SKIP button
- quick view characters - ALTR button
- select text highlight - SEL button

The text widget used for Rx, Tx and Event log displays has been improved to give better performance with proportional fonts. Fixed width fonts still give better performance and are not as demanding on the cpu. There are several very good fixed width fonts that include a slashed zero which are available for both Windows and Linux. If you are using a proportional font and find that the Rx text display gets unresponsive with large amounts of text then you should change to a fixed width font. Do a search on the internet for "Andale Mono" or "Consolas". Both are excellent fonts for this use.

2.13.2 Character Set Selection

Figure 2.26: Character set

This setting affects how fldigi encodes the text that it transmits and how it interprets what it receives. This mainly matters if you intend to transmit and/or receive text containing characters that are not in the English alphabet (for example letters with diacritics, Cyrillic letters, Japanese script and similar). Successful transmission and reception of such symbols is only possible if both your and the correspondent's program are set to use the same encoding. Seeing "strange" characters instead of what you would expect for example "č" always turning up as "è" indicates that there is probably a mismatch between your encoding and the correspondent's.

Leaving this control set to UTF-8 (the default) is strongly recommended as this is a cover-all encoding that enables communication in almost any language and script imaginable. However, there are three cases in which you might want to switch encodings (at least temporarily):

- You never expect to transmit or view any character except for the lower 128 characters (ASCII) of the ANSI data set. This might be true for English to English communications in which you do not want to have decoded noise appear as UTF-8 character renditions.

- You want to communicate using non-English characters, but the correspondent's program does not support UTF-8. In such a case, you should find out what encoding the correspondent is using and change your setting accordingly.

- You expect to exchange a lot of traffic that mostly consists of non-English characters (communication in Cyrillic script being a notable example). UTF-8 encodes each non-English character into a symbol two to four bytes long. Such characters require more time to transmit and thus reduce the effective transmission speed. Choosing another encoding that requires less bytes to be sent might be beneficial if the transmission speed is crucial. For example, a pair of operators wanting to communicate in Cyrillic script might want to choose CP1251 to retain the maximum transmission speed.

Warning: transmission of non-English characters, regardless of the encoding used, requires that the digital mode used be capable of handling 8-bit traffic. The following modes suit this requirement:

- DominoEX

- MFSK

- MT63 (8 bit extended characters must be enabled)

- Olivia (8 bit extended characters must be enabled)

- PSK (all variants)

- THOR

2.13.3 Frequency Display

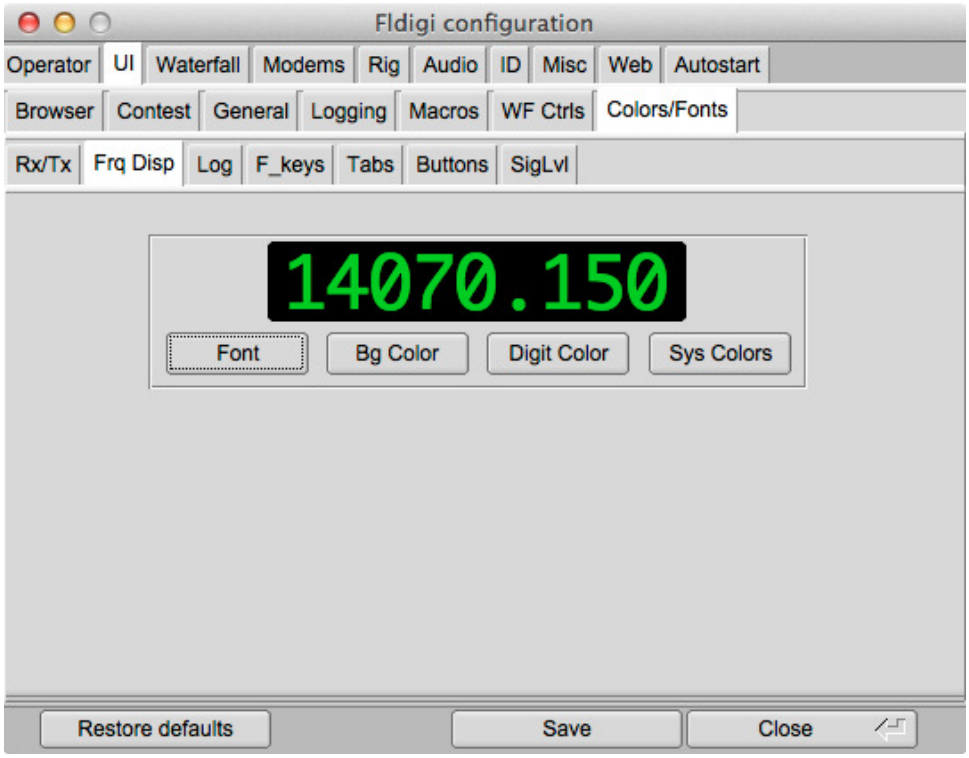


Figure 2.27: Frequency Display

The rig control panel uses a special button for each digit the represents the transceiver frequency. The buttons are responsive to mouse clicks on the upper and lower half with corresponding changes it that unit's value. Unit value is also controlled by the mouse wheel when the cursor is over a particular digit. Select the background and foreground colors to please your overall color scheme and for best visual acuity. The System colors are the same ones that are used by all input and output text controls.

2.13.4 Logging controls

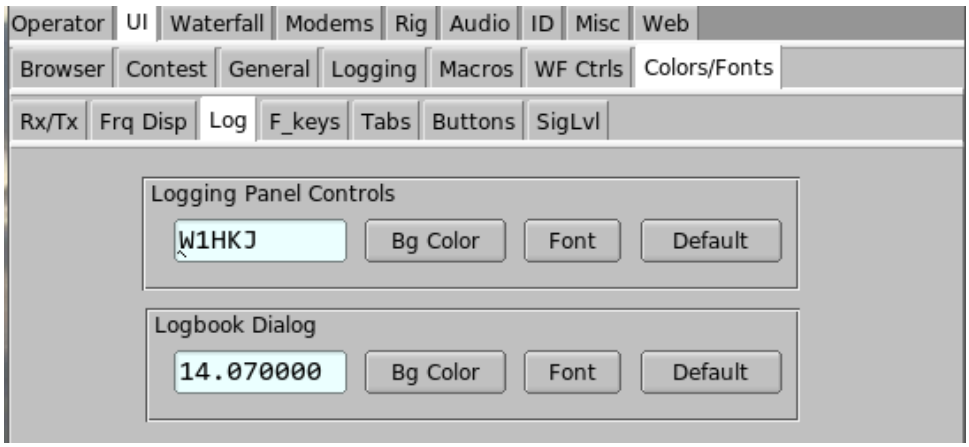


Figure 2.28: Logging controls

You can select the background color, the font, the font color and the font size for both the logging controls on the main dialog and the separate logbook dialog.

2.13.5 Func keys

Figure 2.29: Colors Fonts FKeys

You can color code the macro (function key) buttons in groups of 4, F1-F4, F5-F8, and F9-F12. The background color for each group is adjusted by clicking the respective Bkgnd button. The text color for the button labels is adjusted by clicking on the Label Txt button. The colors will change on these buttons and also on the main dialog as you make these adjustments. The Defaults button restores the colors as shown in this view.

2.13.6 Tab Colors

Figure 2.30: Colors Fonts Tabs

Adjust the color of all tabs to suit your personal taste.

2.13.7 Light Buttons Colors

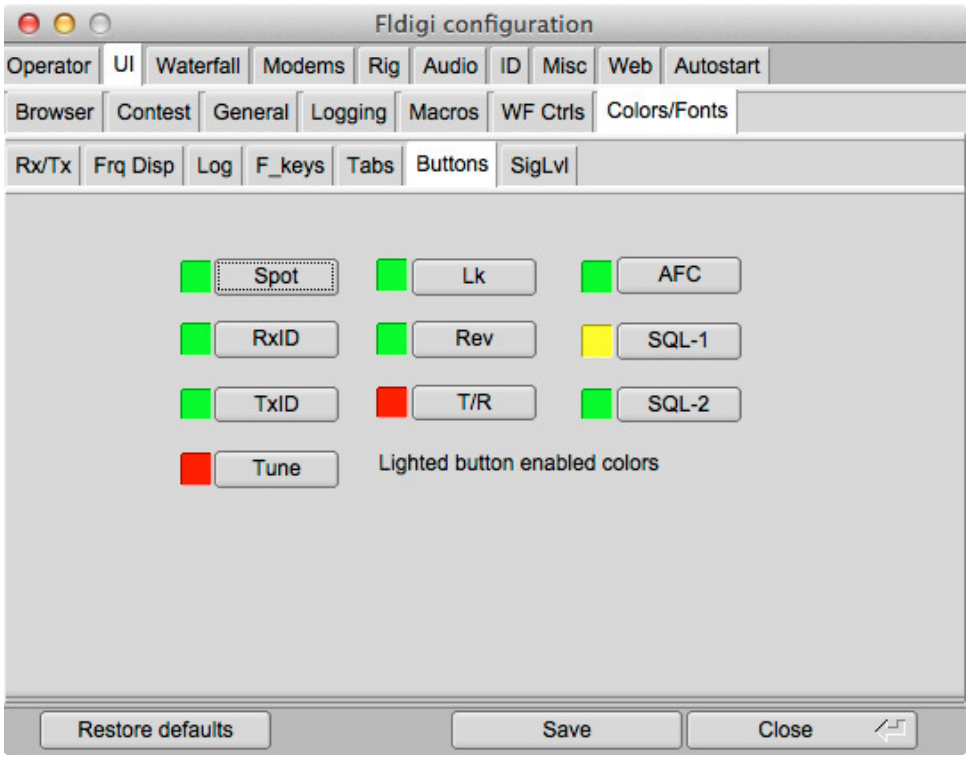


Figure 2.31: Colors Fonts Buttons

- Adjust the "on" color of Spot, RxID, TxID, Tune, Lk, Rev, T/R and AFC button
- Adjust the "enabled" and "on" colors of the Sql button

2.13.8 Signal Level

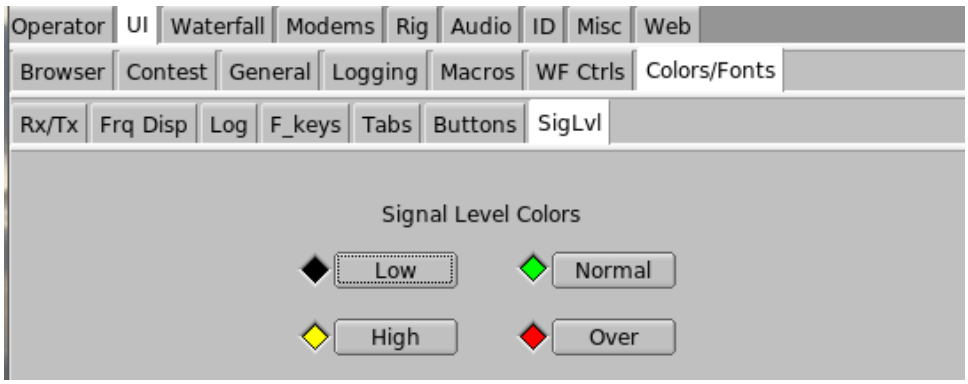


Figure 2.32: Colors Fonts Signal Level

You can select the low/normal/high/over-drive colors for the diamond signal level in the lower right hand corner of the main dialog.

.

Figure 2.33: Level Indicator

Your audio Rx level should be set to allow the loudest signals to drive the indicator into the yellow and never into the red zone.

Return to Top of Page

Return to Main Page

2.14 PSKmail Configuration

Fldigi can act as both a server and a client for PskMail, a separate application from fldigi. The PSKmail specific configuration parameters are all located on the Misc/Pskmail tab of the configuration dialog.

Instructions on setting these parameters are a part of the pskmail installation.

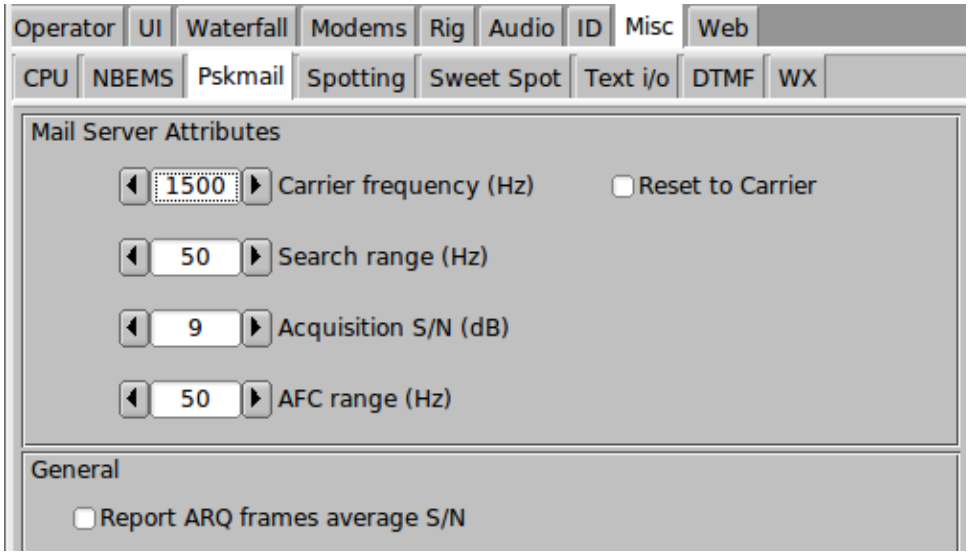


Figure 2.34: PSK Mail Configuration

Return to Top of Page

Return to Main Page

2.15 User Interface Configuration - Browser

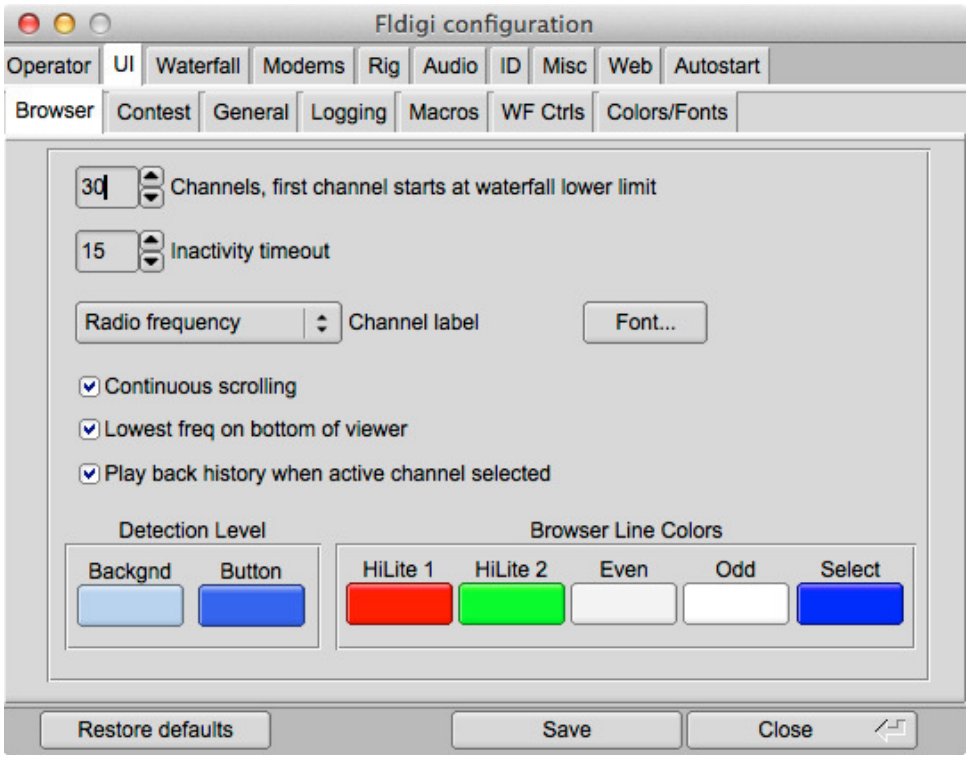


Figure 2.35: UI Browser Configuration

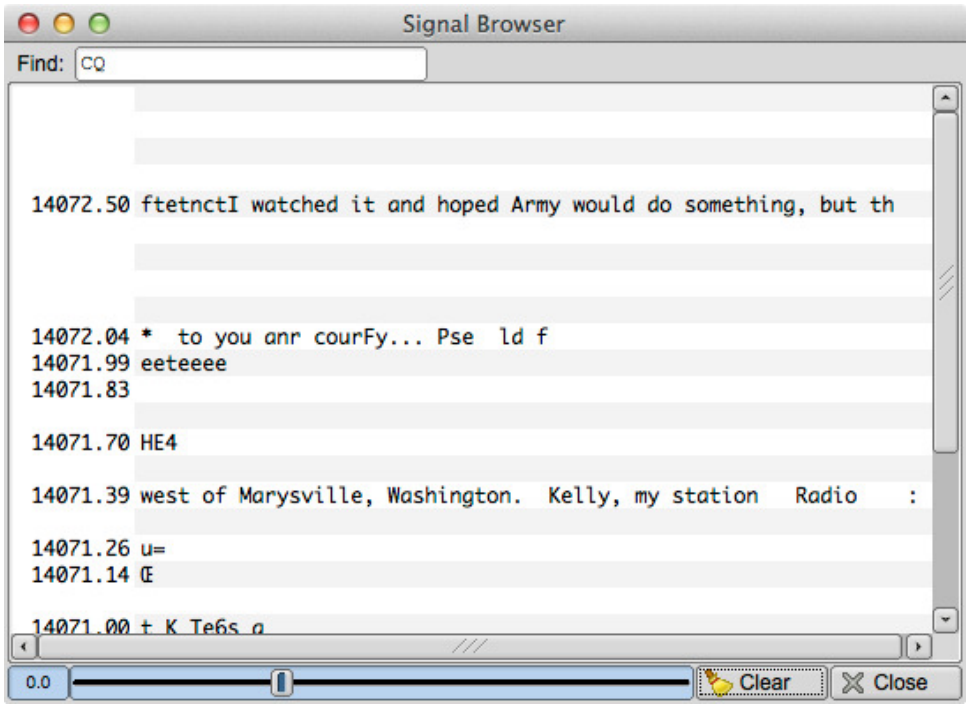


Figure 2.36: Browser Panel

Fldigi can display multiple decoded signals in both PSK and RTTY modes. The multi-channel display is either a separate dialog or an embedded resizable panel.

1) select the number of 100 Hz channels you want visible

2) select the start frequency for the browser (your transceiver may not rx signals below this value)

3) select the inactivity timeout for the browser. After this number of seconds the channel will be cleared and prepared for the next detection cycle.

4) select what kind of label annotation you want on each line

5) select the font and font size to be used in the browser

6) You can enter any text you want to search for in the *Seek Regular Expression* widget. This text can be a simple text snippet such as "CQ" or any regular expression. When the regex is satisfied in a channel the text color for that channel is changed to red. With a regex you can specify a more generic pattern, which means that you can match more things and your search is somewhat noise tolerant. Here is an example for a CQ from a US station (should match most callsigns):

```
>cq.+[aknw][a-z]?[0-9][a-pr-z][a-z]{1,2}
```

This says "cq followed by at least one character, followed by one A, K, N, or W, followed by an optional letter, followed by a digit, followed by a letter that is not q, followed by one or two letters". The search is case-insensitive.

All plain text is a valid regular expression, unless you really had been looking for these metacharacters:

```
>.[{()\*+?|^$</span></big><br>
```

These will have to be escaped with a backslash.

7) select whether you want a marquee type of continuous scrolling, or simply clear the line when it is filled.

8) select whether you want the lowest frequency at the bottom (checked) or the top of browser panel

9) select whether you want the audio stream history buffer to be played back when you select an active channel. The first-in first-out audio history represents the previous 2 minutes of received audio.

10) Both the background and slider highlight colors can be selected for the signal browser detection level control. The default colors are shown in these images.

11) You can set the 2 levels of text hi-lighting that is used in the browser lines

- HiLite 1 - Text color when the regular expression evaluator finds the target text
- HiLite 2 - Text color when your call sign appears in that receive channel

12) You can set the background colors for the odd/even lines and the line selection color

Return to Top of Page

Return to Main Page

2.16 User Interface Configuration - Contest

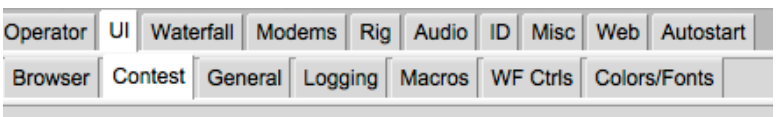


Figure 2.37: UI Tabs

Fldigi supports a generic but robust set of contest functions. In addition to serial-in and serial-out you can capture and transmit three exchange sequences unique to a specific contest. Enter the exchange you want to send for each of the three. You can force the RST in/out to always be 599. That seems to be a norm for many contests. When operating in a CW contest you can have fldigi send cut numbers, T for 0, N for nine.

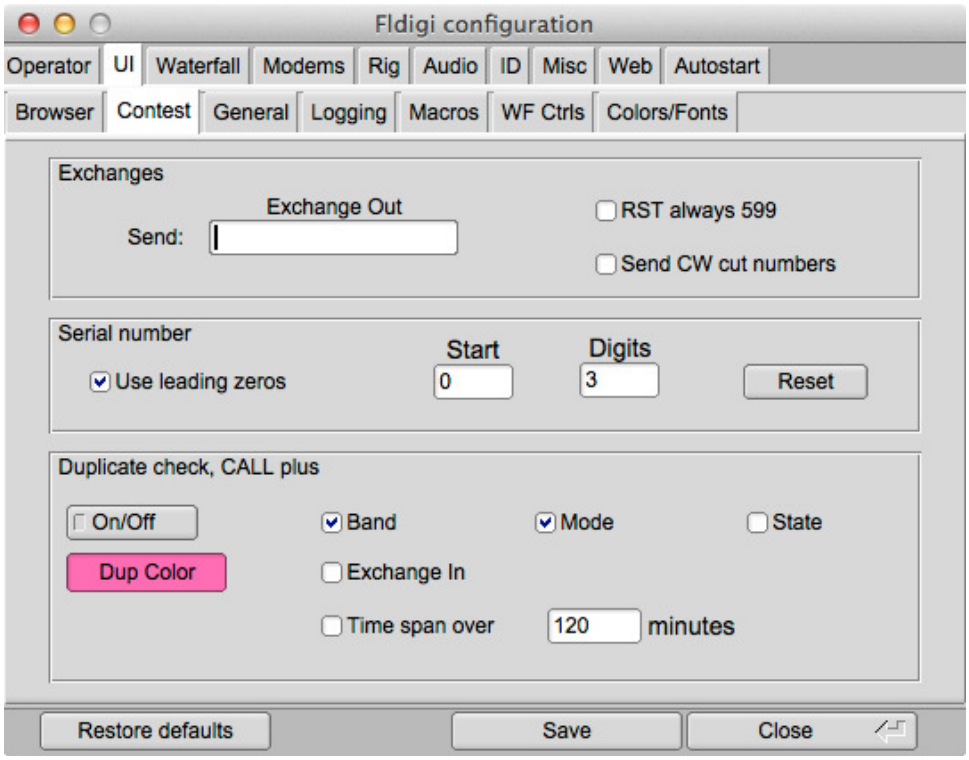


Figure 2.38: UI Contest

The serial number can be set to use leading zeros. You can specify the starting number for the sequence and how many digits are sent, ie: 0024. Pressing Reset will set the starting number to the QSO logging serial out field. See Contest How To for more info.

You can check for duplicates by any combination of the specified named fields. You can also specify that the duplicate had to occur with a given time interval. Some VHF contests allow a duplicate CALL after a given time interval.

Return to Top of Page

Return to Main Page

2.17 User Interface Configuration - General

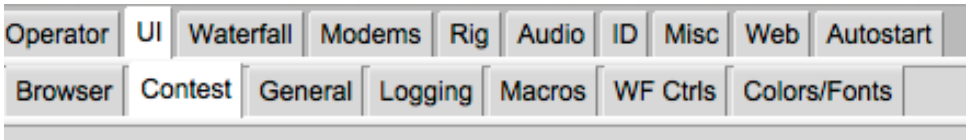


Figure 2.39: UI Tabs

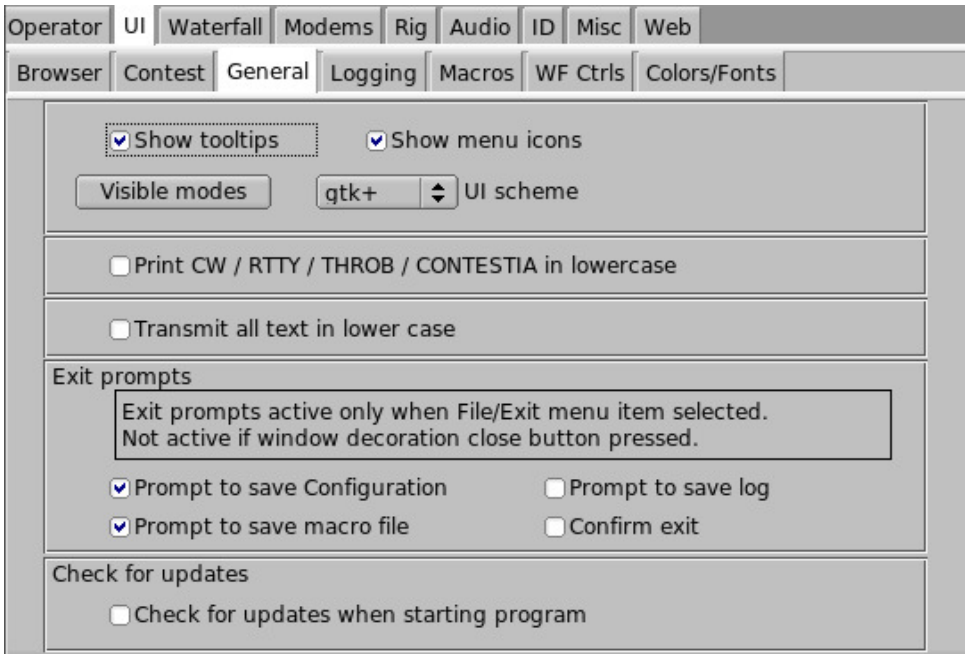


Figure 2.40: General UI Config Panel

Fldigi offers tips on the use of nearly every aspect of its operation. These are particularly useful when you first use the program, but after you are familiar with it's operation they tend to get in the way. You can turn them off by de-selecting "Show tooltips"

Some users prefer to not have icons on the menu system. You can turn them off also.

Fldigi offers three different look and feel based on the parent Fast Light Toolkit graphics interface; "base", "gtk+" and "plastic". These can be combined with command line specifiers for the default background and foreground colors to create a user unique look to fldigi. You will probably discover that the default colors and the gtk+ UI scheme are to be preferred.

Fldigi has internationalization files for French, Italian and Spanish. These control the contents of various menu items and labels. Linux users should build and install fldigi from source to gain access to these. Windows users should select the language of choice from the list, press the "Save" button and then close and restart fldigi. The "UI language" selector is only present on the Windows version of fldigi. The percentage indicates the completeness of the translation.

CW, RTTY (baudot), THROB, and CONTESTIA are modes that only transmit in upper case characters. They also have a very limited set of non-alpha characters. A screen full of UPPERCASE characters can be stressful. Select this option to print all of this text in lower case

Select the configuration items to allow prompting when exiting the program. Note that the prompts are only active if the menu item File/Exit is used.

You can elect to have the program check for updates every time it is started. You can also manually check for updates from the "Help / Check for updates" menu item.

2.17.1 Visible Modes

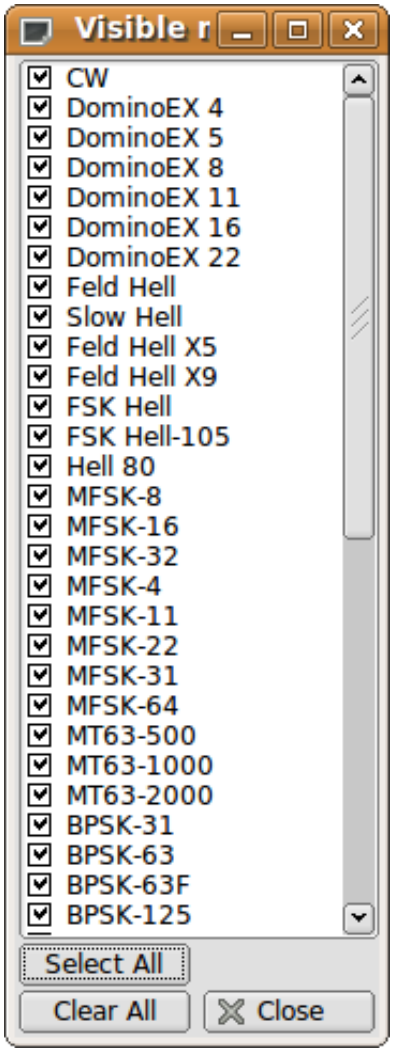


Figure 2.41: Visible Modes

You may not want to use all possible modes when operating fldigi. Press the "Visible modes" button and open the mode selector dialog.

The use of this dialog should be obvious after a few clicks here and there.

2.17.2 Limit Modes

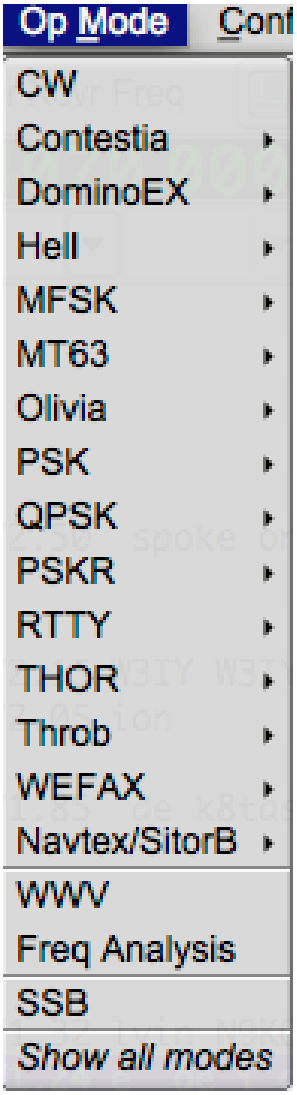


Figure 2.42: Limit Modes

By deselecting all but CW, PSK31, PSK63 and RTTY the Op_Mode menu is uncluttered of all the other modes. The WWV and Freq Analysis modes are always visible. When a subset of the entire mode bank is selected a new menu item is added to the Op_Mode menu, "Show all modes". This is a toggle to restore all modes to the menu. The inverse toggle is "Show fewer modes."

Return to Top of Page

Return to Main Page

2.18 User Interface Configuration - Macros

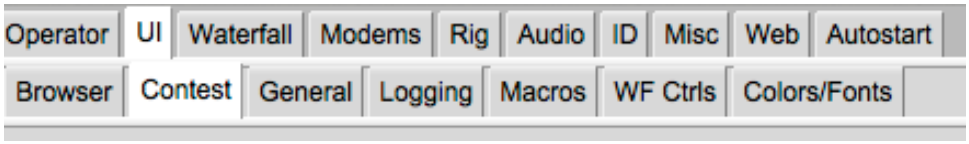


Figure 2.43: UI Tabs

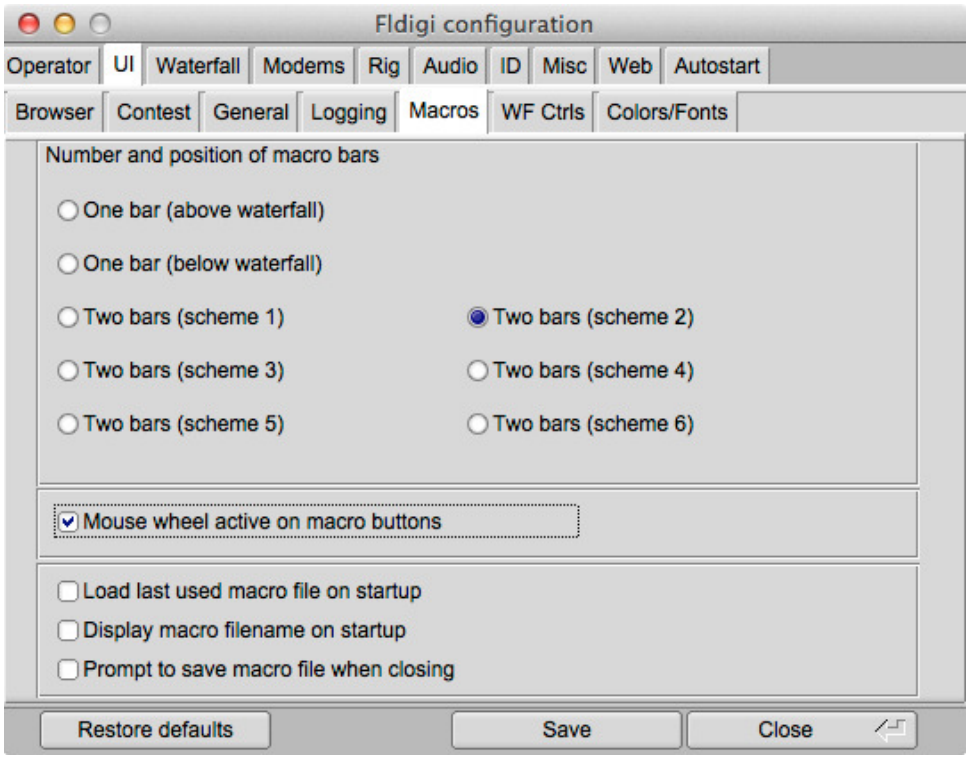


Figure 2.44: UI Configure - Macros

Access to a macro is gained by pressing the associated function key. There are actually 48 separate macros that can be created. With the default configuration you rotate between the sub-sets of 12 using the numbered button to the right of the macro bar, or by selecting a set with the Alt-1, Alt-2, Alt-3 or Alt-4 key combination. (on OS X use the Option-1 etc.)

The default user interface is a single macro bar of 12 buttons located just above the waterfall panel. There are times when you need ready access to more than 12 macro functions. When a 2 row configuration is selected the original row is forced to the Alt-1, or first sub-set of macros, and it's numeric rotate button is disabled. The second or SHIFTED macro button row can be rotate through Alt-2 ... Alt-4 using either it's numeric button or the Alt-#key combination. The fldigi macro bar positions will change immediately so you can see the selection.

With 2 rows shown you obtain access to the primary set with normal Function key press. The secondary set is accessed by a SHIFT-Function-key press.

You edit any macro definition by using a mouse right-click on it's button.

You can also select to use the mouse wheel to rotate through the macro sub-sets. When checked you simply hover the mouse over the macro bar and roll the mouse wheel.

Fldigi manages multiple files that contain macro definitions. You may want to have the last used macro file be the one available the next time you start fldigi. If so, simply enable the "load last used Macro file on startup" check box.

You can also choose to display which macro file was loaded at startup or when a new macro file is loaded. A brief message indicating which file was loaded will be written to the Rx text area if this option is selected.

Return to Top of Page

Return to Main Page

2.19 User Interface Configuration - WF Controls

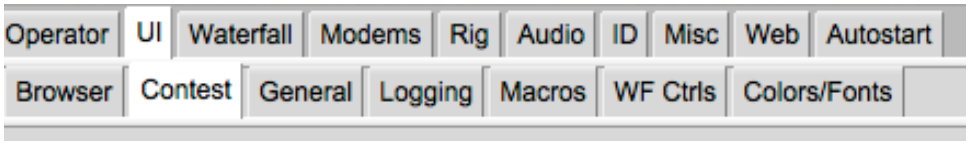


Figure 2.45: UI Tabs

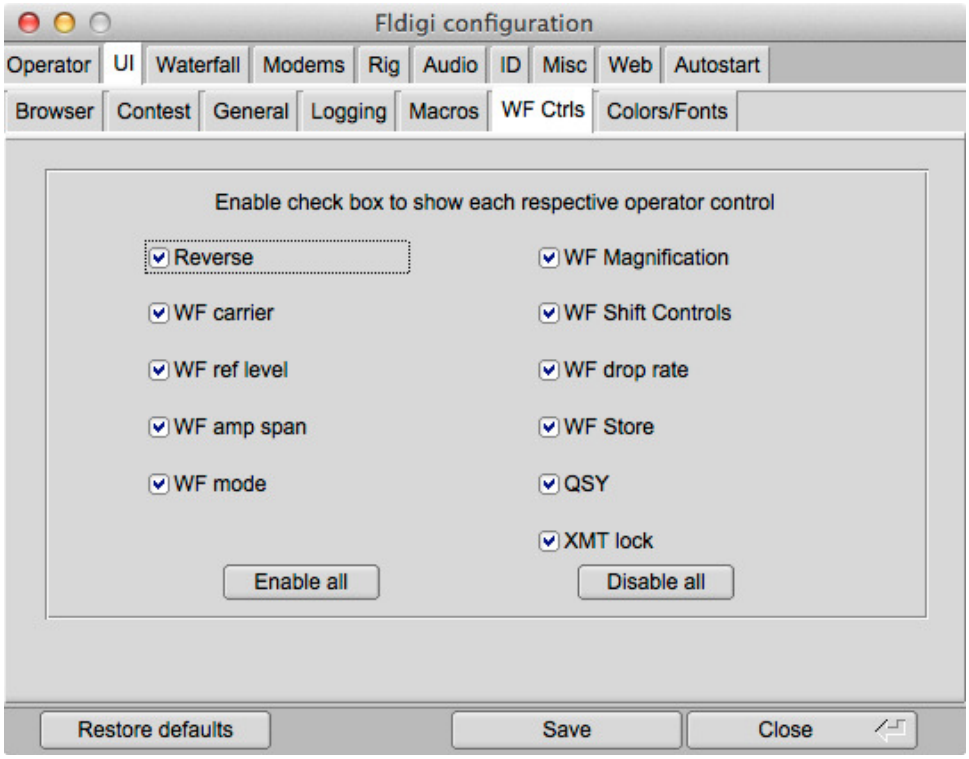


Figure 2.46: UI Configure - WF Controls

You can configure the appearance of fldigi in a variety of ways, including the suppression of unused waterfall controls.

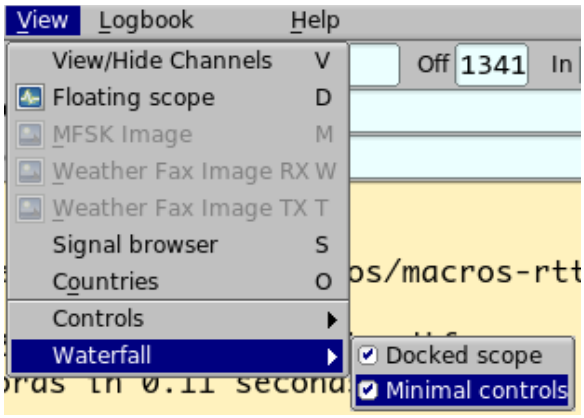


Figure 2.47: UI Configure - WF Controls 2

Before adjusting these settings it is recommended that you enable this menu item. You can then see the effect of enabling and disabling the various selection boxes.

Return to Top of Page

Return to Main Page

2.20 Waterfall Configuration

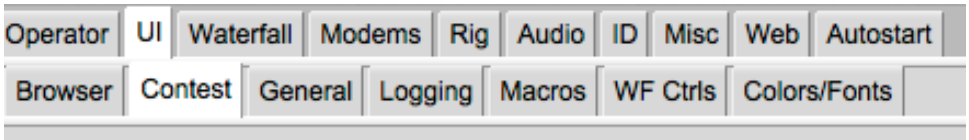


Figure 2.48: UI Tabs

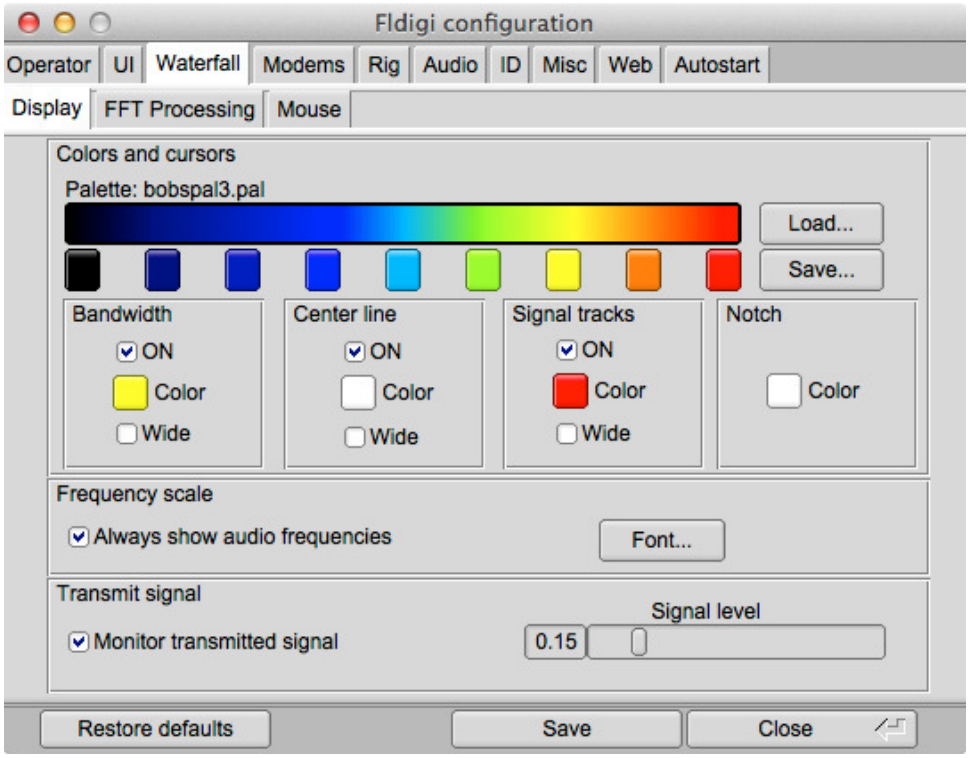


Figure 2.49: UI Waterfall Display

The waterfall palette or color scheme can be altered to suit your personal tastes and visual needs. When fldigi is first started it creates a wide range of pre-built palettes in the $HOME/.fldigi folder. The "Load" button gives you access to those palettes. You may change any palette by clicking on the various color buttons beneath the palette sample. A color picker opens for you to select the color by various means including specifying the RGB values. If you create a palette that suits you better than any of the prebuilt ones you can "Save" the palette.

The waterfall cursor is a set of markers on the frequency scale that are spaced a signal bandwidth apart. You canadd a pair of lines that drop down from those two markers for the full height of the waterfall by selecting Cursor BW. You can add a center line cursor to this pair of BW line by selecting Cursor Center line. You can also add a set of BW lines that straddle the received signal tracking point by selecting Bandwidth tracks. All three of these options are color selectable. Click on the colored button below the check box and a color selection dialog will open.

The frequency scale defaults to RF frequency. You can select to show audio frequencies.

You can monitor the transmitted audio waveform and also set the level of the monitored signal. This IS NOT your final transmitted signal!

Fldigi can set a notch from the waterfall when used with flrig and a transceiver that supports CAT control of a manual notch filter. When the notch is engaged a dotted vertical line is placed on the waterfall at the notch location. You can configure the color of the dotted notch indicator.

2.20.1 FFT Waterfall

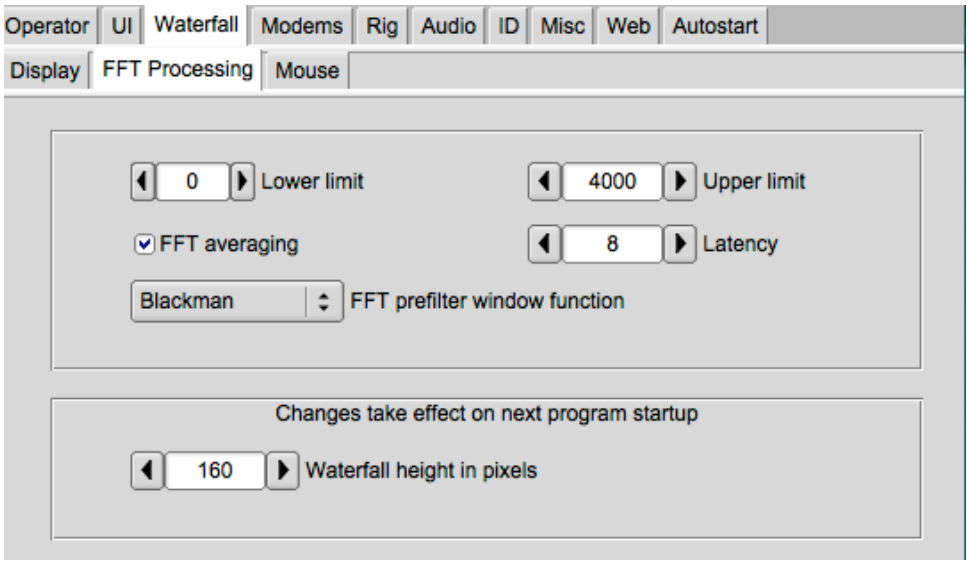


Figure 2.50: UI Waterfall Display

You can extinguish the display of received signals below a particular audio frequency.

Fldigi's waterfall FFT has a bin size of 1 Hz. With an FFT of 8192 and a sampling rate of 8000 it takes almost a second to accumulate enough data to perform the full FFT. A waterfall that dropped at one scan line per second would be hard on the viewer, so fldigi uses a first-in-first-out (FIFO) 8192 byte buffer for the FFT data. 512 byte audio blocks move through the buffer with each successive read of the sound card. The full buffer of 8192 samples is used to compute the FFT. That means that data in the FFT can have a latency of 8 scans. This provides excellent frequency resolution but poor time resolution (the vertical waterfall appearance). The latency control allows you to select the number of 512 byte blocks that are used for the FFT. The default latency is set to 4. You should be able to achieve a reasonable compromise between the time and frequency domain resolutions.

FFT averaging can be used to smooth the waterfall display in the frequency domain.

The FFT Prefilter or window function is used to reduce aliasing in the FFT computation.

The default prefilter for the Fast Fourier Transform associated with the waterfall is *Blackman*. You can try the other windowing filter. Under some conditions you might prefer one of those. The Blackman window has proven best for my setup.

2.20.2 Waterfall Mouse Behavior

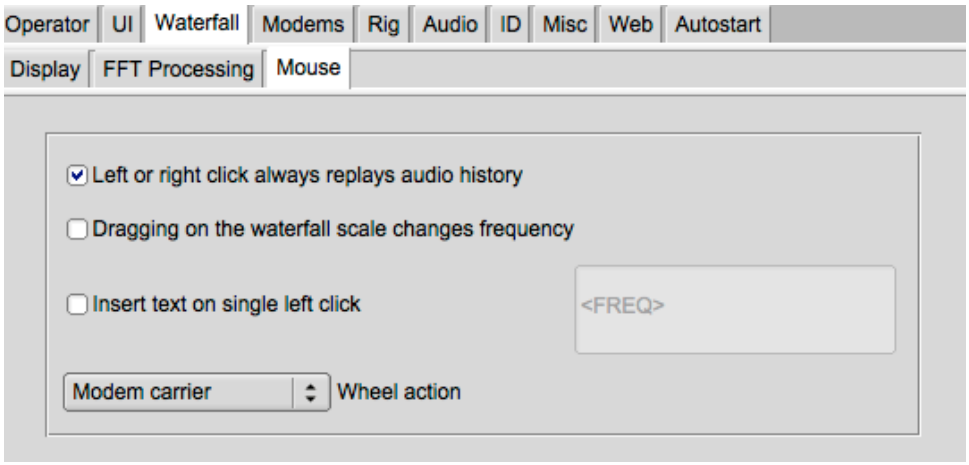


Figure 2.51: UI Waterfall Mouse Behavior

The mouse behavior in the waterfall panel can be controlled to suit your particular operating style. You might want to replay the saved audio history every time you either left click to select or right click to preview a particular signal. You can move the transceiver frequency in increments of 100 Hz by dragging the waterfall scale. You can also choose to insert a line of text into the Rx panel each time you left click a waterfall signal. The text can include expandable macro tags.

2.20.2.1 Waterfall Mouse Behavior Tailoring

The mouse wheel behavior can also be tailored to your liking:

- None - no mouse wheel activity in waterfall panel
- AFC range or BW - adjust the AFC range/BW up/down
- Squelch level - adjust the squelch level up/down
- Signal search - search up / down for next signal in current mode
- Modem carrier - adjust the audio tracking point +/- Hz increments
- Modem - select modem type from a full rotary of available modems
- Scroll - move the waterfall left/right in 100 Hz increments (for 2x, 4x expanded waterfall view)

Return to Top of Page

Return to Main Page

2.21 Working Logs

Working Logs

Fldigi maintains a number of working log files that are found in its default folder. The default folder is easy to find, simply select the menu item "File/Show config" and your OS default files explorer will be opened to that location.

Rx/Tx Capture File

Everytime you start or stop fldigi that event is recorded in a daily log file. The daily log is named as:

flidigYYYYMMDD.log

where YYYYMMDD is the current GMT date. This log will also contain your entire session of Rx and Tx data annotated as to activity and time stamped. Here is a small example of the daily log:

```
--- Logging started at Tue Dec 30 11:37:21 2008 UTC ---

RX (2008-12-30 11:37Z): o ur property. No pwr even for a day is rough.
TX (2008-12-30 11:39Z):
TX (2008-12-30 11:39Z): CQ CQ CQ de W1HKJ W1HKJ W1HKJ
TX (2008-12-30 11:40Z): CQ CQ CQ de W1HKJ W1HKJ W1HKJ pse k

RX (2008-12-30 11:40Z): mG sk

--- Logging stopped at Tue Dec 30 11:48:11 2008 UTC ---
```

This log is appended to with each start and stop. That means that no data is ever overwritten.

Status log

A log of errors, warnings and status reports is written for each session. This file is overwritten each time the program is opened and subsequently closed. Its format is also ASCII text and will contain data such as:

```
Q: main: fldigi 3.04BV log started on Tue Dec 30 05:47:10 2008
W: dxcc_open: Could not read contest country file "/home/dave/.fldigi/cty.dat"
```

This data is identical to that which can be viewed with the event log dialog which is opened using the menu item "Help/Event log"

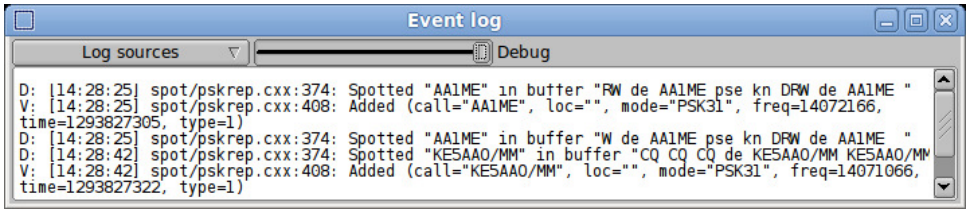


Figure 2.52: Event Log

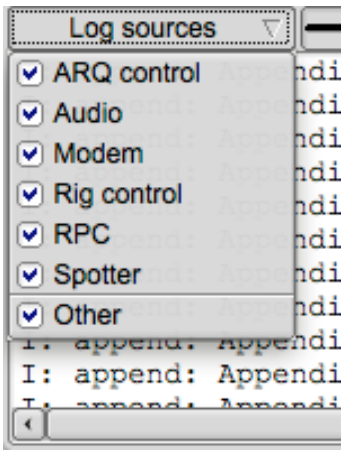
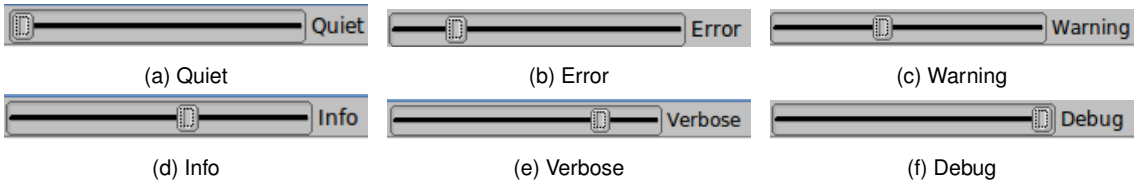


Figure 2.53: Event Log Menu

There are six levels of event logging with increasing depth of reports:



(a) Quiet (b) Error (c) Warning

(d) Info (e) Verbose (f) Debug

Figure 2.54: Event Logging Levels

The default level for logging events is "warning."

The event log show above was captured during a period of psk-reporting. Fldigi was set up to monitor and report all detected signals that satisfied the requirements of the psk reporter web site. The "spotted" signals were then automatically sent to the web site. A complete report of the recorded events was obtained by a right click in the text pane. Select-all and Save as was chosen.

At the Debug level you will probably see more events than you need. You can select which events to suppress using the "Log sources" menu button. It defaults to all enabled.

Return to Top of Page

Return to Main Page

2.22 Contestia Configuration

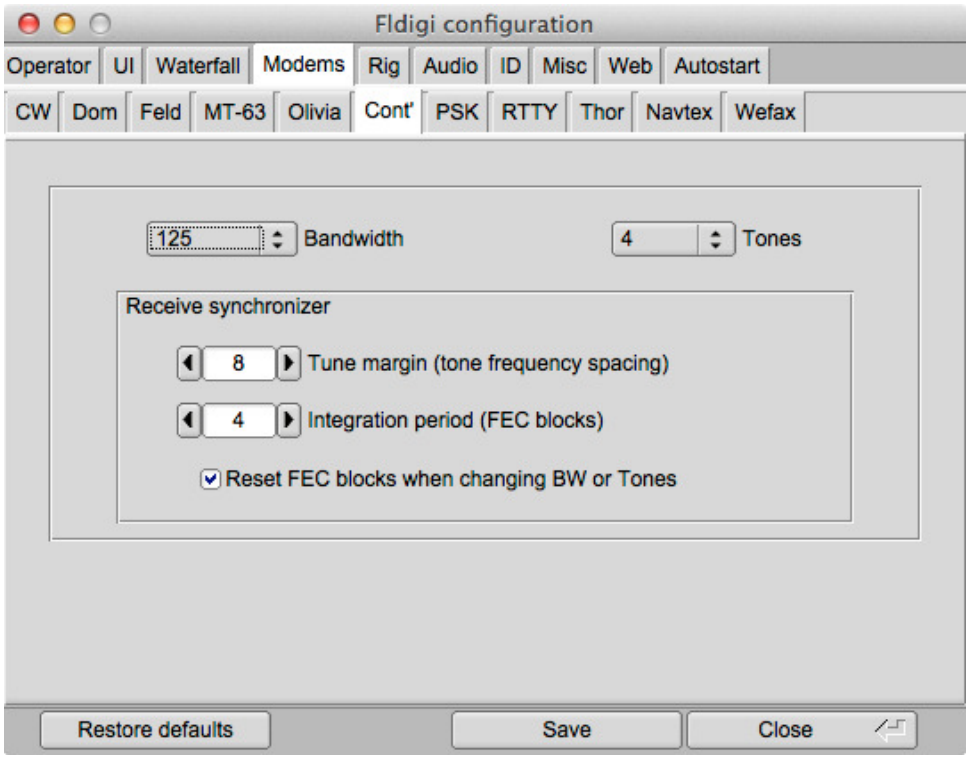


Figure 2.55: Contestia Configuration

Configuration of Contestia is similar to Olivia as Contestia is a derivative of Oliva.

See Contestia for additional information.

Return to Top of Page

Return to Main Page

2.23 CW Configuration

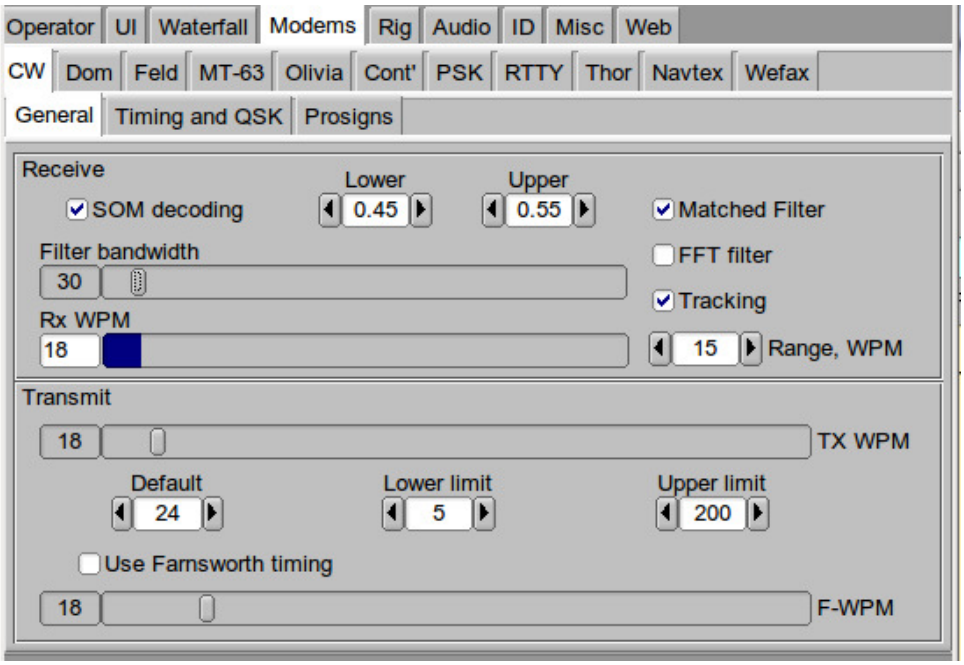


Figure 2.56: General CW Configuration

Fldigi can send and receive morse code from 5 wpm to 200 wpm. The operating controls for CW are found on the Config/CW tab. You can open that tab by selecting the "Configure/Modems" menu item and the clicking on the Modems/CW tab. You can also open up the CW tab by first selecting CW as the operating mode and then clicking on the left-most item "CW" on the status bar at the bottom of the fldigi main window. During operation the Rx and Tx WPM settings are annunciated on the status bar in the two boxes next to the mode indicator.

The CW signals are converted to a baseband signal. It is the digital equivalent of tuning an analog USB transceiver so that the carrier is exactly at the CW carrier frequency. The CW decoder can use one of two different DSP filters. The Fast Fourier Transform (FFT) filter is implemented with a sin(x)/x impulse response. This is a very steep sided low pass filter. Unchecking the FFT filter causes the decoder to use a Finite Impulse Response (FIR) filter with a cutoff slope that is not as steep. The FFT filter is optimum when receiving CW in a white noise environment. The FIR filter may give better response with impulse noise. Selected a "Matched" filter for either the FFT or FIR implementation optimizes the filter bandwidth for white noise suppression. A lot of impuse noise (static) can cause either filter to ring and increasing the filter bandwidth might improve detection in that electrical environment.

Fldigi can track the incoming signal. Enable Rx WPM tracking by enabling the check box "Enable Tx Trkg". The tracking range (+/- Hz around the TxWPM setting) can be set using the "Rx Trkg Rng" control. When tracking is enabled the tracking filter is reset every time the transmit WPM is adjusted.

CW detection is basically an amplitude demodulator. You can set the threshold for detecting when the signal transitions from off-to-on and on-to-off. These signal levels are relative to the average signal level. The on-to-off is the "Lower" value and the off-to-on the "Upper." This implementation provides a hysteresis detector. Early fldigi CW decoders used this scheme but the detection levels were not adjustable.

The RxWPM control is an indicator and is not used for setting the operation of the CW decoder.

"SOM decoding" provides a fuzzy logic implementation to match the RX stream detected on-off sequence to a "best fit" character. It can increase the probability of correctly identifying the text character under very noisy conditions.

The TxWPM sliding controller is used to set the transmit WPM. To make the setting easier two additional controls

are provided. "Lower" sets the lower limit of the slider and "Upper" sets the upper limit of the slider. The resolution of the TxWPM slider is 1 WPM. The Lower/Upper controls are in in 5 WPM increments.

The transmit encoder settings for WPM can also be adjusted with three hot keys:

- Numeric keypad "+" increases the TxWPM by 1
- Numeric keypad "-" decreases the TxWPM by 1
- Numeric keypad "∗" toggles between the selected TxWPM and a default WPM

The "Default" control on the CW tab sets that default value. As shown above the TxWPM is 30 and the default is 18. If during a QSO you needed to slow down to give the other op a better chance to copy what you are sending, just hit the "∗" on the numeric keypad and the CW code will immediately switch to sending CW at the set default value (18 wpm in this example). Press the "∗" again to return to back to the CW speed that you were previously using.

2.23.1 Timing

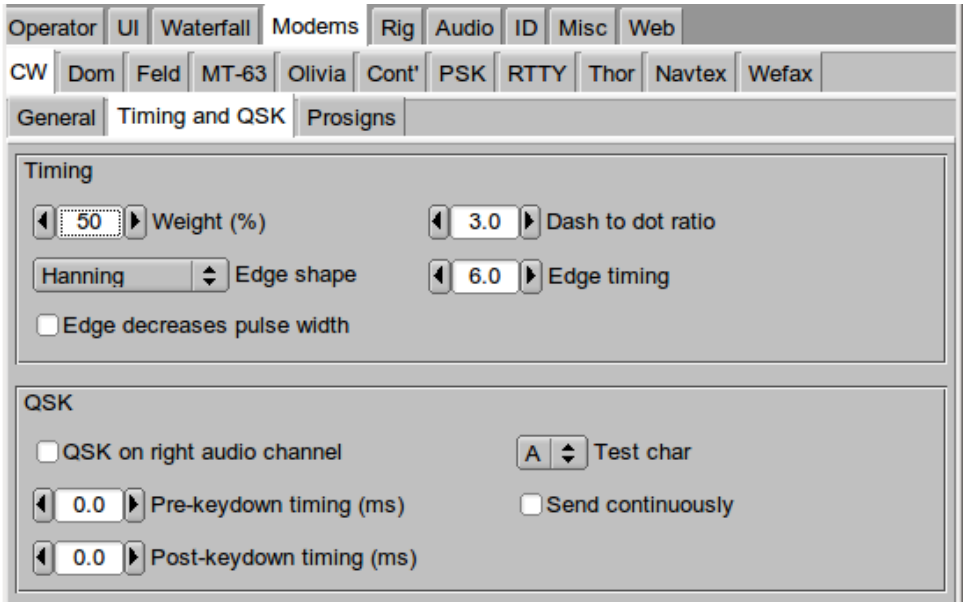


Figure 2.57: Timing

Fldigi generates CW by inserting a keyed tone at the current waterfall audio frequency. The transceiver should be operated in either USB (preferred) or LSB mode. The CW signal is completely generated in the software so it is possible to control many aspects of the CW signal. The actual transmitted signal will be at the USB carrier + the audio frequency, or the LSB carrier - the audio frequency. If fldigi is tracking and receiving a CW signal on the waterfall your transmitted signal will be exactly on the frequency of the other operator. The CW generated this way has a nearly ideal attack and decay time, controlled by the software modem. But ... a caveat ... your transmitter must never be overdriven and it should have excellent opposite sideband suppression. Overdriving the transmitter can cause multiple audio signals within the SSB passband, and cause unwanted interference to other ops. The same is true for a poorly designed or adjusted transmitter with bad sideband suppression. I recommend having a trusted and knowledgable operator assist you when first trying A2 CW. Have them carefully look for evidence of your signal above and below your primary signal (by at least +/- 3 Khz). If there is no evidence of extra signals then your are set to go. If there is you might want to have the transceiver adusted for sideband suppression, or check to be sure you are not over driving the audio.

- Wt % control sets the weight of the CW. Normal CW is at 50% weight, ie: a dot is equal to the interval between dots or between code elements. It has a range of 20 to 80 percent.

- Dash/Dot controls the relative weight between a dash and a dot. The standard for CW is 3 to 1. The dash is 3 times the length of a dot. Some operators prefer the sound of either a heavier or lighter sounding CW. This control can be adjusted from 2.5 to 4.0 in 0.1 increments.

- Edge shape provides two leading/trailing edge shapes (1) Hanning, or raised cosine, and (2) Blackman a modified raised cosine with a steeper attack and decay. Both of these edge shapes give a more narrow bandwidth CW signal than the traditional exponential waveform. They are very easy to listen to even at speeds exceeding 100 wpm.

- The Edge control sets the rise and fall times of the CW waveform. It can be set anywhere from 0.0 to 15.0 milliseconds in 0.1 millisecond increments. DO NOT operate A2 CW with the control set below 4 msec. This is the control that sets the effective bandwidth and sound of your CW. If the edge is too steep you will have a clicky signal and be the bane of the CW bands. The purpose of being able to set the edge to 0.0 or a very quick rise/fall time is explained below. A good setting for nice sounding CW at 40 WPM and below is 4 to 6 milliseconds.

- Edge decreases pulse width, when checked will give a slightly narrower dot length as the edge timing is increased. This is useful when operating QSK and listening between the character elements.

This is what the A2 signal should look like with various settings of weight, Dash/Dot and Edge. The audio frequency is 400 Hz and the TxWPM is 100 WPM.

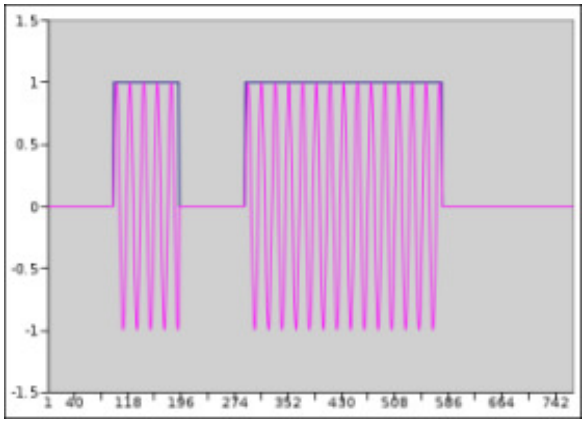


Figure 2.58: Dash/Dot = 3.0, Edge = 0.0

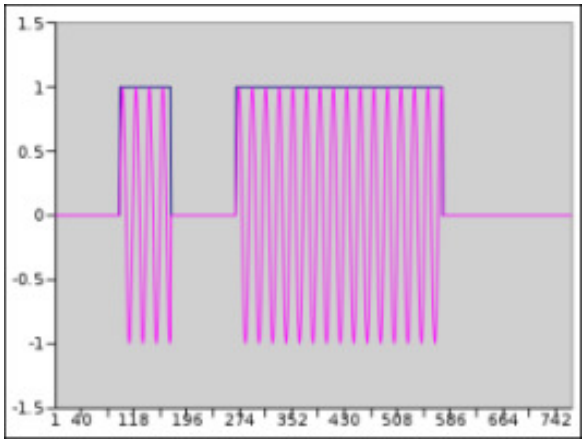


Figure 2.59: Dash/Dot = 4.0, Edge = 0.0

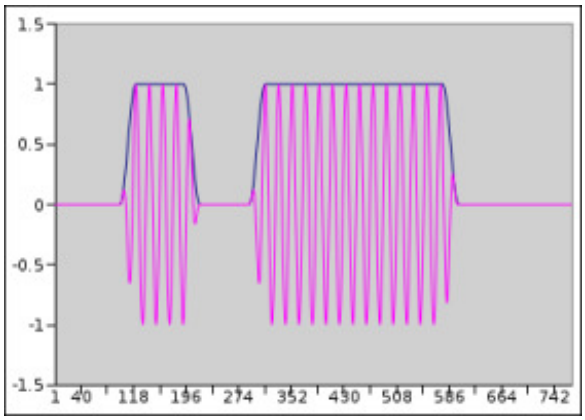


Figure 2.60: Dash/Dot = 3.0, Edge = 3 msec

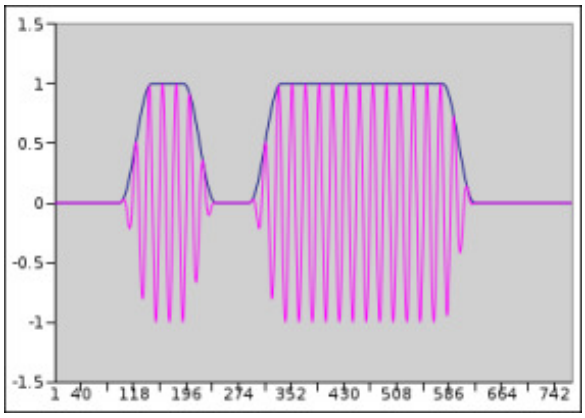


Figure 2.61: Dash/Dot = 3.0, Edge = 6 msec

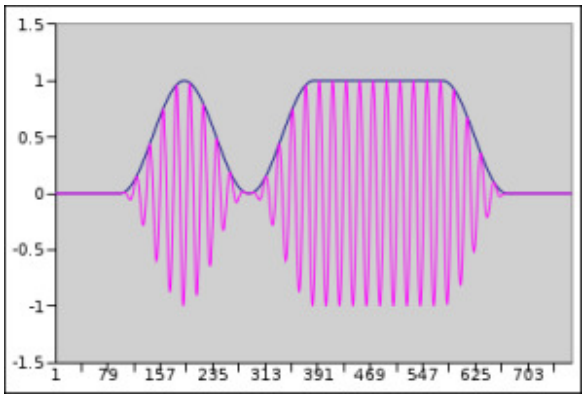


Figure 2.62: Dash/Dot = 3.0, Edge = 12 msec

Changing the weight, dash/dot or edge of the waveform does not change the WPM at which the code is generated. When a conflict occurs between the various settings WPM takes first priority, and Edge second. In the above examples, the Edge setting could not exceed 12 msec even if the control were set higher than 12.0. The figures were generated by capturing the output data being sent to the sound card and then formatting it using Gnumeric. An oscilloscope photo of the signal is virtually identical.

The setting for inter-character and inter-word spacings are fixed at 3 and 7 respectively. The 3 is achieved by sending a silent period of 1 dot (element) length at the beginning of each character and 2 at the end of each

character (shown in the figures). This silent period is sufficient for most transceivers to respond to the PTT signal which occurs at the beginning of the transmission so that the first dit or dash is not lost in transmission.QRQ (high speed CW operation)

You may wonder why fldigi can go as high as 200 WPM. It's hard to believe but there are CW operators who can decode 100+ WPM in their head. These operators also usually operate QSK (full breakin). A2 CW and PTT operation and QRQ/QSK are not a natural mix. But fldigi can be used for this type of operation if an external keyer is used. For that purpose the A2 Tx output from fldigi is full wave rectified and detected to create a keyline control. The outboard conversion from A2 to keyline requires a nearly square wave pulse output of audio at the CW keying rate. Setting the Edge control to 0.0 and then the audio frequency to about 1000 Hz provides the needed signal to effect this type of keyline control.

If you are operating QSK with a separate receiver / transmitter you can very quickly stop your transmit signal with the TAB key. In the CW mode only the TAB key causes the program to skip over the remaining text in the transmit text buffer. The text that is skipped will be color coded blue. The program remains in the transmit mode (PTT enabled), but since the buffer is now empty no A2 CW signal is generated. Code transmission will then restart with the very next keyboard closure of a valid CW character. The Escape and Pause/Break keys still can be used to respectively abort and pause transmission.

2.23.2 QSK

You might ask why fldigi doesn't simply provide a keyline output on one of the parallel port pins or on RTS or DTR via a comm port. The answer is quite simple. Linux is a multi-tasking operating system and the interaction between the OS and the application causes the timing to be adversely effected. The driver implementation of the audio sub system must be responsive and so the OS gives that sub system a very high priority in its multi-tasking structure.

Many QSK operators use high speed diode antenna switching between receiver and antenna. fldigi generates a signal that can be used for that purpose. The left audio channel is always theAFCW signal. When selected the right audio channel can be configured to generate a square wave signal that begins earlier and ends later than each of the CW elements. The square wave signal can be rectified and filtered to provide the diode switching signal for the Rx/Tx antenna switching.

The right audio channel QSK signal is selected by checking the box and then adjusting the pre and post timing in millisecond increments. Additional information and a schematic diagram of a QSK keying circuit is described in CW Keying. Setting up a QSK device can be quite difficult. Fldigi helps to ease the adjustment by generating a continuous series of characters. This allows a dual trace scope to be properly synched while making the adjustments to both the software and the associated QSK hardware. You enable continuous characters by selecting the checkbox, and then enabling the T/R button for transmit. The repeated character can be change on the fly with the pick control. It can be one of either E, I, S, T, M, O or V.

2.23.3 PROSIGNS

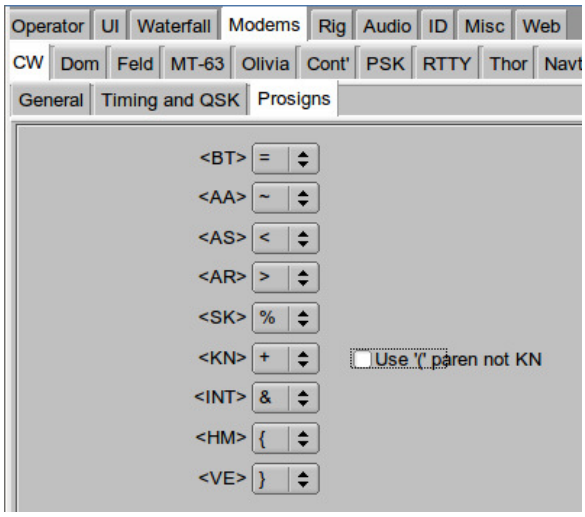


Figure 2.63: CW Prosigns

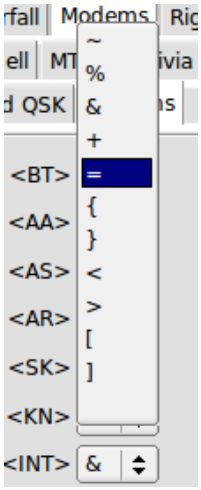


Figure 2.64: Prosign Popup

You can assign keyboard characters to be used for Morse prosigns. The available characters are: ~ % & + = { } < > [ ]

The default assignments are shown above. You can also elect to send and receive the KN prosign as an open parenthesis '('. This is commonly used on MARS CW operations.

See Operating CW for additional information.

Return to Top of Page

Return to Main Page

2.24 DominoEX Configuration

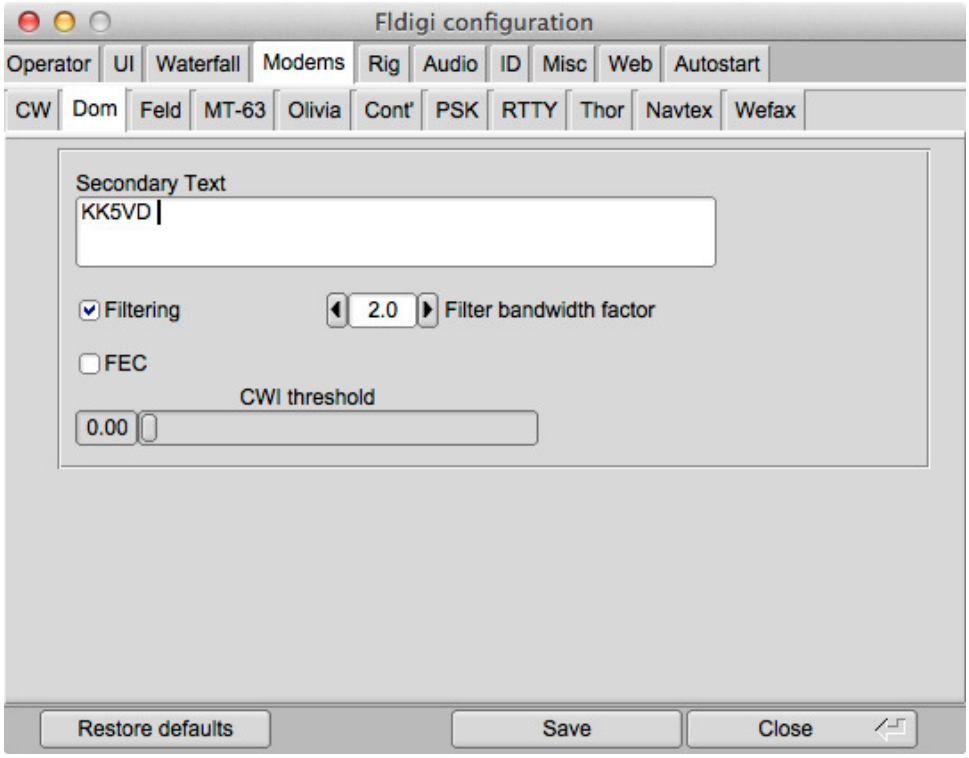


Figure 2.65: DominoEX Configuration

Enter the secondary text. This text will be sent during periods when your keyboard is inactive (between letters for slow typists). The default for this text will be your callsign when you have entered that in theOperator configuration tab.

Set the BW factor for the decoding prefilter. 2.0 should be adequate unless you are experiencing nearby continuous wave interference (CWI). You can enable and disable the prefilter with the checkbox. Please note that the filter requires additional cpu cycles. Older and slower cpu models might give better decoding with the filter disabled.

Fldigi can send and receive FEC in accordance with the DomEX-FEC specification for MultiPsk. This type of FEC is achieved by some loss of non printing characters in the primary character set. It is therefore not usable as an FEC mode for ARQ (automatic repeat request) transmissions.

The DominoEX decoder can detect the presence of CWI within the passband set by the BW factor. Increasing the CWI threshold increases the sensitivity to such interference. When the interference is detected the associated data is culled using a technique called puncturing.

DominoEX operations are described in DominoEX .

Return to Top of Page

Return to Main Page

2.25 Feld Hell Configuration

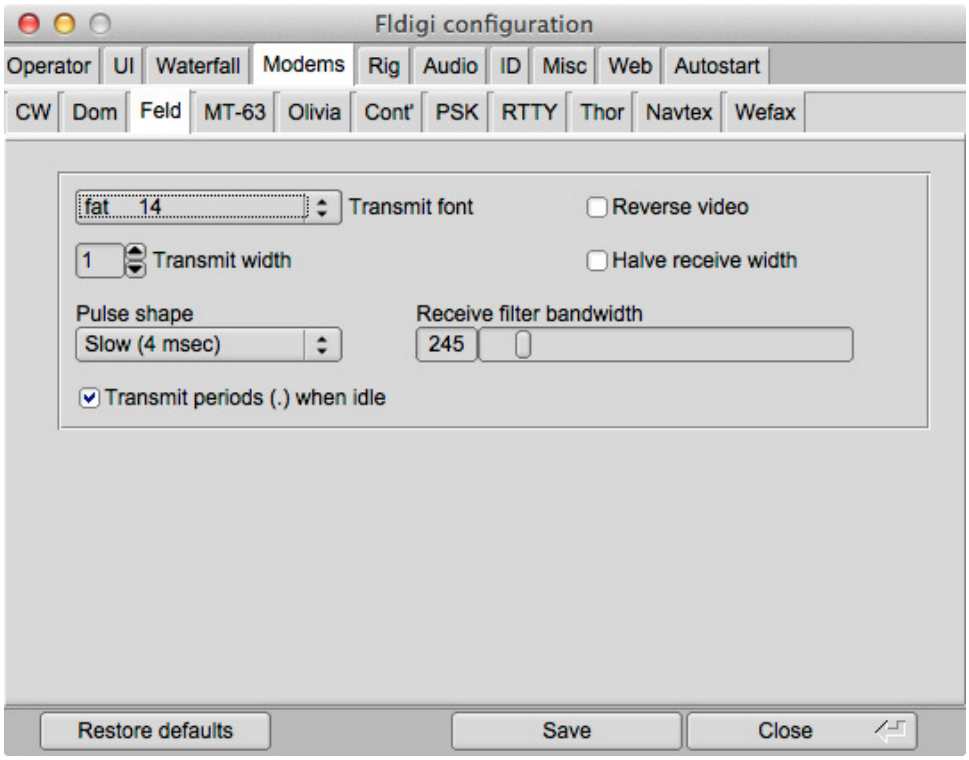


Figure 2.66: Feld Hell Config Panel

The Hellschreiber modes all use a video display that is basically a character-by-character facsimile. The shape of the characters is determined at the transmitting station. You can select from 15 different fonts, all of which have been designed for Feld Hell use. In normal use each dot of the character font is transmitted twice. You can increase the effective video s/n by transmitting the dots 2 or 3 times the normal. That is controlled by the Transmit width.

FeldHell is a pulse amplitude modulated signal. The shape of the pulse is a raised cosine. This helps to control the bandwidth of the transmitted signal. It is customary to use a 4 millisecond risetime for the raised cosine, especially on HF. You can change that to 2 milliseconds. The video edges will be sharper, but the bandwidth twice as large. You might find 2 msec a better choice for VHF and above.

Fldigi provides three controls for the receive function. The video is normally black on white. You can change that to white on black. You can also compress the horizontal scan rate of the video. You might find this effective in displaying received fonts that are broad (or if the other end is using a multiple dot transmit width).

As you change Hellschreiber modes the optimum filter bandwidth will be set. You can change that and might find a narrower filter effective if you are experiencing CWI interference. FeldHell is susceptible to CWI. If you narrow the filter the received video will become blurred.

Return to Top of Page

Return to Main Page

2.26 ID Configuration

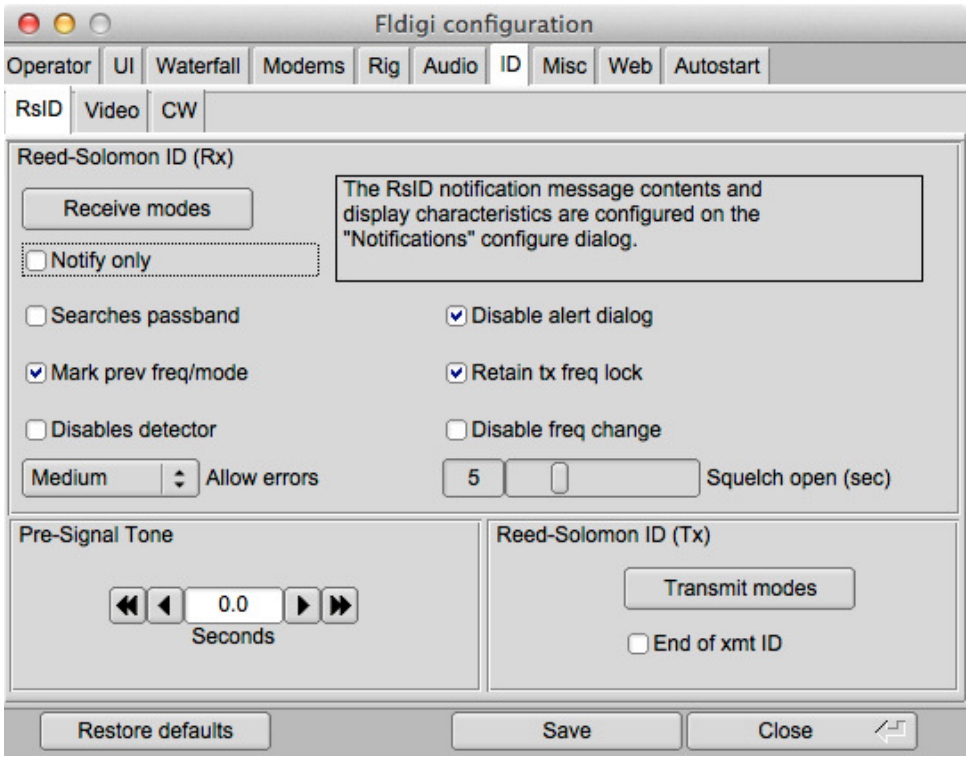


Figure 2.67: RSID Config Panel

Fldigi offers several ways to identify the operator or mode that is being used. This is particularly useful when using a hard to recognize mode such as Thor, Olivia or MT63.

2.26.1 Reed Solomon Identifier

RSID, **R**eed **S**olomon **ID**entifier is a special transmission designed by Patrick Lindecker, F6CTE, for the modem program MultiPsk. It has been adapted to other modem programs. Fldigi's implementation is compatible with the MultiPsk RSID, but provides a slight variation. You can transmit RSID at both the beginning and end of a transmission. The detection of RSID normally only occurs in the near vicinity of the current waterfall tracking point. This cuts down on extraneous RSID detections when the band is crowded and several RSID signals might be present. If you want fldigi to search the entire waterfall for RSID signals you can do so by enabling the "Detector searches entire passband". You start the search for a signal based on RSID by using the main panel switch. The RSID detector is a separate decoder that operates in parallel with all other modem decoders. If you select the "Mark previous frequency and mode" a restore link will be inserted into the RX text upon detecting an RSID signal. Clicking on this link restores the previous frequency and mode of operation. You elect to disable the RSID upon first detection. You also have the option of just receiving notification when an RSID signal is detected. The notification occurs with a pop-up message box.

You can select which modes will include the transmitted RS identifier, and which modes will react to a received and decoded RS identifier.

The mode to identifier relationships are selected by pressing the associated "modes" button.

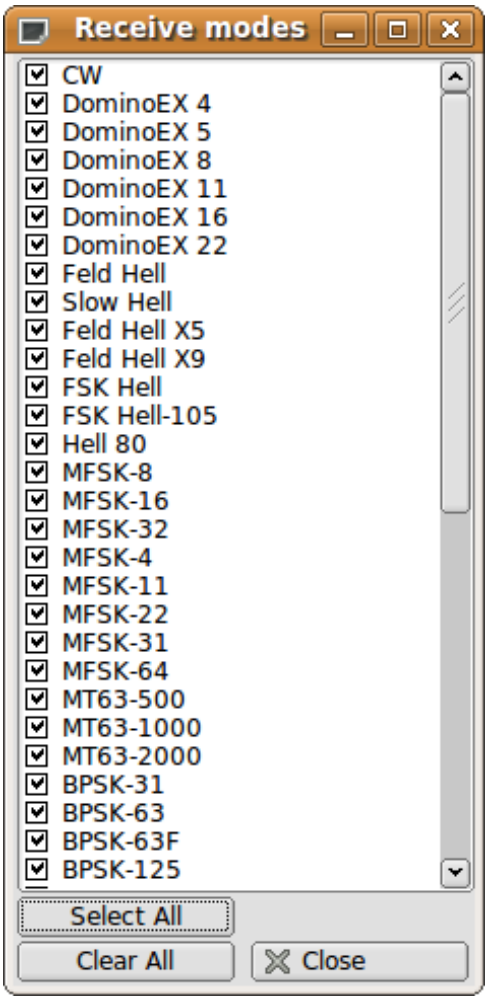


Figure 2.68: Receive Modes

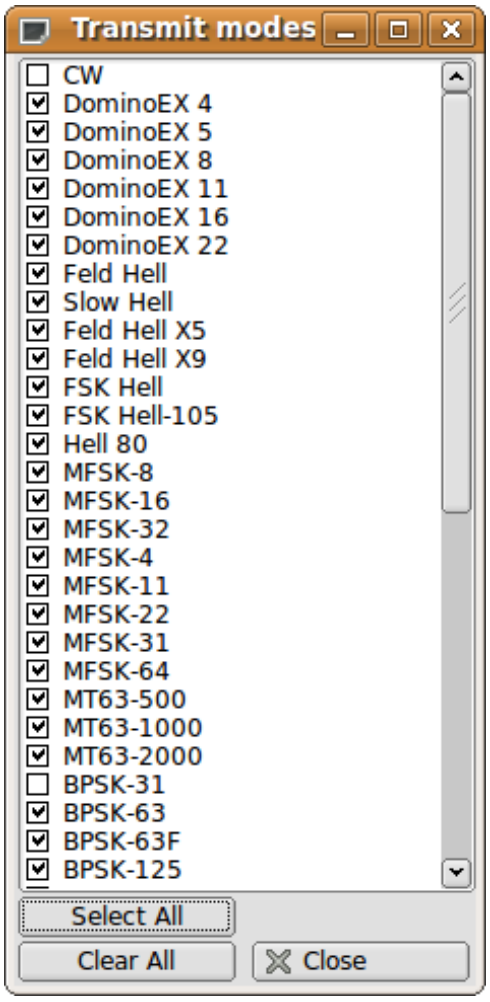


Figure 2.69: Transmit Modes

2.26.2 Video Text

Transmitted video text will appear as a sequence of characters on the waterfall.The text can be a brief mode identifier or some user specified text. You can use a small font that always appears as a 2 character wide sequence or a larger font that can be 1 to 4 characters wide. You should be aware that the video signal is a constant energy signal and the content will be spread across multiple characters. The highest s/n at the receiving end will be for 1 character wide video. Small font at 2 character width is next in s/n performance followed by 2 character large font etc. You can select which modes will include the video text preamble. You can limit the horizontal (frequency width) of the video signal in one of several inclusive ways.

- Number of characters per row of text
- Constrain to be less than or equal to 500 Hz
- Constrain to be within the bandwidth limits of the mode in use

Fldigi uses abbreviated acronyms for the mode and it's characteristics when you are transmitting the mode ID using a video text. Here are two examples, one in small and the other in large font.

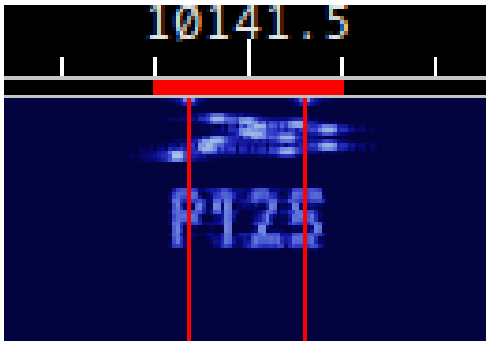


Figure 2.70: Small Font

Figure 2.71: Large Font

Notice that Olivia 16-500 is abbreviated to OL-16/500 and that the number of characters is limited to 8 per row. You might want to use the large characters by default if you routinely have QSO's with operators using older digital mode programs or one whose waterfall visual is not on a par with fldigi's.

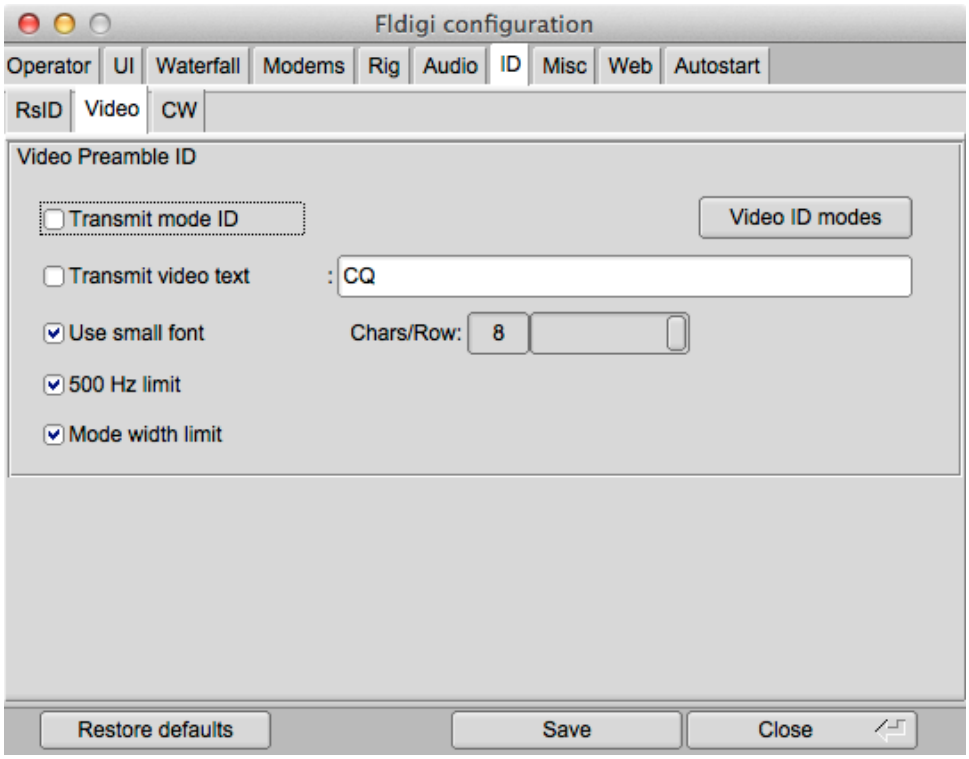


Figure 2.72: Video ID Config Panel

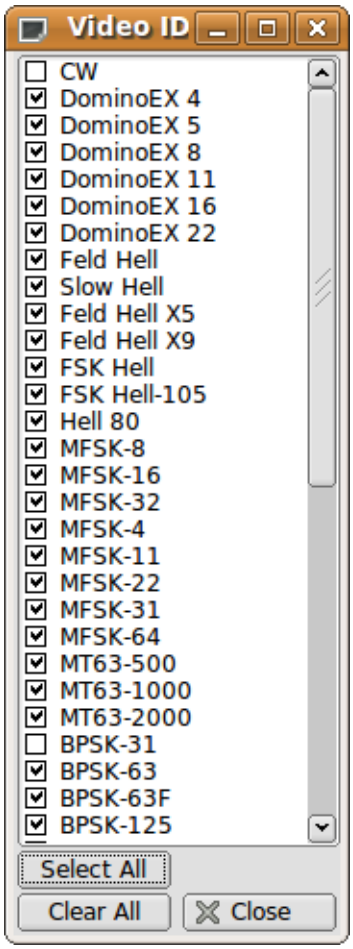


Figure 2.73: Video ID Modes

2.26.3 CW Postamble

You can transmit your callsign in CW as a postamble to all modes except of CW (a bit redundant to do that). You can select which modes will include the CW postamble.

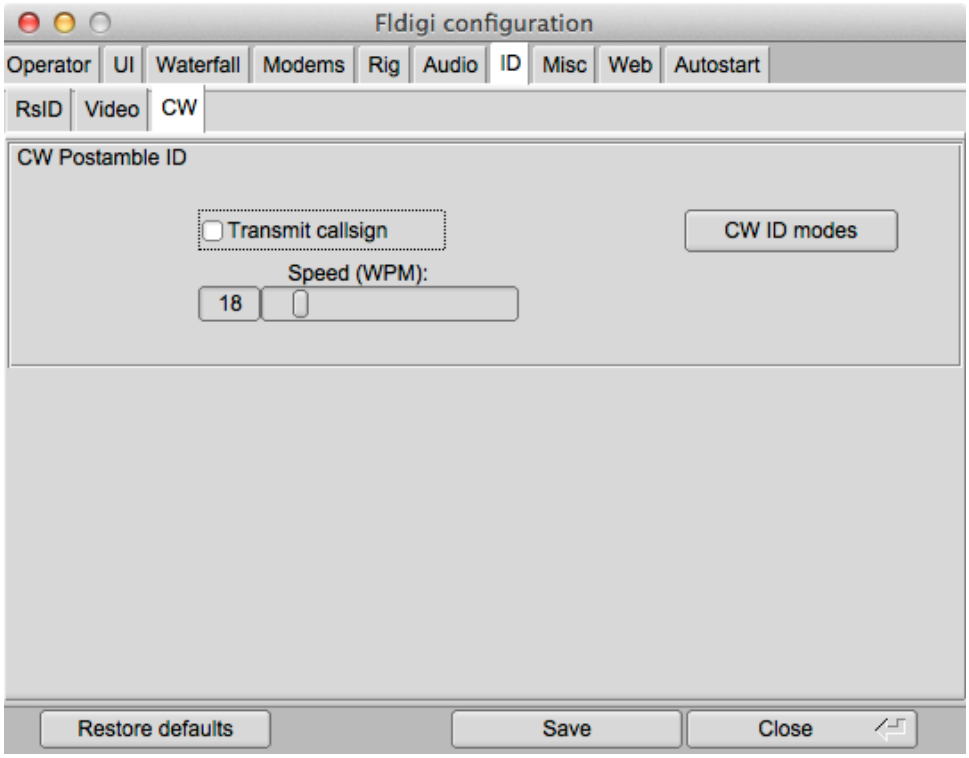


Figure 2.74: CW Postamble ID Config Panel

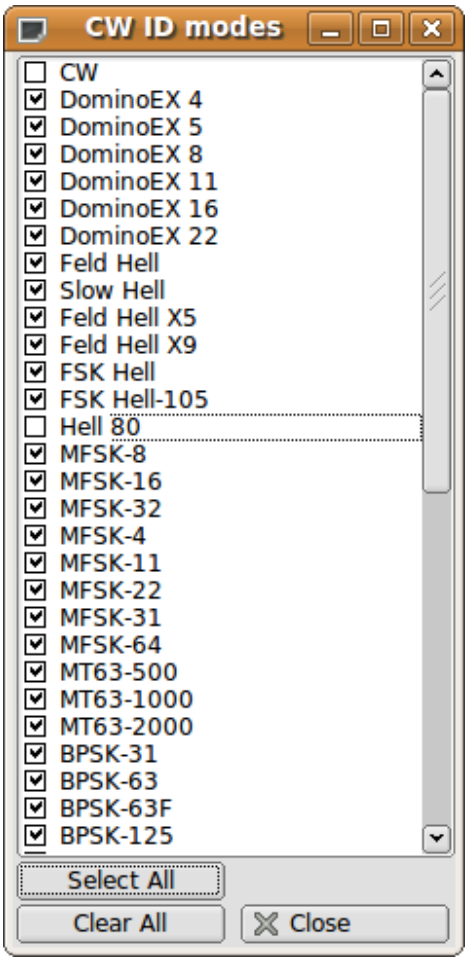


Figure 2.75: CW ID Modes

Return to Top of Page

Return to Main Page

2.27 Miscellaneous Configuration

2.27.1 Sweet spot

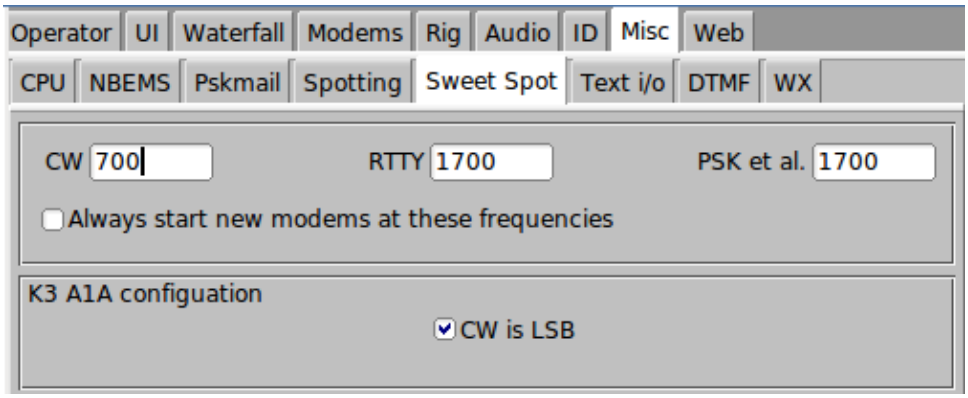


Figure 2.76: Misc Sweet Spot Config

The sweet spot is the audio frequency at which your transceiver provides the best filtering for a particular signal type. You can specify the value of the sweet spot for CW, RTTY and all others. You can also elect to have the audio cursor placed at the sweet spot when changing modes. The sweet spot is used for the QSY function.

The K3 A1A lower sideband can be selected for any transceiver that only provides LSB in the CW mode.

2.27.2 Callsign spotting

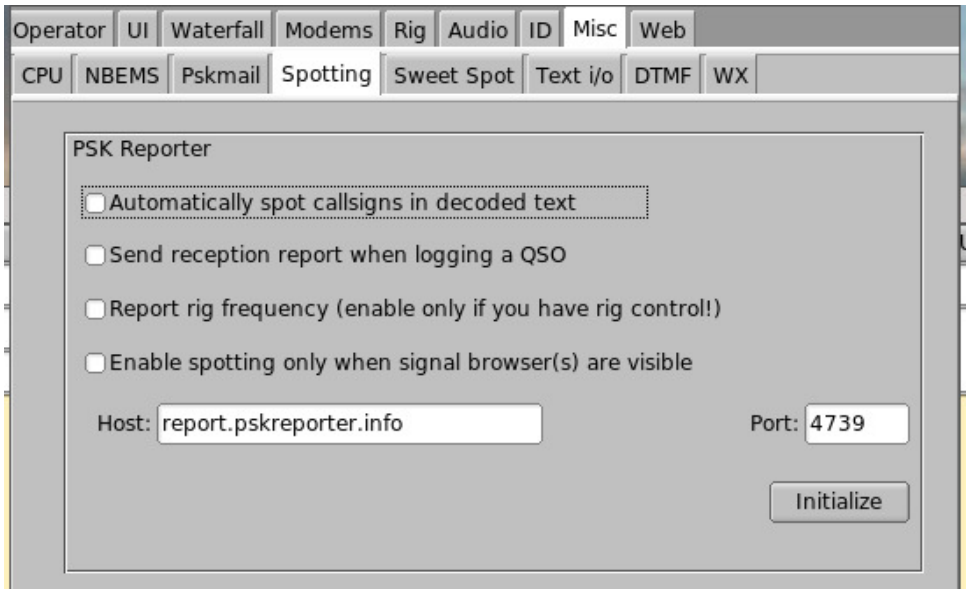


Figure 2.77: Callsign spotting

Fldigi allows you to automatically participate in a spotting network maintained by Philip Gladstone. You can see what the web based reporter looks like by visiting this web site: http://pskreporter.info/pskmap?W1HKJ or by simply selecting the menu item "Help / Reception reports...".

Fldigi will continuously scan for spotted callsigns in the decoded text and send reports in the background if you check the "Automat..." option.

Reports will also (or only) be sent when you log the QSO into the logbook.

If you have rig control enabled the reported rig frequency will also be sent to the spotting network. Do not change the Host and Port numbers unless these are changed by Philip.

You need to press the Initialize to begin reporting spot information. You will receive a warning message if you did not enter your antenna information on the Operator tab.

If your CPU performance is marginal you can disable spotting when no signal browser is visible.

2.27.3 CPU performance

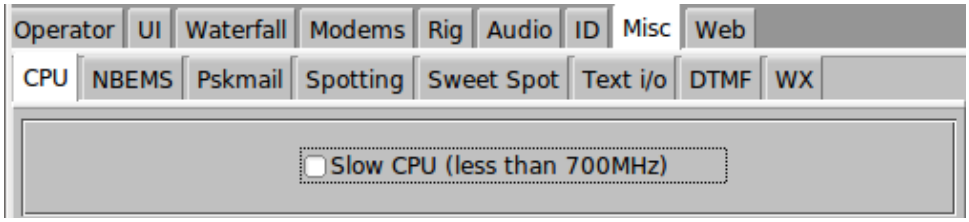


Figure 2.78: Misc CPU

When fldigi is executed for the first time it does some tests to determine the performance factor for your central processor unit. If it determines that the cpu is below a critical speed it tries to compensate by modifying some of its timing and algorithms. If you are using a "slow" cpu the "Slow cpu" check box will be enabled. You can also manually check this box if you find that fldigi is not performing well on some of the more esoteric modes such as PSK250, MFSK32, etc.

2.27.4 Text i/o

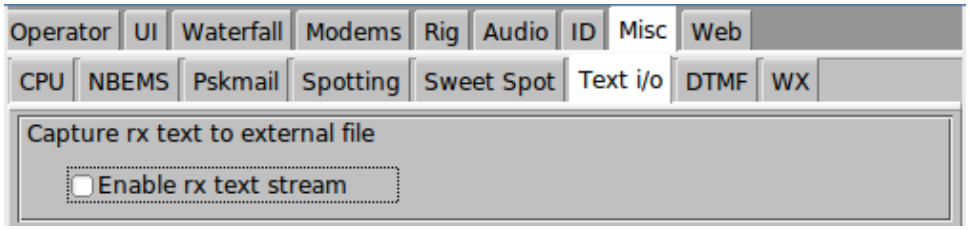


Figure 2.79: Misc Text I/O

Fldigi can perform automatic capture of the Rx text stream. The simplest is to simply capture all incoming text to a file. Select this from the lower of the two frames. The Rx file is named "textout.txt" and is written to the directory as shown above. The file can be used to review an execution session, or it can be accessed by an external program. For example it could be parsed to provide a text to speech conversion.

2.27.5 NBEMS (flmsg / flwrap) interface

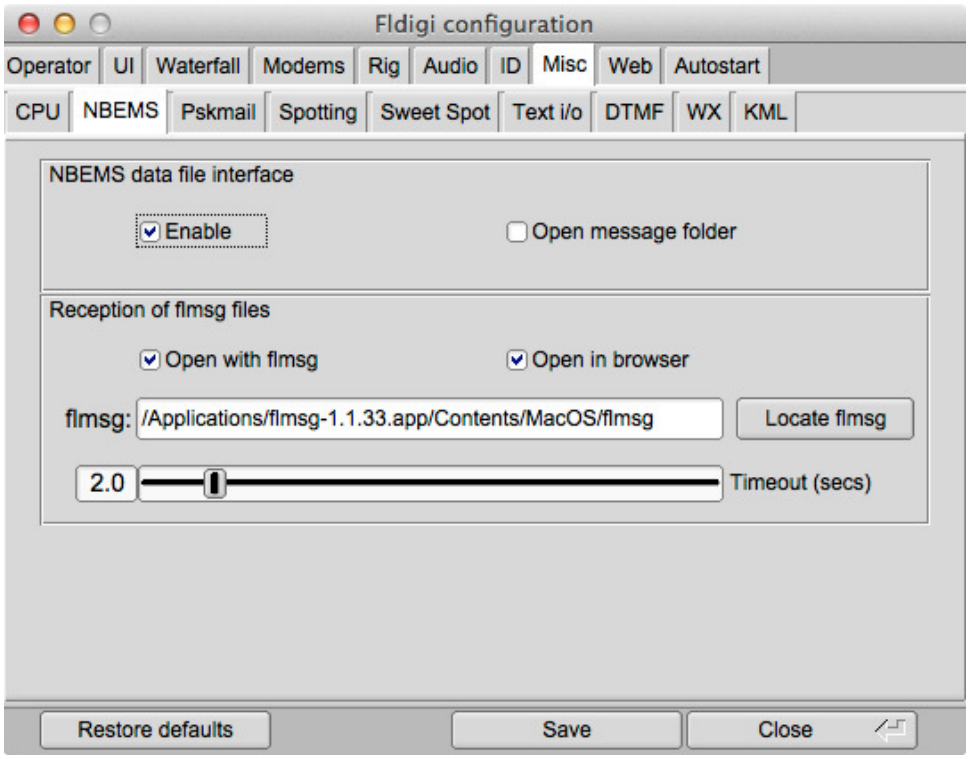


Figure 2.80: Config Misc NBEMS

The NBEMS suite of programs, fldigi, flarq, flwrap and flmsg provide the emergency operator with a set of tools to assist in the transfer of data files over HF and VHF radio.Additional information on flarq is available here:

- `flarq help system`.
- `wrap help system`.
- `flmsg help system`

The reception of a flwrap and flmsg files can be automated by selecting the "Enable detection & extraction" option. The wrap program can then be used to test for validity and data extraction at some later time. fldigi can recognize flmsg data files and automatically open the flmsg program with the newly received data stream. It can also transfer the data stream to flmsg and instruct flmsg to save the data file, unwrap and decode it, display the data in a fully formatted html page and then exit. Pressing "Locate flmsg" performs differently on the different OS that are supported.

- Linux - a file finder is opened to the /usr/local/bin/ folder. Select the flmsg executable and the entry box is correctly populated
- Windows - a file finder is opened to the "C:\Program Files\" folder. Drill down to the most current flmsg folder and then select the flmsg.exe file. The entry box will be correctly populated.
- OS X - a file browser is opened to the "Applications" folder. Locate the flmsg icon, and right click on it. Select "Show Package Contents". Double click "Contents". Double click on "MacOS". You will be viewing an icon labeled "flmsg". Drag and drop the icon on to the "flmsg:" entry box and the the value will be correctly entered.

2.27.6 DTMF decoding

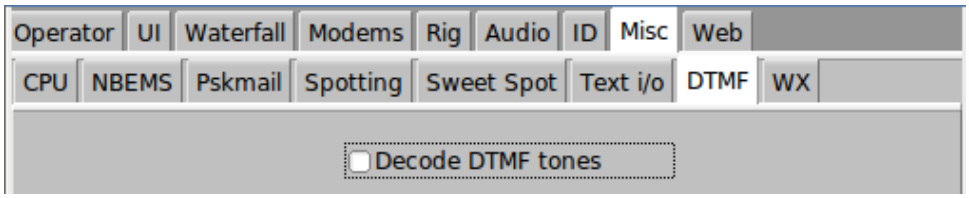


Figure 2.81: Misc DTMF Decoding

Fldigi can encode and decode DTMF tone sequences. Enable this check box to display the decoded tone sequence in the Rx panel. DTMF encoding is accomplished by a MACRO tag.

2.27.7 WX rss-xml queries

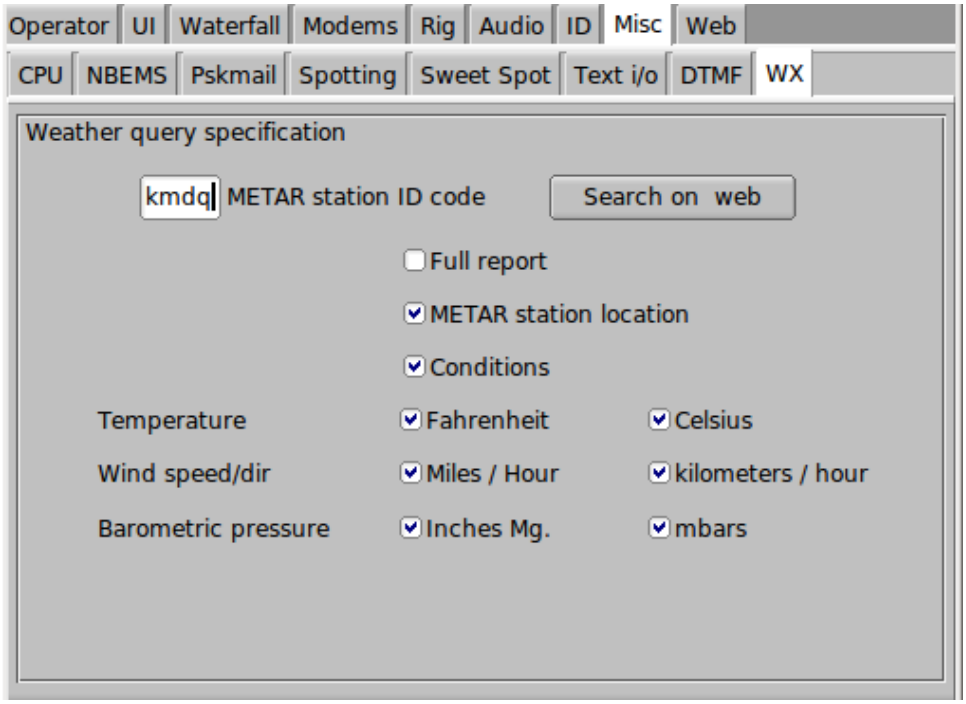


Figure 2.82: Misc WX

Fldigi provides an automated query of a specified RSS-XML feed to obtain and format weather data. The report is added to the transmit text stream using the appropriate MACRO tag. The report for the above configuration is:

```
Huntsville, Madison County Executive Airport, AL, United States (KMDQ) 34-51-41N 086-33-26W
Cond: overcast
Wind: 210 at 12 mph 19 kph
Temp: 62 F 17 C
Baro: 30.04 in Hg 1017 mbar
```

The full report option creates this report:

```
Huntsville, Madison County Executive Airport, AL, United States (KMDQ) 34-51-41N 086-33-26W
Feb 29, 2012 - 07:15 AM EST / 2012.02.29 1215 UTC
Wind: from the SSW (210 degrees) at 12 MPH (10 KT) gusting to 17 MPH (15 KT):0
```

```
Visibility: 10 mile(s):0
Sky conditions: overcast
Temperature: 62 F (17 C)
Dew Point: 60 F (16 C)
Relative Humidity: 93%
Pressure (altimeter): 30.04 in. Hg (1017 hPa)
```

The "Search on web" button will open your browser toGreg Thompson's global listing of METAR station.

RSS feeds for world wide airports can be searched here http://www.airrouting.com/content/-AirportLocatorForm.aspx.

Return to Top of Page

Return to Main Page

2.28 MT63 Configuration

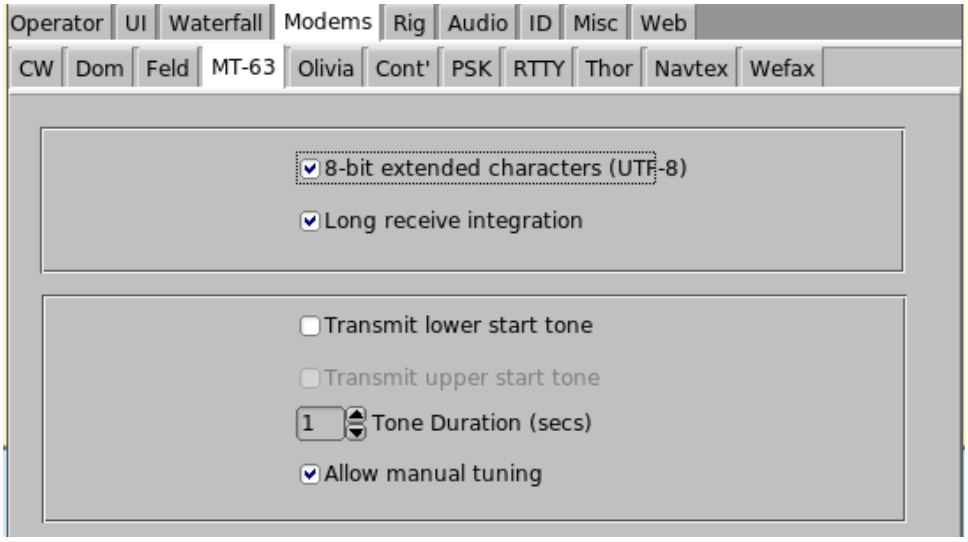


Figure 2.83: Modems MT63 Config Panel

MT63 is an orthogonal frequency division multiplexed mode consisting of 64 parallel carriers each carrying a part of the transmitted signal. There are 3 bandwidths and baudrates that fldigi implements in MT-63:

- 500 Hz - 5 baud
- 1000 Hz - 10 baud
- 2000 Hz - 20 baud

The lowest frequency transmitted is always 500 Hz. If you have a scheduled MT63 qso or are trying to copy what you think is MT63 you should tune the signal so that the lowest observable signal is at 500 Hz. Fldigi is capable of decoding signals that are mistuned by as much as +/- 100 Hz.

Selection of interleave (short/long) is made from the modem menu; MT63-500S, MT63-500L, MT63-1000S, MT63-1000L, MT63-2000S, MT63-2000L.

Interleave and 8-bit extended characters are usually agreed upon before a QSO exchange begins. The default is to use the long interleave. 8-bit extended characters allow the transmission of Latin-1 accented characters.

To assist the Rx operator you can transmit a short tone at the lowest (and highest) tone frequencies. You can set the tone duration.

You may also elect to use manual tuning (position on the waterfall) of the Tx / Rx signal. Be careful when doing this so that the signal does not extend beyond the limits of your transceiver SSB filters.

MT63 is used extensively in the Military Affiliate Radio System (MARS).

Return to Top of Page

Return to Main Page

2.29 Olivia Configuration

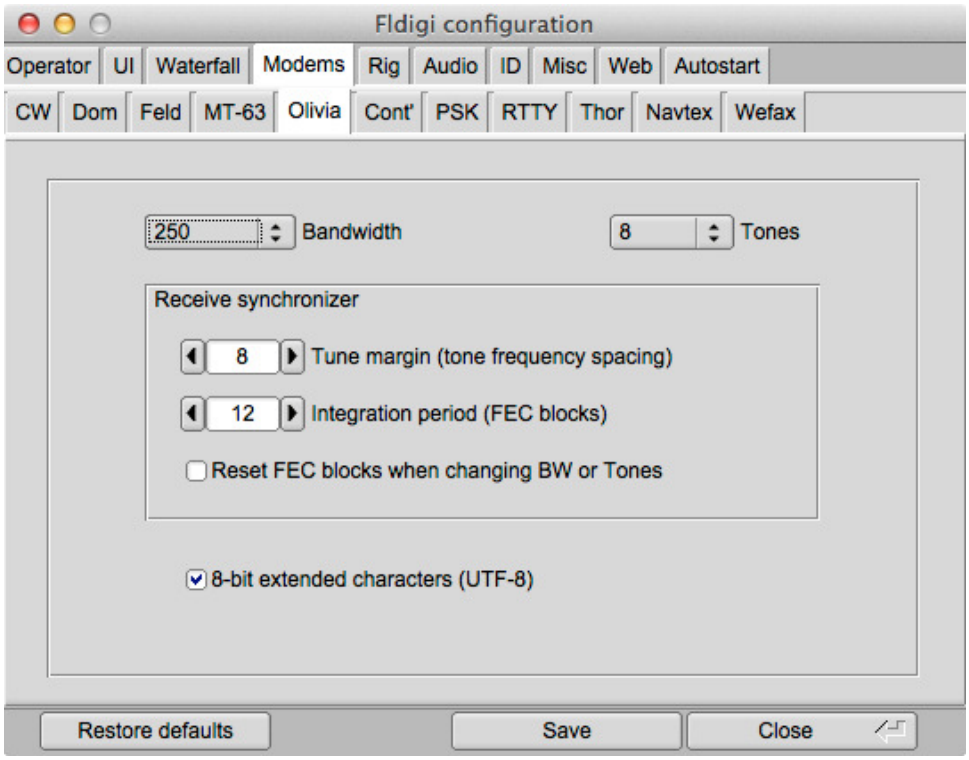


Figure 2.84: Description

Olivia is a family of MFSK modes with a high redundancy Forward Error Correction system similar to MT63. The family is very large, with 40 or more different options, which can make it very difficult to work out which is which. The mode works well on poor HF paths and has good sensitivity. There are three popular modes, which have 8-FSK, 16-FSK and 32-FSK, thus having three, four or five bits per symbol. These three modes can be selected without additional configuration. The tone frequency spacing and integration period should always be left at 8 and 4 respectively unless you are experimenting with another station running an Olivia modem that can be changed. These must always be the same at both ends of the Olivia QSO. The modes have two serious shortcomings - excessive bandwidth combined with slow typing rate, and excessive latency which is the apparent typing delay caused by the integration period.

See Operating Olivia for additional information.

Return to Top of Page

Return to Main Page

2.30 PSK Configuration

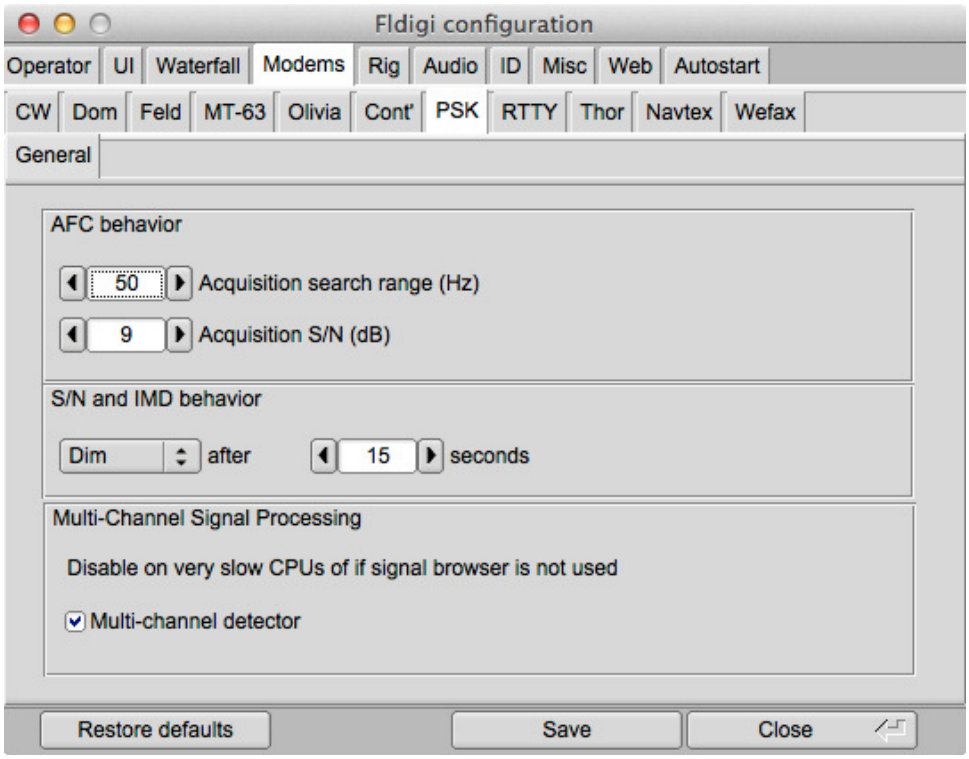


Figure 2.85: Modems PSK General

You should set the acquisition search range for waterfall left click action. As you adjust this control you will see the red mode width change on the waterfall scale. You can also adjust this value by pointing the mouse to the waterfall. Hold down the Control key and rotate the mouse wheel. The search routine which finds the PSK signal operates on a s/n threshold detector as well as recognizing the PSK phase modulation. You can adjust the acquisition signal to ratio threshold for the search routine.

The PSK decoder estimates the signal to noise ratio, S/N, and the intermodulation distortion, IMD, of the received signal. This measurement is valid during periods when the other station is transmitting the idle signal. The estimates are displayed on the status bar. You can control how these values are displayed; **clear** or **dim** after NN seconds. Setting the seconds to 0 disables the clear/dim action.

Fldigi has a multi channel browser than can display simultaneous reception of up to 30 PSK signals. The browser is described here:

Signal Browser

Return to Top of Page

Return to Main Page

2.31 RTTY / FSK Configuration

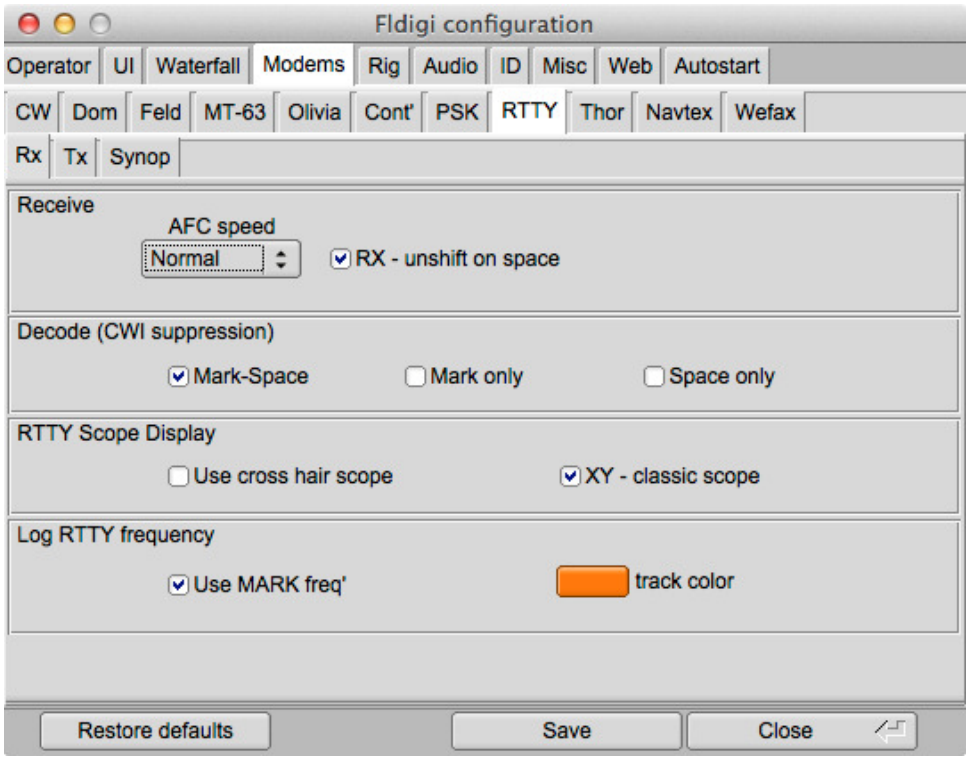


Figure 2.86: RTTY RX

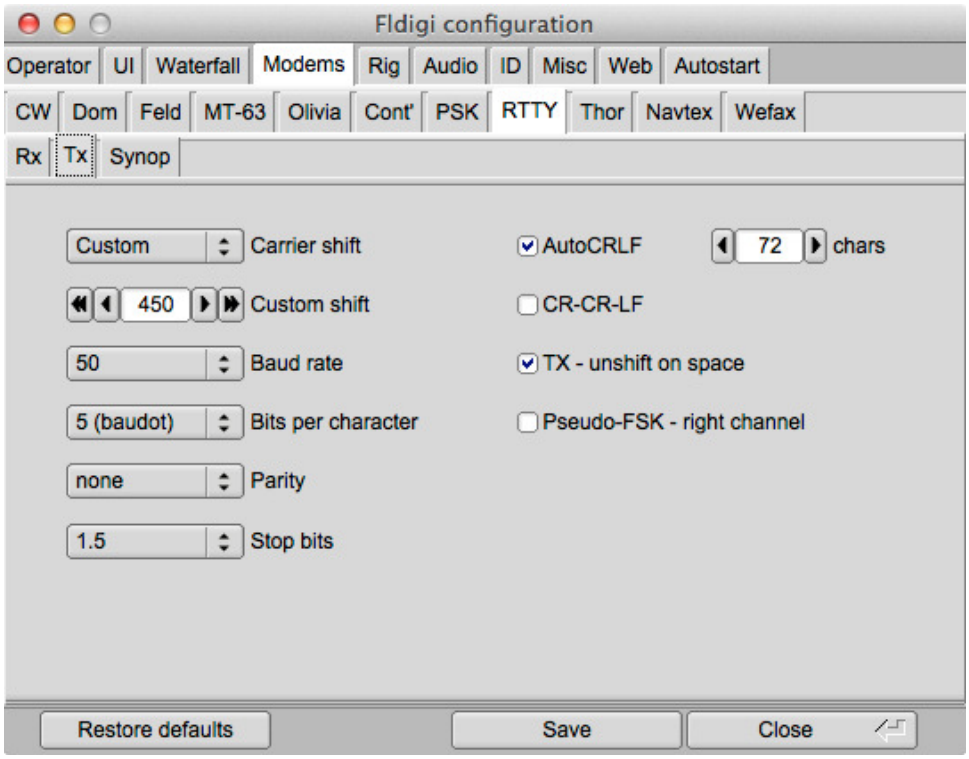


Figure 2.87: RTTY TX

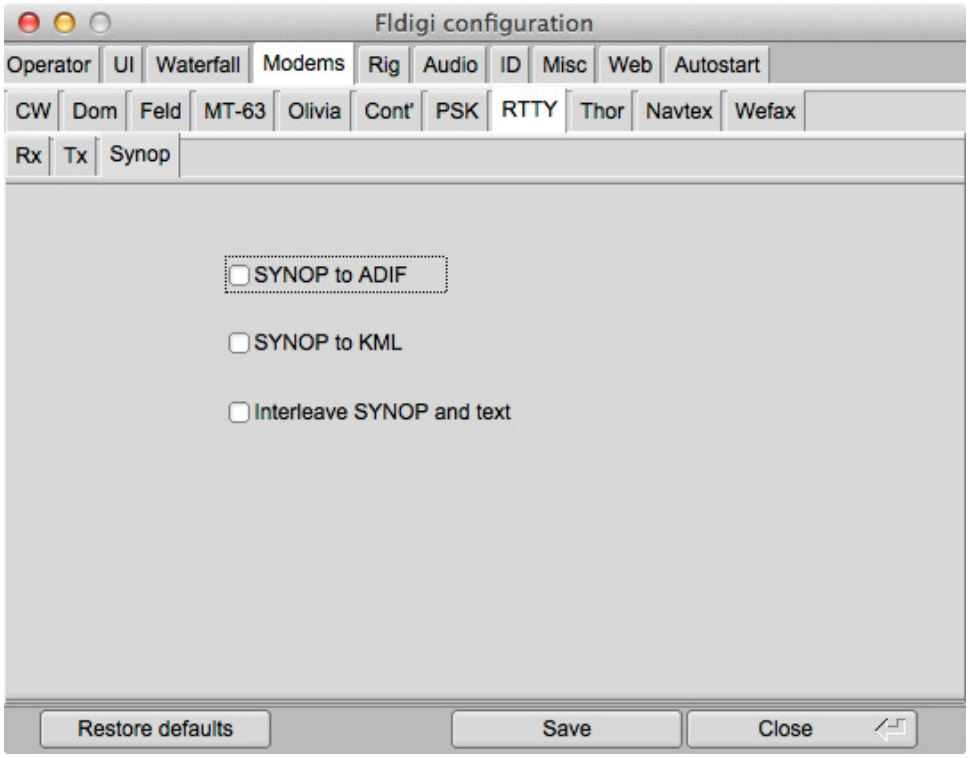


Figure 2.88: RTTY SYNOP

Fldigi operates RTTY using AFSK and the transceiver set to USB. The RTTY signal can be transmitted anywhere within the USB passband of the transceiver.

You can select from various Shifts, Bauds, Bits, Parity and Stop Bits for both AFSK and FSK keying of the transmitter. You can elect to have fldigi automatically insert a CFLF when it reaches character 72 on a line. You can also have it insert a CR-CR-LF sequence instead of the standard CR-LF sequence. This is very useful if you are communicating with someone using a hardware TTY printer. The extra carriage return will give the physical device time to move to the left margin before new characters arrive.

The RTTY decoder maintains an internal AFC system for tracking the desired signal. Depending on operating conditions you may need to adjust the action of the AFC loop. Select from the Slow, Normal or Fast AFC loop. You can also disable AFC with the AFC button on the main panel.

The received signal processing consists of a tuned Raised Cosine Filter followed by an optimized Automatic Threshold Correcting (ATC) detector. The resulting bit stream is then processed to extract the byte data.

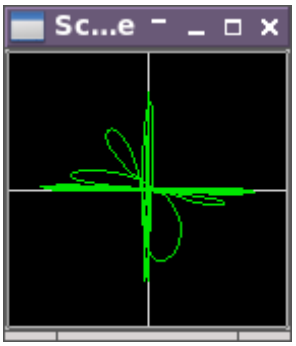


Figure 2.89: Classic RTTY XY Scope

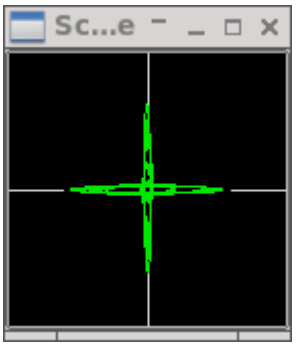


Figure 2.90: Pseudo Signal XY Scope

The Digiscope display can be defaulted to the X-scope or the Signal scope.

The classic XY-scope is similar to older hardware scopes that show the output of the separate Mark-Space filters, one applied to the horizontal and one to the vertical axis.

The Pseudo XY-scope is similar with the exception that the horizontal and vertical are phase related to the Mark--Space signals, but not the actual signal.

The Classis XY-scope will appear noisier than the Pseudo XY-scope.

PseudoFSK selection generates an additional audio signal on the right channel. This signal is a burst tone at the FSK keying rate. You can full wave rectify and filter the signal so that it can be used as the FSK keyline signal to a rig that supports FSK transmissions. See Pseudo FSK for additional information and a suitable keying circuit.

Your log can record either the center frequency between the Mark and Space or the Mark frequency. You can also select the color that should distinguish the Mark frequency.

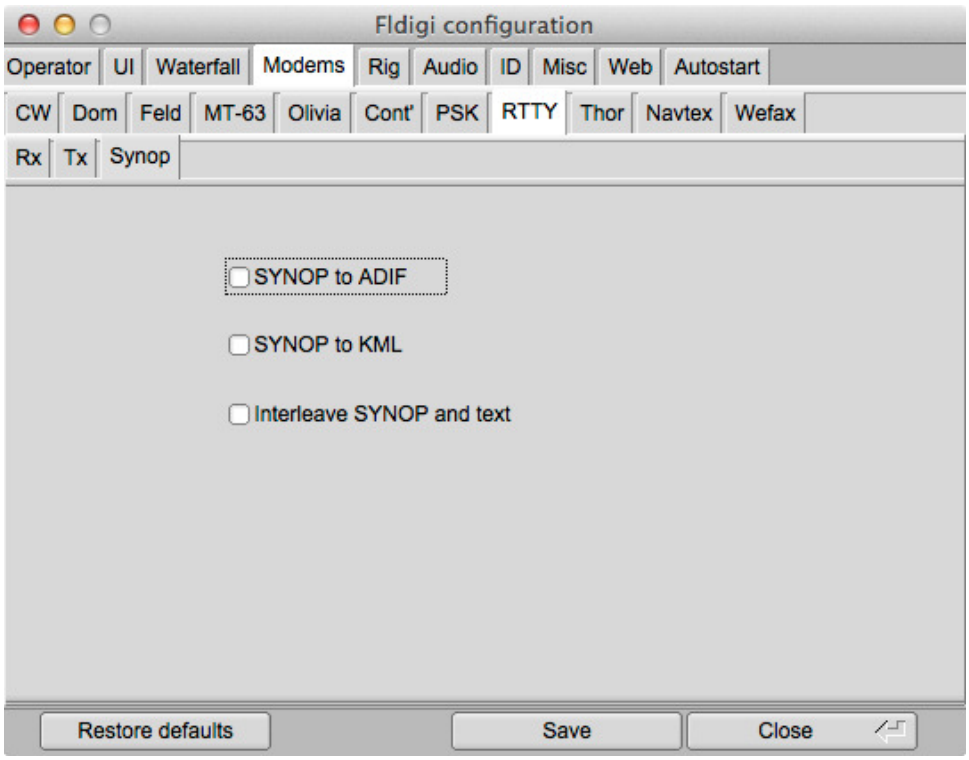


Figure 2.91: RTTY SYNOP

See Synop and RTTY for additional information.

Return to Top of Page

Return to Main Page

2.32 Thor Configuration

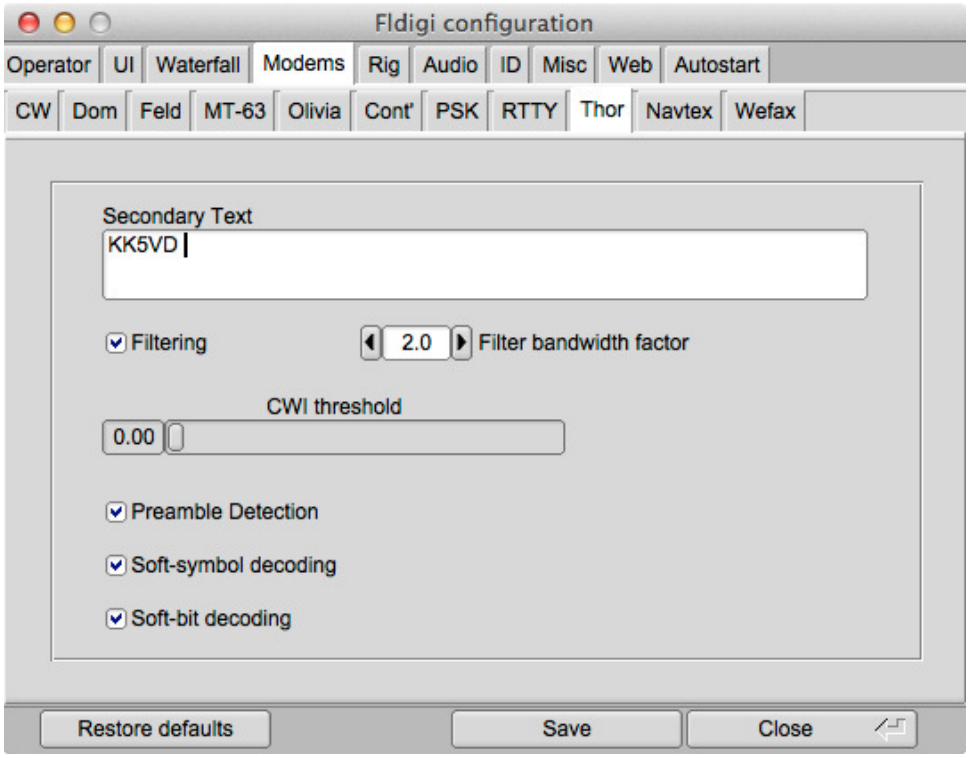


Figure 2.92: Modems Thor

The decoder can detect and defeat a modest amount of CWI that is within the BW set by the BW factor. Increasing the CWI threshold increasing the sensitivity for this correction. The offending tones are punctured thereby rendering them null to the Viterbi decoder.

Enter the secondary text. This text will be sent during periods when your keyboard is inactive (between letters for slow typists). The default for this text will be your callsign when you have entered that in the Operator configuration tab.

Set the BW factor for the decoding prefilter. 2.0 should be adequate unless you are experiencing nearby continuous wave interference (CWI). You can enable and disable the prefilter with the checkbox. Please note that the filter requires additional cpu cycles. Older and slower cpu models might give better decoding with the filter disabled.

The DominoEX decoder can detect the presence of CWI within the passband set by the BW factor. Increasing the CWI threshold increases the sensitivity to such interference. When the interference is detected the associated data is culled using a technique called puncturing.

Thor has been specifically designed to be used with ARQ text transmissions. It is also an easy to use keyboard chat mode. Thor operations are described in Operating Thor.

Return to Top of Page

Return to Main Page

Chapter 3

Logging

- Cabrillo Report
- Exporting Logbook Data
- QSO Logbook
- User Interface Configuration - Log server

3.1 Cabrillo Report

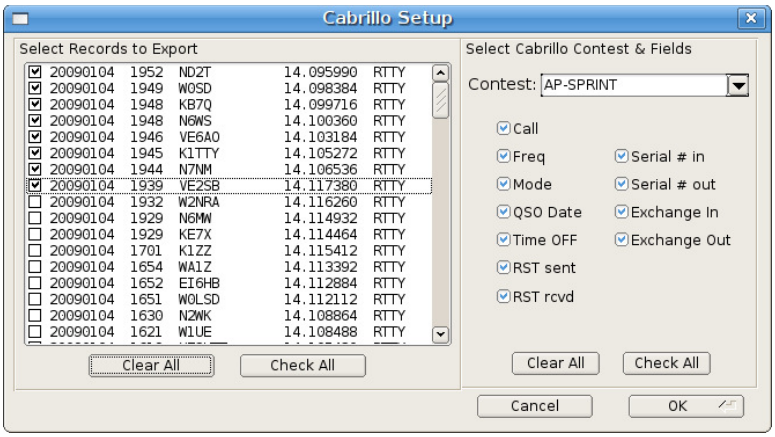


Figure 3.1: Cabrillo Report

Fldigi can generate a basic Cabrillo report that meets most contest needs.

Selecting the "File/Log/Cabrillo report" menu item opens the following dialog:

If you want to export every record press the "Check All" in the left panel.

Select the Contest type from the pull down menu in the right panel. Fldigi knows how to format the various fields for each contest. When satisfied with the setup press OK. You will then have the opportunity to specify the location and name of the cabrillo output file.

If you use the Contest - How To <XBEG>...<XEND> to capture your outgoing exchange, OR if you choose to capture all received exchange information in the "Exch" field on the QSO Logbook panel then you should be careful not to duplicate the data in the Caprillo report. For example:

You captured outgoing exchange data as <XBEG><RST> <CNTR><XEND>. Do not enable the "RST sent" and "Serial # out" check boxes as this info is already contained in the "Exchange in" field.

You captured incoming exchange data into the Exch field as RST SER# TIME. Do not enable the "RST rcvd" or the "Serial # in" check boxes.

You must then open the file with a plain text editor and modify the appropriate entries. Check with each contest sponsor to see what their requirements are.

Here is an example of a generated cabrillo report format before being edited:

```
START-OF-LOG: 3.0
CREATED-BY: fldigi 3.11

# The callsign used during the contest.
CALLSIGN: W1HKJ

# ASSISTED or NON-ASSISTED
CATEGORY-ASSISTED:

# Band: ALL, 160M, 80M, 40M, 20M, 15M, 10M, 6M, 2M, 222, 432, 902, 1.2G
CATEGORY-BAND:

# Mode: SSB, CW, RTTY, MIXED
CATEGORY-MODE:

# Operator: SINGLE-OP, MULTI-OP, CHECKLOG
CATEGORY-OPERATOR:

# Power: HIGH, LOW, QRP
CATEGORY-POWER:

# Station: FIXED, MOBILE, PORTABLE, ROVER, EXPEDITION, HQ, SCHOOL
CATEGORY-STATION:

# Time: 6-HOURS, 12-HOURS, 24-HOURS
CATEGORY-TIME:

# Transmitter: ONE, TWO, LIMITED, UNLIMITED, SWL
CATEGORY-TRANSMITTER:

# Overlay: ROOKIE, TB-WIRES, NOVICE-TECH, OVER-50
CATEGORY-OVERLAY:

# Integer number
CLAIMED-SCORE:

# Name of the radio club with which the score should be aggregated.
CLUB:

# Contest: AP-SPRINT, ARRL-10, ARRL-160, ARRL-DX-CW, ARRL-DX-SSB, ARRL-SS-CW,
# ARRL-SS-SSB, ARRL-UHF-AUG, ARRL-VHF-JAN, ARRL-VHF-JUN, ARRL-VHF-SEP,
# ARRL-RTTY, BARTG-RTTY, CQ-160-CW, CQ-160-SSB, CQ-WPX-CW, CQ-WPX-RTTY,
# CQ-WPX-SSB, CQ-VHF, CQ-WW-CW, CQ-WW-RTTY, CQ-WW-SSB, DARC-WAEDC-CW,
# DARC-WAEDC-RTTY, DARC-WAEDC-SSB, FCG-FQP, IARU-HF, JIDX-CW, JIDX-SSB,
# NAQP-CW, NAQP-RTTY, NAQP-SSB, NA-SPRINT-CW, NA-SPRINT-SSB, NCCC-CQP,
# NEQP, OCEANIA-DX-CW, OCEANIA-DX-SSB, RDXC, RSGB-IOTA, SAC-CW, SAC-SSB,
# STEW-PERRY, TARA-RTTY
CONTEST: ARRL-RTTY

# Optional email address
EMAIL:

LOCATION:

# Operator name
NAME:

# Maximum 4 address lines.
ADDRESS:
ADDRESS:
ADDRESS:
ADDRESS:

# A space-delimited list of operator callsign(s).
OPERATORS:
```

```
# Offtime yyyy-mm-dd nnnn yyyy-mm-dd nnnn
# OFFTIME:

# Soapbox comments.
SOAPBOX:
SOAPBOX:
SOAPBOX:

QSO: 14095 RY 2009-01-04 1952 W1HKJ 599 GA 12345   ND2T          599 CA 67890
QSO: 14098 RY 2009-01-04 1949 W1HKJ 599 GA         W0SD          599 SD
QSO: 14099 RY 2009-01-04 1948 W1HKJ 599 1234567890 KB7Q          599 1234567890
QSO: 14100 RY 2009-01-04 1948 W1HKJ 599 GA         N6WS          599 CA
QSO: 14103 RY 2009-01-04 1946 W1HKJ 599 GA         VE6AO         599 AB
END-OF-LOG:
```

Return to Top of Page

Return to Main Page

3.2 Exporting Logbook Data

Fldigi provides automatic export of log records as they are recorded. On Linux the data is forwarded to Xlog compatible programs using the SysV message queue system. On Windows the records are exported via a temporary file structure and are accepted by Logger32.

The user may also export all or selected records consisting of all or selected fields. Access to this export function of available from the menu "File/Log/Export ADIF", "File/Log/Export Text", and "File/Log/Export CSV".

3.2.1 Export ADIF

Selecting the Export ADIF menu item opens the following dialog:

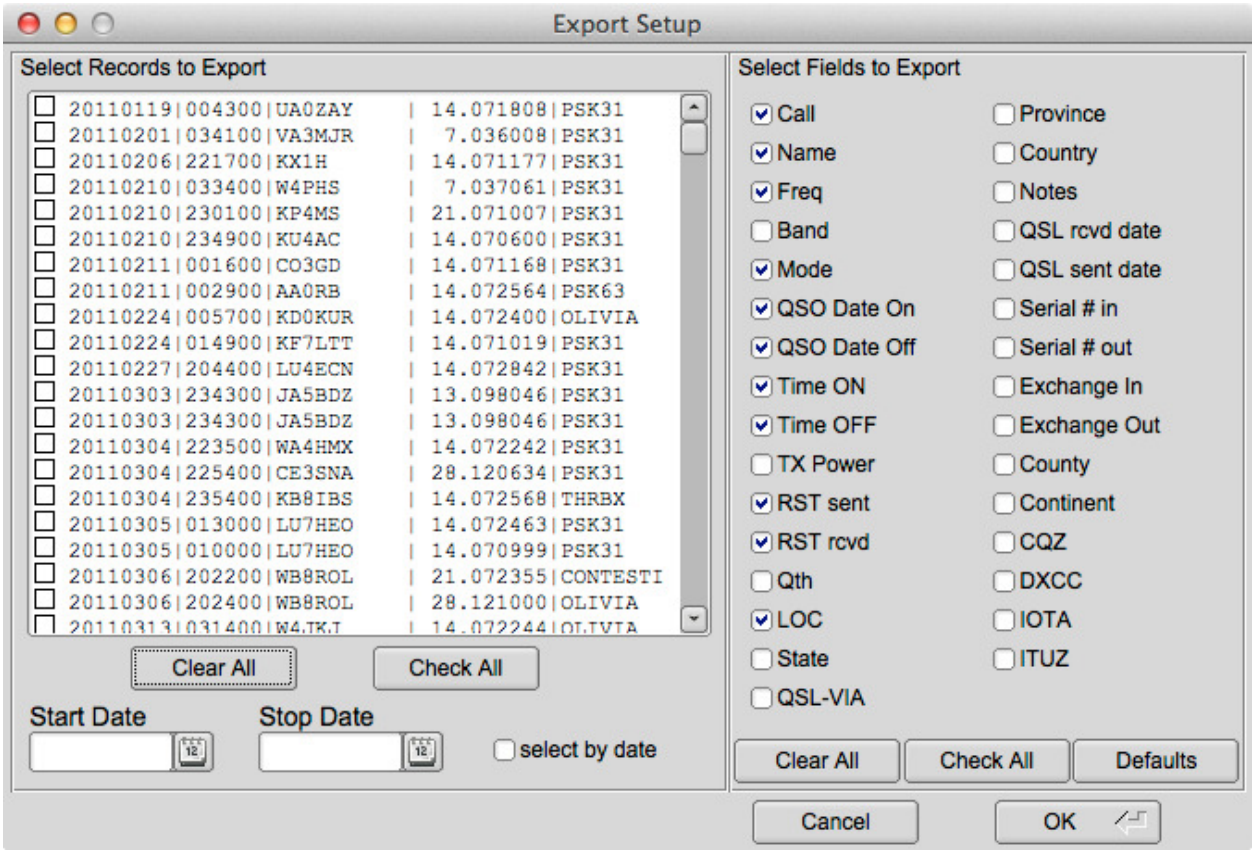


Figure 3.2: Export Setup

If you want to export every record press the "Check All" in the left panel. You can also select and deselect individual records. Choose which fields you want to export with the right panel controls. Press the OK button to continue or Cancel to abort the operation. A file chooser dialog will open which allows you to specify the name and location of the exported file. Use the extension ".adi" on Windows and ".adif" on the other OS's.

3.2.2 Export Text / CSV

The same Export Setup dialog is used for Text and CSV exports.

The Text export produces a simple space delimited file with columns set at locations dictated by the field size for each field that is exported. It is suitable for use with a word processing program or for printing a hardcopy of your activities.

The CSV is a "Character Separated Value" file with the TAB character used as the field separator. This type of file can be imported into nearly all spreadsheet programs such as Gnumeric, Open Office or MS Excel.

Return to Top of Page

Return to Main Page

3.3 QSO Logbook

Fldigi maintains a large set of QSO logbook fields that will probably be sufficient for casual operating, contesting and some certificate logging. All of the fields that are captured in the logbook are maintained in an ADIF database that can be read by any logbook program that can read the ADIF text format.

3.3.1 List of Log Fields

The complete set of logbook fields are:

ADIF FIELD		USE
BAND		QSO band (computed from frequency)
CALL	*	contacted stations call sign
COMMENT	*	comment field for QSO
COUNTRY	*	contacted stations DXCC entity name
CQZ		CQ zone
DXCC		contacted stations DXCC country code
FREQ	*	QSO frequency in MHz
GRIDSQUARE	*	contacted stations Maidenhead Grid Square (Loc)
IOTA		Islands On The Air
IOTA		Islands-On-The-Air designator
ITUZ		ITU zone
MODE		QSO mode
MYXCHG		sent contest exchange
NAME	*	contacted operators name
QSLRDATE		QSL received date
QSLSDATE		QSL sent date
QSO_DATE	*	QSO data at start of contact
QTH	*	contacted stations city
RST_RCVD	*	received signal report
RST_SENT	*	sent signal report
SRX	*	QSO received serial number
STATE	*	contacted stations state
STX	*	QSO transmitted serial number
TIME_OFF	*	end time of QSO in HHMM format
TIME_ON	*	start time of QSO in HHMM format
TX_PWR	*	power transmitted by this station
VE_PROV	*	2 letter abbreviation for Canadian Province
XCHG1	*	received contest exchange

- - These fields are either captured on the main dialog, computed from internal values, or determined by configuration.

The data in the fldigi logbook can be exported to external text files; ADIF, text, and CSV (comma separated value). The ADIF can be read by any ADIF compatible logbook program. The text output is suitable for use in a word-processor and for printing. The CSV can be read into many spreadsheet programs such as Excel, Open Office or Gnumeric.

3.3.2 Digital Modes Signal Reports

Fldigi does not enforce any rules on signal reporting. It could very well do so for many of the modes in which signal quality is inherently measured as a part of the decoder. Learning how to evaluate a signal, to properly report it, and then help in correcting deficiencies should be the goal of every amateur operator. Please read further on using both RST and RSQ signal reports.

3.3.3 Capturing QSO Data

Fldigi supports two QSO capture panels. The first for casual QSO logging

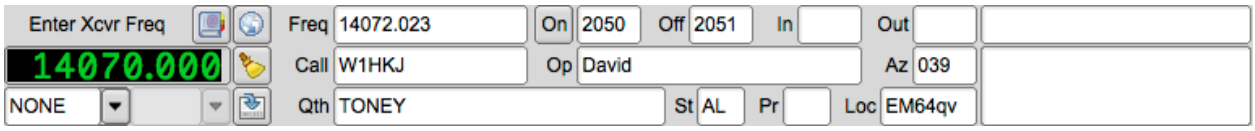


Figure 3.3: QSO capture panel

and the second for contest fields

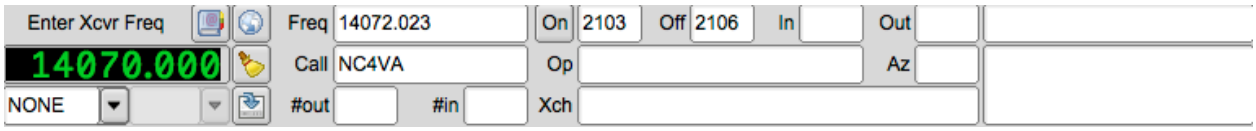


Figure 3.4: Contest capture panel

You might prefer a more minimal view of the logging fields. You can select to completely suppress the log panel or to use a single line view as in either of these two:

Figure 3.5: Minimal QSO Capture Panel

Figure 3.6: Minimal Contest Capture Panel

These are selectable from the View menu:

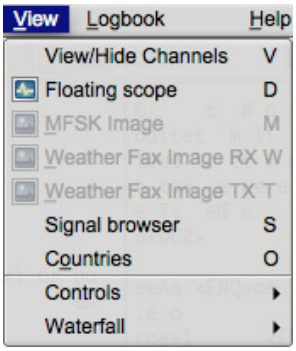


Figure 3.7: View menu

These three buttons are associated with logbook entries.

- The globe button provides access to QRZ queries
- The brush button clears all of the logging entries
- The save-into button saves the current logging entries into the logbook

Figure 3.8: Log Butons

The frequency, Off (time off), and #Out are filled by the program. All the others can be populated by manual keyboard entry or by selection from the Rx panel. The time off, Off, is continuously update with the current GMT. The time on, On, will be filled in when the Call is updated, but can be modified later by the operator.

A right click on the Rx panel brings up a context sensitive menu that will reflect which of the two QSO capture views you have open.

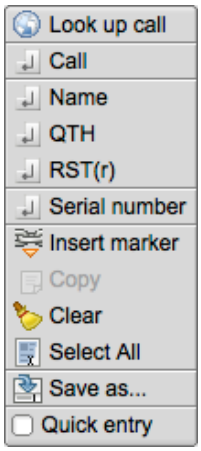


Figure 3.9: Normal: Short Menu

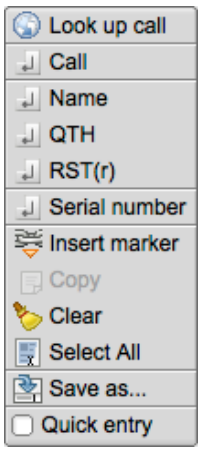


Figure 3.10: Normal: Long Menu

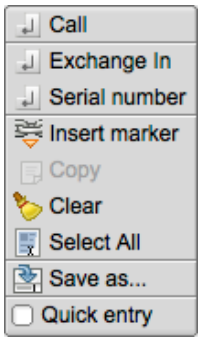


Figure 3.11: Contest: Short Menu

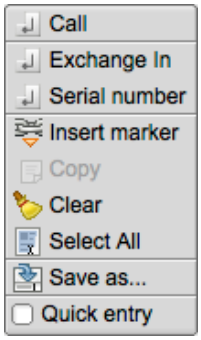


Figure 3.12: Contest: Long Menu

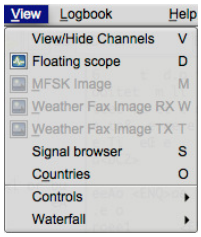


Figure 3.13: Contest: Long Menu

If you highlight text in the Rx pane then the menu selection will operate on that text. If you simply point to a word of text and right click then the menu selection will operate on the single word.

Certain fields may also be populated with automatic parsing, Call, Name, QTH and Loc. You point to the Rx pane word and then either double-left-click or hold a shift key down and left click. The program will attempt to parse the word as a regular expression to populate the Call, Name, QTH, and Loc fields in that order. It may place some non standard calls into the Loc field if they qualify as a proper Maidenhead Grid Square, such as MM55CQ. That may be a special event station, but it also looks like a grid square locator value. You need to decide when that occurs and use the pop up menu for those special cases. The first non-Call non-Loc word will fill the Name field and subsequent qualify words will go into the QTH field.

A highlighted section of text, can always be copied to the clipboard for subsequent pasting elsewhere. The Copy menu item will be active when text in the Rx pane has been highlighted. That text can also be saved to a file. Use the "Save as..." menu item for that purpose. All data fields in fldigi share a common set of keyboard shortcuts. Linux users will recognize these as familiar Emacs shortcuts. There is also a small popup menu that can be opened for each field by right clicking the contents with the mouse:

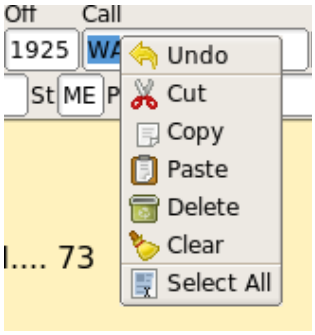


Figure 3.14: Menu Popup Edit Field

Highlighted text will be overwritten when a paste is selected. Otherwise the clipboard will be pasted at the current cursor position.

You can query on-line and local CD based data base systems for data regarding a Call. Set up your query using the Callsign DB configuration tab. You make the query by either clicking on the globe button, or selecing "Look up call" from the menu. The latter will also move the call to the Call field and make the query.

If you have previously worked a station the logbook will be searched for the most recent qso and fill the Name, Qth and other fields from the logbook. If the logbook dialog is open that last QSO will be selected for viewing in the logbook.

You open the logbook by selecting from the Logbook menu; Logbook/View. The logbook title bar will show you which logbook you currently have open. Fldigi can maintain an unlimited (except for disk space) number of logbooks.

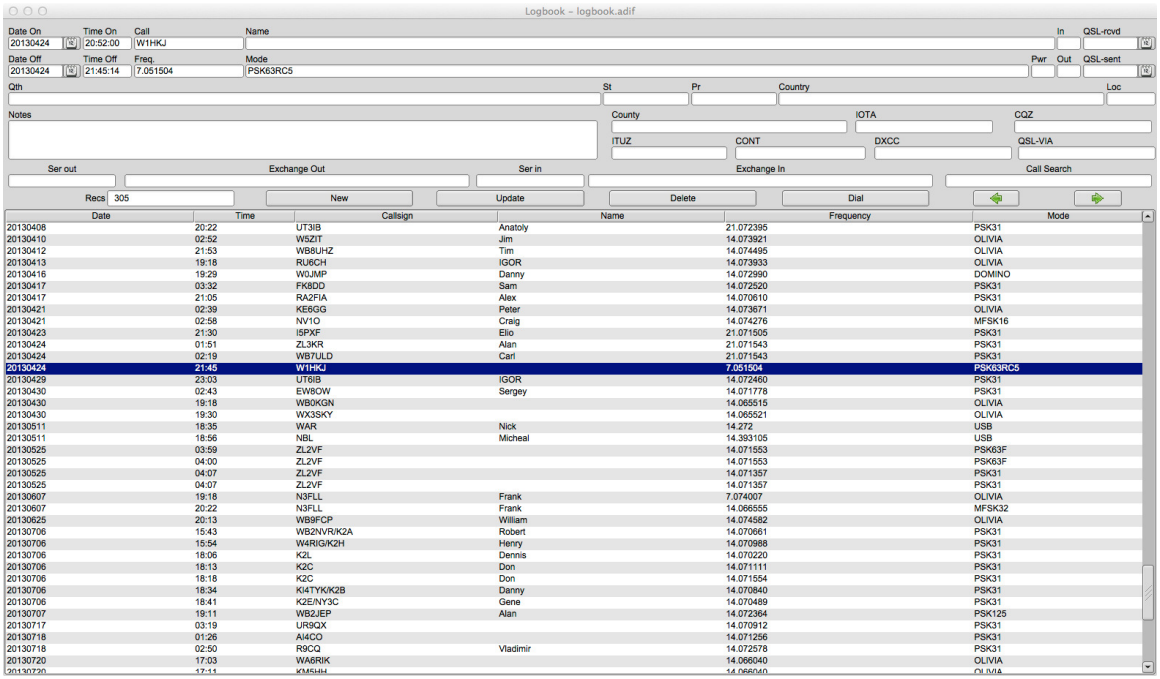


Figure 3.15: Log Book

You can resize the dialog to suit your screen size and operating needs. Fldigi will remember the placement and size for subsequent use.

You can create new entries, update existing entries, and delete entries using this dialog. You can also search for an entry by callsign. The browser can be sorted by Date, Callsign, Frequency or Mode. The sort can be forward or backward with the most recent being the default selected entry after each sort. You execute the sort by clicking on the column button at the top of the column to be sorted. Each click causes the sort to reverse. I like to view my log

with the most recent at the top. You might want to view it with the most recent on the bottom.

There are no frills such as keeping track of DXCC worked, fancy printouts etc. Fldigi's logbook is primarily a capture function. You can export your data for use with an external database or for uploading to LOTW or eQSL. Data from those sources can also be used for importing into the logbook.

Exporting logbook data: Log Exports.

Cabrillo reporting: Contest Reports.

Return to Top of Page

Return to Main Page

3.4 User Interface Configuration - Log server

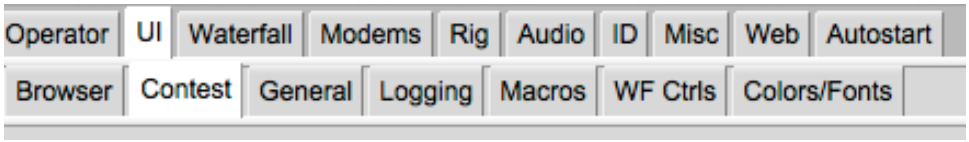


Figure 3.16: UI Tabs

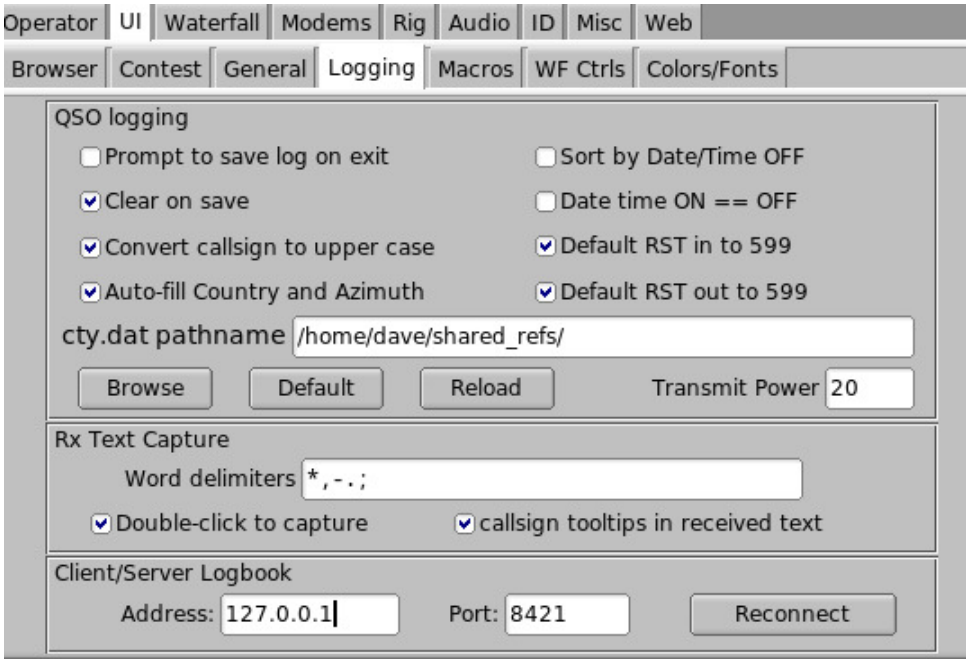


Figure 3.17: QSO Logging

3.4.1 Capturing Log data

Fldigi has a built in logbook. You can request to be prompted whenever there is an unsaved entry in the qso log area. You can also elect to whether to clear all of the qso fields when the log is saved or to leave them intact. Auto-fill Country and Azimuth uses the data found in the file "cty.dat" that you should download and place in the fldigi default folder. You can force the callsign field to be upper case independent of capture or keyboard entry. You enter your default Transmit Power which is used for the logbook record.

Fldigi has various ways to transfer data in the Rx panel to the qso logging fields. The default is to use a Shift-Left--Click paradigm. You can also use a double click method if you prefer. The Shift-Left-Click will still function. Each

data item is considered to be a single word normally delimited by the space, tab or end-of-line character. You can add word delimiter characters in the designated text box. The default is to add ∗-,.; to the normal delimiters. This is a useful tool for extracting contest exchange data. The exchange might be RST, STATE and NAME. The station being worked might send this as 599-NJ-Bozo. Double clicking on the 599 NJ and Bozo would treat each as a separate word.

You can elect to have the RST in/out preset to 599 after you clear the QSO entry fields.

If you check the "callsign tooltips in received text" then the Rx text area will popup an information box whenever the mouse is held over a callsign for more than 2 seconds. The popup will look like one of the following:

3.4.2 Pop ups

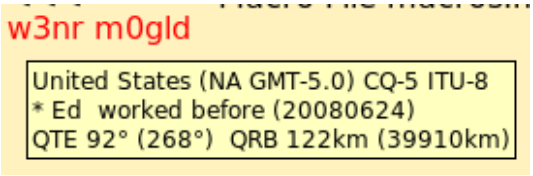


Figure 3.18: Pop ups

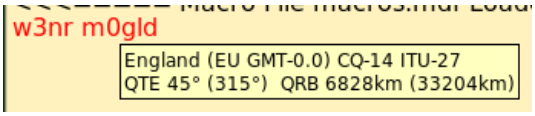


Figure 3.19: Pop ups 2

3.4.3 Country Files

The data is derived by parsing the callsign and referral to both the logbook and the "cty.dat" file. If the station was previously worked the operator's name and azimuth/distance will be computed from the logbook gridsquare entry (Loc). Otherwise the azimuth/distance is computed from the data in the cty.dat file.

This file is maintained by, and can be downloaded from the following web site:

http://www.country-files.com/

The default location for this file is in the fldigi default files folder. You have the option of changing that file location by either entering the new folder pathname in the edit control, or by using the "Browse" or "Default" button. If you change the contents of cty.dat while fldigi is running you should force fldigi to reload the data from the file. This data is normally only read when the program starts.

3.4.4 Internal Logbook

Fldigi has an internal logbook. That log should be used for single operator operations. There may be times that you need to share a log, either between programs on a single computer, or with other operators running fldigi on other computers on a LAN (or even WAN).

In lieu of the internal logbook you can elect to use a common logbook server. This logbook is maintained by a separate logbook program, fllog. fllog provides access to read, query and update records via an xmlrpc socket interface. fllog provides the server function and connecting applications are clients.

You need to specify both the socket address and socket port. The defaults are as shown and are for the instance when both fllog and fldigi are on a single computer. Unless configured otherwise, fllog will always use the port address 8421.

3.4.5 Logbook Network Address

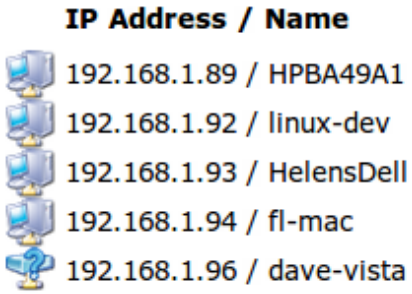


Figure 3.20: Network Address

My home network has IP address assignments as shown. If fllog were running on the mini-mac (fl-mac) and fldigi running on the linux-dev machine. I would enter the server address 192.168.1.94 into the fldigi configuration for the Client/Server Logbook.

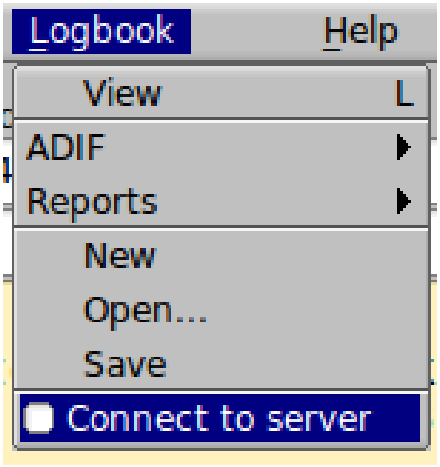


Figure 3.21: Connect to Server

The "Logbook/Connect to server" menu item allows you to connect to the remote logbook. If successful then the toggle remains checked and the menu items for accessing the internal logbook are disabled.

Return to Top of Page

Return to Main Page

Chapter 4

Macros

4.1 Macros

Macros are short text statements that contain imbedded references to text data used by the program fldigi. Macro definition files(s) are located in the $HOME/.fldigi/macros/ directory and all have the extention ".mdf". The default set of macros are contained in the file $HOME/.fldigi/macros/**macros.mdf**.

Fldigi will create this file with a set of default macros on its first execution.

Fldigi supports up to 48 macro definitions in sets of 12. Macro definitions are not recursive, that is; a macro cannot reference another macro or itself.

The imbedded references are similar to those used by DigiPan and other fine modem programs. The imbedded reference is an uppercase plain text descriptor contained with the <> brackets.

4.1.1 Macro tags

Macro	Description
<FREQ>	my frequency
<MODE>	mode
<MYCALL>	configuration call
<MYLOC>	configuration locator
<MYNAME>	configuration name
<MYQTH>	configuration QTH
<MYRST>	my RST
<ANTENNA>	configuration antenna
<VER>	Fldigi version

Macro	Description
<CALL>	other stations callsign

<INFO1>	S/N or other data contained in first info field of status bar
<INFO2>	IMD or other data contained in second info field of status bar
<LOC>	other stations locator
<NAME>	other stations name
<QTH>	other stations QTH
<RST>	other stations RST

Macro	Description
<MAPIT>	map other stations locator on google
<MAPIT:adr | lat | loc>	map other stations address, latitude-longitude locator as specified
<CLRRX>	clear RX pane
<CLRTX>	clear TX pane

Macro	Description
<GET>	text to NAME/QTH
<TALK:on|off|t>	Digitalk On, Off, Toggle; this is a Windows only tag and Digitalk must be running
<LOG>	save QSO data to logbook immediately
<LNW>	log at xmt time
<CLRLOG>	clear log fields
<EQSL:[message>	submit the current log entry to www.eQSL.cc This macro tag should be put before <LOG> or <LNW>

Macro	Description
<QSOTIME>	insert current logbook time HHMM, ie 0919
<ILDT>	insert current local date-time in iso-8601 format, ie 2011-08-28 04:16-0500
<LDT>	insert Local date-time, ie 2011-08-28 04:16-0500
<IZDT>	insert Zulu date-time in iso-8601 format, ie 08/28/2011 04:16 CDT
<ZDT>	insert Zulu date-time, ie 2011-08-28 09:16Z
<LT>	insert local time, ie 0416
<ZT>	insert zulu time, ie 0916Z
<LD>	insert local date, ie 2011-08-28
<ZD>	insert Zulu date, ie 2011-08-28
<WX> <WX:xxxx>	insert current weather data from METAR as specified on WX configuration tab. see WX configure replace xxxx with the 4 letter METAR designator for a report on a station other than the one specified on the weather configuration tab.

Macro	Description
<CNTR>	insert current value of contest counter
<DECR>	decrement contest counter
<INCR>	increment contest counter
<XIN>	send exchange in string

<XOUT>	send exchange out string
<XBEG>	mark exchange in start
<XEND>	mark exchange in end
<SAVEXCHG>	save marked text to contest exchange in

Macro	Description
<RX>	receive
<TX>	transmit
<TX/RX>	toggle Transmit / Receive

Macro	Description
<SRCHUP>	search UP for next signal
<SRCHDN>	search DOWN for next signal

Macro	Description
<CSV:on|off|t>	on, off, toggle Analysis file

Macro	Description
<GOHOME>	return to waterfall cursor to sweet spot
<GOFREQ:NNNN>	move waterfall cursor to freq NNNN Hz
<QSYTO>	same as left-clk on QSY button
<QSYFM>	same as right-clk QSY button
<QSY:FFF.F[:NNNN]>	Qsy to transceiver frequency in kHz, optional waterfall (Audio) frequency in Hz (If not specified, it i not changed). Several QSY frequencies, or ranges of frequencies provided by one increment, can be proposed, in which case the first frequency after the current frequency is chosen.

Macro	Description
<RIGMODE:mode>	send CAT command to transceiver to change to a valid mode
<FILWID:width>	send CAT command to transceiver to change to a valid filter width example to QSY to sweetspot (center of bandpass filter) and select narrow filter

Macro	Description
<QSYTO><FILWID:200> <QSYFM><FILWID:2700>	example to restore previous waterfall cursor frequency and bandwidth
<FILE:>	insert text file; a file selection box will open when this tag is selected during editing

Macro	Description
<IDLE:NN.nn>	transmit idle signal for NN.nn sec
<TIMER:NN>	repeat this macro every NN sec
<TUNE:NN>	transmit single tone tune signal for NN sec
<WAIT:NN>	insert delay of NN seconds before transmitting
<REPEAT>	repeat macro continuously
<SKED:hhmm[:YYYYDDMM]>	schedule execution to begin at time and optionally date specified

Macro	Description
<CWID>	transmit a CW callsign identifier
<ID>	transmit mode ID using waterfall video text
<TEXT>	transmit video text defined on ID configuration tab
<TXRSID:on|off|t>	transmit RSID on, off, toggle
<RXRSID:on|off|t>	receive RSID on, off, toggle
<NRSID:NN>	transmit multiple RsID bursts NN < 0 will transmit NN bursts for current modem and then return to Rx NN > 0 will transmit NN bursts for current modem and continue in Tx NN = 0 will transmit 1 bursts and continue (same as NN = 1)
<DTMF:[Wn:][Ln:]>	tones transmit DTMF tone sequence at start of transmission; options W-wait n msec, default 0 L-tone symbol length n in msec; default 50 msec '-', ' ' and ',' insert silence symbol eg: <DTMF:W250:L100:1-256-827-3200>

Macro	Description
<POST:+/-nn.n>	CW QSK post-timing in milliseconds
<PRE:nn.n>	CW QSK pre-timing in milliseconds
<RISE:nn.n><RISE:nn.n>	CW rise time in milliseconds
<WPM:ww[:ff]>	CW WPM, ww = word rate, optional ff = Farnsworth rate

Macro	Description
<AFC:on|off|t>	AFC on, off, toggle
<LOCK:on|off|t>	lock waterfall cursor; on, off, toggle
<REV:on|off|t>	Reverse on, off, toggle

Macro	Description
<QRG:text>	Insert current operating info with "text" into Rx stream, ie: info text <<2013-01-12T21:18Z RTTY @ 14005000+0760>> which can be used to return to a mode, rf, audio frequency.
<PAUSE>	Cause transmission to pause at place of occurance in macro text. "Pause/Break" key on keyboard resumes transmission
<TXATTEN:nn.n>	set fldigi tx attenuator to value -30 dB <= val <= 0
<COMMENT:text>	allow macro to contain a comment field; everything between < and > is ignored by macro parser
<SAVE>	save the current macro definitions to the current file

Macro	Description
<MACROS:>	load a new macro defs file; file prompt when editing macro

```
* Added macro to insert QRG text into Rx stream
  <QRG:text to insert>
* Added save macro <SAVE> tag
* Added export strings
  - FLDIGI_LOG_FILE - current logbook file name
  - FLDIGI_MACRO_FILE - current macro file name
* Added <PAUSE> tag
```

```
  - returns to receive, but does not clear Tx buffer
* Added <TXATTEN:nn.n> <!TXATTEN:nn.n> tag to control
  transmit attenuator control from within a macro.
* Added <COMMENT:text> macro tag
  a do nothing that disappears from the transmitted
  text
```

4.1.2 Modem macro tags

Macro tags are also assigned to each supported modem type and sub-modem type that is supported by fldigi:

		Data Modems		
<MODEM:BPSK1000>	<MODEM:BPSK125>	<MODEM:BPSK250>	<MODEM:BPSK31>	<MODEM:BPSK500>
<MODEM:BPSK63>	<MODEM:BPSK63F>	<MODEM:CTSTIA-:1000:16>	<MODEM:CTSTIA-:1000:8>	<MODEM:CTSTIA:250-:8>
<MODEM:CTSTIA:500-:16>	<MODEM:CTSTIA:500-:8>	<MODEM:CTSTIA>	<MODEM:CW>	<MODEM:DOMEX4>
<MODEM:DOMEX5>	<MODEM:DOMEX8>	<MODEM:DOMX11>	<MODEM:DOMX16>	<MODEM:DOMX22>
<MODEM:DOMX44>	<MODEM:DOMX88>	<MODEM:FELDHELL>	<MODEM:FSKH105>	<MODEM:FSKHELL>
<MODEM:HELL80>	<MODEM:HELLX5>	<MODEM:HELLX9>	<MODEM:MFSK11>	<MODEM:MFSK128>
<MODEM:MFSK128L>	<MODEM:MFSK16>	<MODEM:MFSK22>	<MODEM:MFSK31>	<MODEM:MFSK32>
<MODEM:MFSK4>	<MODEM:MFSK64>	<MODEM:MFSK64L>	<MODEM:MFSK8>	<MODEM:MT63-1KL>
<MODEM:MT63-1KS>	<MODEM:MT63-2KL>	<MODEM:MT63-2KS>	<MODEM:MT63-500-L>	<MODEM:MT63-500-S>
<MODEM:NAVTEX>	<MODEM:Olivia-16-1-K>	<MODEM:Olivia-16-500>	<MODEM:Olivia-32-1-K>	<MODEM:Olivia-4-250>
<MODEM:Olivia-4-500>	<MODEM:Olivia-64-2-K>	<MODEM:Olivia-8-1K>	<MODEM:Olivia-8-250>	<MODEM:Olivia-8-500>
<MODEM:OLIVIA-:1000:32>	<MODEM:OLIVIA-:1000:8>	<MODEM:OLIVIA:250-:8>	<MODEM:OLIVIA:500-:16>	<MODEM:OLIVIA:500-:8>
<MODEM:OLIVIA>	<MODEM:PSK1000-C2>	<MODEM:PSK1000R>	<MODEM:PSK1000R-C2>	<MODEM:PSK125-C12>
<MODEM:PSK125R>	<MODEM:PSK125R-C10>	<MODEM:PSK125R-C12>	<MODEM:PSK125R-C16>	<MODEM:PSK125R-C4>
<MODEM:PSK125R-C5>	<MODEM:PSK250C6>	<MODEM:PSK250R>	<MODEM:PSK250R-C2>	<MODEM:PSK250R-C3>
<MODEM:PSK250R-C5>	<MODEM:PSK250R-C6>	<MODEM:PSK250R-C7>	<MODEM:PSK500C2>	<MODEM:PSK500C4>
<MODEM:PSK500R>	<MODEM:PSK500R-C2>	<MODEM:PSK500R-C3>	<MODEM:PSK500R-C4>	<MODEM:PSK63R-C10>
<MODEM:PSK63R-C20>	<MODEM:PSK63R-C32>	<MODEM:PSK63R-C4>	<MODEM:PSK63R-C5>	<MODEM:PSK800C2>
<MODEM:PSK800R-C2>	<MODEM:QPSK125>	<MODEM:QPSK250>	<MODEM:QPSK31>	<MODEM:QPSK500>
<MODEM:QPSK63>	<MODEM:RTTY:170-:45.45:5>	<MODEM:RTTY:170-:50:5>	<MODEM:RTTY:850-:75:5>	<MODEM:RTTY>
<MODEM:SITORB>	<MODEM:SLOWHEL-L>	<MODEM:THOR100>	<MODEM:THOR11>	<MODEM:THOR16>
<MODEM:THOR22>	<MODEM:THOR25x4>	<MODEM:THOR4>	<MODEM:THOR5>	<MODEM:THOR50x1>
<MODEM:THOR50x2>	<MODEM:THOR8>	<MODEM:THRBX1>	<MODEM:THRBX2>	<MODEM:THRBX4>
<MODEM:THROB1>	<MODEM:THROB2>	<MODEM:THROB4>	<MODEM:WEFA-X288>	<MODEM:WEFA-X576>

4.1.3 Other Modems

The following modems perform other functions not involving data transportation:

Non Data Modems	Description
<MODEM:NULL>	Null modem - Rx stream is discarded. Tx stream is silent but PTT is enabled
<MODEM:SSB>	SSB modem (place holder)
<MODEM:ANALYSIS>	Analyze Frequency

<MODEM:WWV>	Calibrate Sound Card

Local references are specified during the program configuration and can be changed during program operation.

Remote references are all part of the qso log field definitions and are routinely changed from contact to contact.

Global references are for items like Greenwich Mean Time.

The macros.mdf file can be edited with any ascii text editor such as kedit, gedit, geany, nano etc. But it is much easier to use the built-in macro editor provided in the program.

4.1.4 Macro Editor

Right click on any macro key (or the alternate set) and a macro editing dialog opens with the current copy of that macro and its label. This looks very similar to the DigiPan macro editor at the urging of Skip Teller, KH6TY.

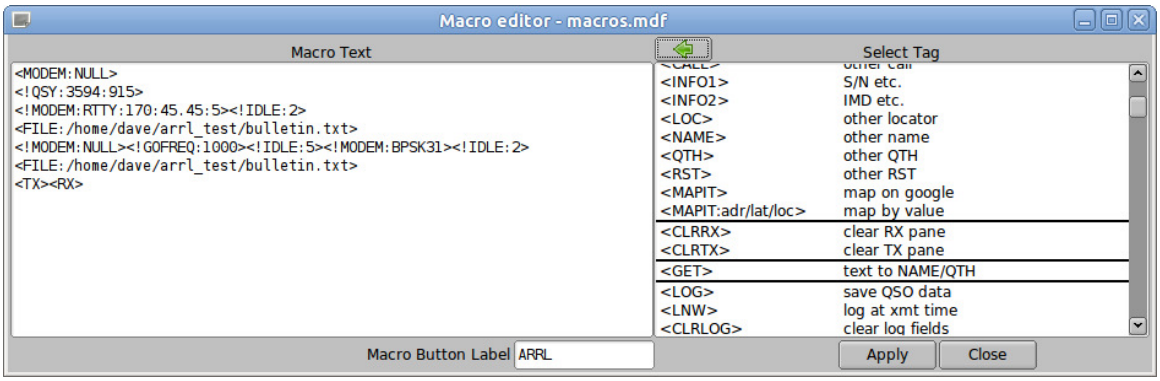


Figure 4.1: Macro Editor

The Text box is a mini-editor with a very limited set of control functions. You can mark, bound and select text for deletion (ctrl-X), copy (ctrl-C), and paste (ctrl-V). Marked text can also be deleted with the delete or the backspace keys. Marked text modification can also be invoked by using the mouse right click after highlighting.

The macro reference in the pick list can be transfered to the current editing cursor location. Highlight the desired macro reference and then press the double << arrow key for each occurance of the reference to be put into the macro text. You can change the label name but any more than 8 characters may exceed the width of the button for the default sized main dialog.

The <TIMER:NN> and <IDLE:NN> macro tags should have the NN replaced with the time interval in seconds.

<TX><IDLE:5>CQ CQ de <MYCALL> <MYCALL> k<RX><TIMER:20>

- will enable the PTT
- cause 5 seconds of idle signal
- send the CQ CQ de W1HKJ W1HKJ k
- disable PTT
- and count down 20 seconds before repeating the macro
- after sending the text the count down timer button (upper right hand corner of main dialog) will display the current timer value in seconds. Press this button to disable the timer.
- the timer be disabled if the Escape key is pressed, the T/R is pressed, and macro key is pressed, or if a callsign is copied from the Rx text area to the callsign logbook entry.
- the time will be disabled if any mouse activity occurs in the waterfall control.

The label associated with each macro key can be individually annotated with a symbol. Here are the symbols that are recognized by the button label drawing routine:

4.1.5 Macro Display Symbols

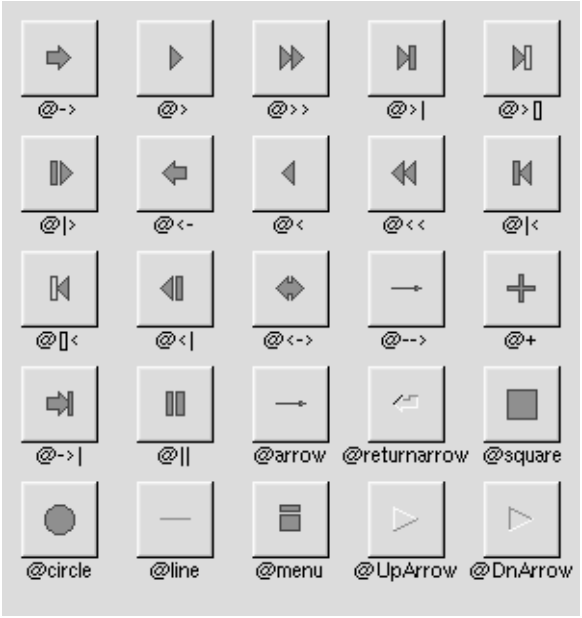


Figure 4.2: Symbols

The @ sign may also be followed by the following optional "formatting" characters, in this order:

- '#' forces square scaling, rather than distortion to the widget's shape.
- +[1-9] or -[1-9] tweaks the scaling a little bigger or smaller.
- '$' flips the symbol horizontally, '' flips it vertically.
- [0-9] - rotates by a multiple of 45 degrees. '5' and '6' do no rotation while the others point in the direction of that key on a numeric keypad. '0', followed by four more digits rotates the symbol by that amount in degrees.

Thus, to show a very large arrow pointing downward you would use the label string "@+92->".

Here are my macro buttons suitably annotated:

Figure 4.3: Macro Buttons

There are 4 sets of 12 macro functions. You can move between the 4 sets using the keyboard and the mouse.

1. Left click on the "1" button to move to set #2. Right click on the "1" button to move to set #4.
2. Move the mouse to anywhere on the macro buttons. Use the scroll wheel to move forward & backward through the macro sets
3. Press the Alt-1, Alt-2, Alt-3 or Alt-4 to immediately change to that macro set.

- The label for CQ is "CQ @>|", denoting that both <TX> and <RX> are present in the macro text.
- The label for QSO is "QSO @>>", denoting that only <TX> is present in the macro text.
- The label for KN is "KN @||", denoting that only <RX> is present in the macro text.

You could use any label that is symbolic to the function required. Refer to the FLTK web site for a full list of label types.

If you modify the macros and do not save them ("Files/Save Macros" on the main window) fldigi will prompt you to save the macros when you exit the program if you have the Exit Prompts option selected.

4.1.6 Contest macro tags

Refer to Contest-How-To

Return to Top of Page

Return to Main Page

4.2 Inline Macros

The following macro tags will be parsed and acted upon then they occur during the transmission of the macro.

Tag	Description
<!WPM:NN>	CW words per minute
<!POST:+/-nn.n>	CW post delay
<!PRE:nn.n>	CW pre delay
<RISE:nn.n><RISE:nn.n>	CW rise/decay time
<!MODEM: ,...>	Change to specified modem with parameters
<!GOHOME>	Move audio carrier to mode sweet spot frequency
<!GOFREQ:NNNN>	Move audio carrier to specific audio frequency
<!QSY:FFF.F[:NNNN]>	Move to specific RF and Audio frequency
<!IDLE:NN.nn>	Transmit idle signal for specified number of seconds
<!WAIT:NN>	Wait (no audio) for specified number of seconds
<!WPM:ww[:ff]>	CW - ww = WPM, optional ff = Farnsworth wpm

Note that each of these tags is identical to their immediate mode counterparts. The exception is the addition of the exclamation mark following the leading '<'. The use of these tags is best explained by example.

4.2.1 CW Code Practice Transmission

```
<MODEM:CW><TX>
<!GOFREQ:600><!WPM:10:15>
NOW IS THE TIME - now 180 wpm
<!IDLE:2><!RISE:1.0><!PRE:0.4><!POST:+0.2><!WPM:180>
FOR ALL GOOD MEN TO COME TO THE AID of their country.
now 30 wpm<!IDLE:2>
<!WPM:30>de <MYCALL> k
<RX>
```

- Modem type is changed to CW before start of transmission (<MODEM:CW>) does not have the addition of the ! symbol
- Audio carrier is changed to 600 Hertz
- Transmit WPM set to 30 words per minute, followed by text
- Transmit is silent for 2 seconds
- Rise time, pre and post delays are adjusted and the WPM changed to 180 words per minute
- Text is sent at 180 WPM
- WPM changed to 30 words per minute, followed by identification string and signoff
- Return to receive

This is a more complex macro that might be used for a code practice transmission such as the W1AW broadcast on 3580 KHz.

```
<MODEM:NULL>
<!QSY:3579.200:800>
<TX>
<!MODEM:CW>
<!WPM:5:15><!IDLE:2><FILE:/home/dave/fldigi.ft950/scripts/practice-5wpm.txt>
<!WPM:10:15><!IDLE:5><FILE:/home/dave/fldigi.ft950/scripts/practice-10wpm.txt>
<!WPM:15:15><!IDLE:5><FILE:/home/dave/fldigi.ft950/scripts/practice-15wpm.txt>
<!WPM:18><!IDLE:10><FILE:/home/dave/fldigi.ft950/scripts/bulletin.txt>
<!IDLE:5>end of broadcast de <MYCALL> k
<RX>
```

4.2.2 QSY:ffff.f:aaaa test

```
<MODEM:NULL><TX>
<!QSY:3583.0:1750>
<!MODEM:RTTY:170:45.45:5>
RYRYRYRYRYRY de <MYCALL> k
<RX>
```

- Change modem type to NULL to suppress sending any signal when transmit enabled. This is necessary to allow the transmit data stream to be processed.
- Send frequency change command to transceiver, new frequency is 3583.0 kHz. Set audio frequency to 1750 Hz
- Change modem to RTTY, 170 Hz shift, 45.45 Baud, 5 Bit code (standard Baudot)
- Send RY... text
- Return to receive

4.2.3 Advanced QSY operations

Several QSY frequencies or ranges of frequencies provided by one increment, can be proposed, in which case the first frequency after the current frequency is chosen.

Advanced QSY operations

4.2.4 ARRL style broadcast in multiple modes

```
<MODEM:NULL>
<!QSY:3594:915>
<!MODEM:RTTY:170:45.45:5><!IDLE:2>
<FILE:/home/dave/arrl_test/bulletin.txt>
<!MODEM:NULL><!GOFREQ:1000><!IDLE:5><!MODEM:BPSK31><!IDLE:2>
<FILE:/home/dave/arrl_test/bulletin.txt>
<TX><RX>
```

- Change modem type to NULL to suppress transmission and start the Tx data stream processor
- Send frequency change command to transceiver, new frequency is 3594 kHz, audio frequency 915 Hz. Note that this puts the MARK tone at 1000 Hz.
- Change modem to RTTY 170 Hz shift, 45.45 baud (default is 5 bits). Idle for 2 seconds.
- Send the file contents of specified file
- Change to NULL modem. Turns off the RTTY diddle ... audio stream is silent.
- Change audio frequency to 1000 Hz. Idle for 5 seconds (complete silence of transmit)

- Change modem type to BPSK-31. Send BPSK idle for 2 seconds
- Send the file contents of specified file
- Return to receive (note that the <TX> can appear anywhere is the macro definition as it is executed when the macro text is parsed). <RX> is always moved to the end of the Tx buffer. It could also have appeared anywhere in the macro string.

4.2.5 Appearance of tx buffer

```
^!
^!^!
QST de W1HKJ
Test bulletin for 9/7/2011
QST de W1HKJ SK

^!^!^!^!^!
QST de W1HKJ
Test bulletin for 9/7/2011
QST de W1HKJ SK

^r
```

This is the contents of the ARRL broadcast macro text at the time the macro button is pressed. Each of the <!... macro tags has been specified by the string "^!" which the Tx processor interprets as the command to process the top most tag in the first-in, first-out sequence of tags. As each "^!" is executed the referenced tag is printed to the Rx buffer using color coded text.

```
<!QSY:3594:915>
<!MODEM:RTTY:170:45.45:5>
<!IDLE:2>

QST DE W1HKJ
TEST BULLETIN FOR 9/7/2011
QST DE W1HKJ SK

<!MODEM:NULL>
<!GOFREQ:1000>
<!IDLE:5>
<!MODEM:BPSK31>
<!IDLE:2>

QST de W1HKJ
Test bulletin for 9/7/2011
QST de W1HKJ SK
```

Return to Top of Page

Return to Main Page

4.3 Advanced QSY operations

The<QSY:FFF.F[:NNNN]> macro tag and its delayed flavour<!QSY:FFF.F[:NNNN]> allows the user to set the transceiver frequency to FFF.F (In kHz) and optionaly set the audio frequency to NNNN (In Hz).

It is possible to give several frequencies, which specify a frequency set. When the macro is executed, it choses the first frequency of this set greater than the current one. If the current frequency is greater than any frequency of the set, then the macro choses the smallest, first frequency of this set.

The utility of this feature is to allow a scan of a range of frequencies by re-executing the same macro over and over. Each time the macro is executed, the next frequency of the set is selected. It can typically be used to iteratively try several frequency for a given test.

The frequency set can be specified in two distinct ways, which can be freely combined together.

4.3.1 Distinct frequencies

Several distinct separated with a semi-column, in increasing order. For example:

<QSY:2616.6;3287.6;3853.1;4608.1;4780.1;7878.1;8038.1:1900>

It means that the transceiver frequency will be set to 2616.6 kHz, then at next execution to 3853.1 kHz etc... and will loop back to 2616.6 kHz. Each time the same (optional) audio frequency will be set to 1900 Hz. In this specific case, it allows to test several Weather Fax broadcast stations, until the operator finds an active one.

4.3.2 Frequencies must be increasing

A frequency can come with an increment: This means an implicit range of frequencies from this one to the next frequency. If the last frequency has an increment, it is never taken into account: The last frequency is always an upper limit. Let's consider this example:

<QSY:89000+100;102000>

It is equivalent to:

<QSY:89000;89100;89200;89300;...;101900;102000>

In this case, it allows fldigi to scan all FM frequencies by clicking the macro button.

4.3.3 Combination with <TIMER> macro tag.

If the macro is automatically reexecuted using the <TIMER>, the same logic applies. At each run of the macro, the next frequency is chosen. The following macro transmits the same message on the frequencies 144800 MHz, 144900 ... until 146000 then loops back, waiting five seconds between each transmission.

<TX><QSY:144800.00+100;146000><MODEM:NULL><!MODEM:PACKET>

<FREQ> CQ CQ de <MYCALL><RX><TIMER:5>

4.3.4 Execution errors.

Several error messages can be displayed in the macro editor in case of a parameter. If this happens, the execution of the macro is stopped. Here is the list of possible messages:

4.3.5 Invalid frequency range

There must be valid frequencies. This macro will display the message:

<QSY:abcdef>

4.3.6 Increment must be positive

The frequency increment must be positive. This macro will fail:

<QSY:89000-1000;88000>

4.3.7 Frequency not positive

All frequencies must be strictly positive.

4.3.8 Frequencies must be increasing

The sequence of frequencies must be strictly increasing. Thus, this error message can appear with a macro such as:

<QSY:89000;88000>

4.3.9 Inline Macro Tags

Unless otherwise noted all of the macro tags discussed thus far are meant to be executed at the time that the macro is invoked by the button closure. It is possible by execution of the tag to be delated until it appears in the transmit data stream. This delayed execution capability is limited to a select number of tags. See Inline Macro Tags for a list of these tags and examples.

Return to Top of Page

Return to Main Page

4.4 Exec Macro

The <EXEC> ... </EXEC> macro is designed to be used on the Linux OS as it supports fully functional pipes. Windows' version of file pipes is not fully POSIX compliant, but the function might work in the environment. Consider all that the following allows you to do from within fldigi and you might want to consider changing over to Linux.The <EXEC> macro defines an external child process (or processes) that will be called by fldigi when the macro key is invoked.

4.4.1 Exported variables

Fldigi exports a set of variables to the child process and adds ~/.fldigi/scripts to the PATH variable before running the shell code. This is the directory location for all executable scripts and programs which you might want to call from within the macro. Some examples will be given later. Open the macro editor for an undefined macro key and enter the following:

```
<EXEC>env | grep FLDIGI</EXEC>
```

Save the macro; call it ENV. Then press the newly defined macro key. All of the exported variables will be shown in the transmit window.

Here is an example of the results:

FLDIGI_RX_IPC_KEY=9876

FLDIGI_LOG_LOCATOR=FM02BT

FLDIGI_TX_IPC_KEY=6789

FLDIGI_LOG_RST_IN=

FLDIGI_LOG_FREQUENCY=3581.000

FLDIGI_AZ=108

FLDIGI_MY_CALL=W1HKJ

FLDIGI_LOG_TIME=2113

FLDIGI_MY_NAME=Dave

FLDIGI_VERSION=3.0preG

FLDIGI_LOG_NOTES=

FLDIGI_LOG_QTH=Mt Pleasant, SC

FLDIGI_MY_LOCATOR=EM64qv

FLDIGI_DIAL_FREQUENCY=3580000

FLDIGI_CONFIG_DIR=/home/dave/.fldigi/

FLDIGI_LOG_RST_OUT=

FLDIGI_MODEM=BPSK31

FLDIGI_LOG_CALL=KH6TY

FLDIGI_MODEM_LONG_NAME=BPSK-31

FLDIGI_AUDIO_FREQUENCY=1000

FLDIGI_LOG_NAME=Skip

FLDIGI_PID=14600

FLDIGI_FREQUENCY=3581000

All of the above envelope variables can be referenced in a shell script that is called from within fldigi.

4.4.2 Detection of existing scripts

In anticipation of a collection of useful "fldigi scripts", the macro browser contains a <exec> </exec> macro line for each executable file found in the scripts directory. The EXEC macro allows the text that is read from the child process to be parsed for more fldigi macros. For example, try this macro:

```
<EXEC>cat foo</EXEC>
```

where foo is a file that contains:

```
<MYCALL>
```

This may have some interesting uses but, if it is undesirable, it can be suppressed with an extra layer of redirection. Instead of <EXEC>command</EXEC>, you would use <EXEC>noexp command</EXEC> where noexp is the following very simple script:

```
snip---------------------------------------
#!/bin/bash
echo -n "<STOP>"
"$@"     # run the command
r=$?     # save its exit code
echo -n "<CONT>"
exit $?
snip---------------------------------------
```

There are three additional MACRO definitions that expand the capability of the <EXEC> command: <STOP>, <CONT> and <GET>. The <STOP> and <CONT> macros stop and resume the expansion of all <MACRO> strings. For example, <STOP><MYCALL><CONT><MYCALL> would only expand the second <MYCALL>.

By wrapping the command output in this way we can be sure that no text will be expanded. You might even use

```
"$@" | sed "s/<CONT>//g"
```

if you feel paranoid. You can "fork and forget" with an exec macro defined as: <EXEC>exec command -args >/dev/null</EXEC>

Any of the text that appears between the <EXEC> and </EXEC> can reference an executable program or shell command found in the ~/.fldigi/scripts directory.

Any text output that is returned by the program or script program (or the result of the in-line command) is always returned to the transmit buffer and appears as appended to the transmit window.

4.4.3 Querying an external database

The <GET> command captures returned text from the external process and parses it for the following content:

$NAMEtext_name$QTHtext_qth

If either $NAME or $QTH is present the trailing text is transferred to the LOG_NAME or LOG_QTH widgets respectively. This means that you can create a script that accesses a local or net based database of callsign data and parse that data to form the above console output. Fldigi will accept that output, parse it and populate the associated log entries. Cool! Now for some examples. Here is a perl script that performs the above for the University of Arkansas on-line callsign database, ualr-telnet. The matching macro key definition for the above is:

```
<EXEC>ualr-telnet.pl $FLDIGI_LOG_CALL</EXEC><GET>
```

which I named "ualr ?"

4.4.4 Google Earth Map

Here is a really cool perl script, Google Earth Mapping, that accepts the current "Loc" field in the logging area and generates a Google Earth map which is displayed in your default browser.

The macro call is:

```
<EXEC>map.pl</EXEC>
```

4.4.5 Custom dates/times

You can use <EXEC> to create custom date/time entries. For example, BARTG contesters use HM, but in other circumstances a user might prefer H:M or H.M etc. Create the following script file in the ~/.fldigi/scripts directory, call it mytime:

```
snip---------------------------------------
#!/bin/sh
date --utc "+%H:%M"
snip---------------------------------------
```

date calls strftime, the same C function used by fldigi for the ZDT/LDT expansion, so it has an equally vast number of format strings to choose from. Look for them in its manual page.

Give "mytime" execute permissions with a file manager or with chmod: chmod u+x ~/.fldigi/scripts/mytime.

Test it on the command line and make sure it works correctly: ~/.fldigi/scripts/mytime

Restart fldigi. The mytime script will now appear at the end of the list in the macro browser, and can be entered with the << button as usual. Test that macro and you will see that <EXEC>mytime</EXEC> inserts the datetime in the specified format. Of course you could have entered:

```
<EXEC>date --utc "+%H:%M"</EXEC>
```

in the macro body text directly Many other uses for the <EXEC>...</EXEC> macro pair can be imagined when used the with ENV parameters. For example you could send Azimuth data to an automated antenna rotor. The exported variables should be sufficient for a script writer to create custom loggers and clients.

Return to Top of Page

Return to Main Page

Chapter 5

Modems

5.1 Contestia

Fldigi can operate on the following Contestia modes without special setup by the operator:

Mode	Symbole Rate (Baud)	Typing speed (WPM)	Bandwidth (Hz)
Contestia 4-250	62.5	∼ 40	250
Contestia 8-250	31.25	∼ 30	250
Contestia 4-500	125	∼ 78	500
Contestia 8-500	62.5	∼ 60	500
Contestia 16-500	31.25	∼ 30	500

Contestia 8-1000	125	∼ 117	1000
Contestia 16-1000	62.5	∼ 78	1000
Contestia 32-1000	31.25	∼ 48	1000

Unusual combinations of symbol rate and bandwidth can be selected using the Contestia Configuration tab.

Contestia is a digital mode directly derived from Olivia that is not quite as robust - but more of a compromise between speed and performance. It was developed by Nick Fedoseev, UT2UZ, in 2005. It sounds almost identical to Olivia, can be configured in as many ways, but has essentially twice the speed.

Contestia has 40 formats just like Olivia - some of which are considered standard and they all have different characteristics. The formats vary in bandwidth (125,250,500,1000, and 2000hz) and number of tones used (2,4,8,16,32,64,128, or 256). The standard Contestia formats (bandwidth/tones) are 125/4, 250/8, 500/16, 1000/32, and 2000/64. The most commonly used formats right now seem to be 250/8, 500/16, and 1000/32.

Contestia performs very well under weak signal conditions. It handles QRM, QRN, and QSB very well also. It decodes below the noise level but Olivia still outperforms it in this area by about 1.5 - 3db depending on configuration.

It is twice as fast as Olivia per configuration. It is an excellent weak signal, ragchew, QRP, and DX digital mode. When ragchewing under fair or better conditions it can be more preferable to many hams than Olivia because of the faster speed. For contests it might also be a good mode IF the even faster configurations such as 1000/8 or 500/4 are used.

Contestia get it's increased speed by using a smaller symbol block size (32) than Olivia (64) and by a using 6bit decimal character set rather than 7bit ASCII set that Olivia does. Therefore, it has a reduced character set and does not print out in both upper and lower case (like RTTY). Some traffic nets might not want to use this mode because it does not support upper and lower case characters and extended characters found in many documents and messages. For normal digital chats and ham communications that does not pose any problem.

Return to Top of Page

Return to Main Page

5.2 CW

5.2.1 Carrier Frequency

Fldigi generates CW by inserting a keyed tone at the current waterfall audio frequency. The CW carrier frequency is the USB carrier + the audio frequency, or the LSB carrier - the audio frequency. If fldigi is tracking and receiving a CW signal on the waterfall your transmitted signal will be exactly on the frequency of the other operator. You probably cannot use your transceivers CW filter unless that filter can be used with the SSB mode.

5.2.2 QSK and the TAB KEY

If you are operating QSK with a separate transmitter / receiver you can very quickly stop your transmit signal with the TAB key. In the CW mode only the TAB key causes the program to skip over the remaining text in the transmit text buffer. The text that is skipped will be color coded blue. The program remains in the transmit mode (PTT enabled), but since the buffer is now empty no A2 CW signal is generated. Code transmission will then restart with the very next keyboard closure of a valid CW character.

5.2.3 Pausing transmit

The Pause/Break momentarily key stops sending text. Pressing it again resumes transmission.

5.2.4 Aborting transmit

The Escape key is used to immediately stop text transmission. The Tx buffer is cleared.

5.2.5 WPM adjustment

In CW mode the status bar is changed to include a transmit WPM adjuster. Use the arrow buttons or the mouse. Mouse wheel up/down changes transmit WPM by +/- 1. Hold the shift and mouse mouse wheel changes transmit WPM by +/- 10. The "∗" button immediately to the right of the WPM adjuster is used to toggle between the current and the default transmit WPM.

The transmit WPM can also be adjusted with three hot keys:

- Numeric keypad "+" increases the transmit WPM by 1
- Numeric keypad "-" decreases the transmit WPM by 1
- Numeric keypad "∗" toggles between the selected transmit WPM and the default transmit WPM

The "Default" control on the CW tab sets that default value. If during a QSO you needed to slow down to give the other op a better chance to copy what you are sending, just hit the "∗" on the numeric keypad and the CW code will immediately switch to sending CW at the set default value (18 wpm in this example). Press the "∗" again to return to back to the CW speed that you were previously using.

Each time the transmit WPM is changed the receive decoder WPM tracking is reset to the new transmit WPM. This allows you to quickly force the decoder to a new WPM range.

Figure 5.1: The Rx and Tx WPM are shown in the status bar.

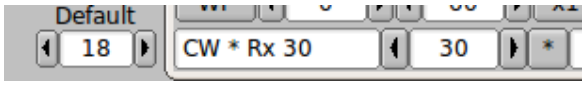


Figure 5.2: ∗ indicates that default WPM is selected

5.2.6 Farnsworth keying

You might want to use farnsworth keying to provide a character rate that is faster than the word per minute rate.

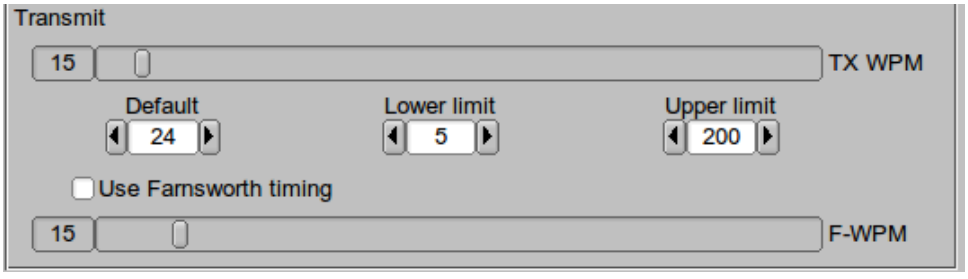


Figure 5.3: Farnworth Timing

Set the F-WPM slider to tthe character rate and enable the Use Farnsworth timing check box. When the TX WPM is set to a speed below the Farnsworth value then the character rate will be at the Farnsworth setting and the word rate will be at the TX WPM rate. You can also set the TX WPM and F-WPM from within a Macros.

5.2.7 CW configuration

The CW Configuration is easily reached from the Config menu or by right clicking on the left most entry in the status bar (CW).

5.2.8 Prosigns

The prosigns are configurable (see CW Configuration). The defaults are:

PROSIGN	KEYBOARD	DISPLAYED AS
BT	=	<BT>
AA	~	<AA>
AR	>	<AR>
AS	<	<AS>
HM	{	<HM>
INT	&	<INT>
SK	%	<SK>
KN	+	<KN>
VE	}	<VE>

Return to Top of Page

Return to Main Page

5.3 DominoEX

The modem code for dominoEX uses a wide band multiple frequency detector that can lock on and detect the incoming signal even when badly mistuned. Frequency domain oversampling is used to allow proper tone detection without the need for AFC. The AFC control does not alter the decoder in any way.

The waterfall and digiscope will appear as:

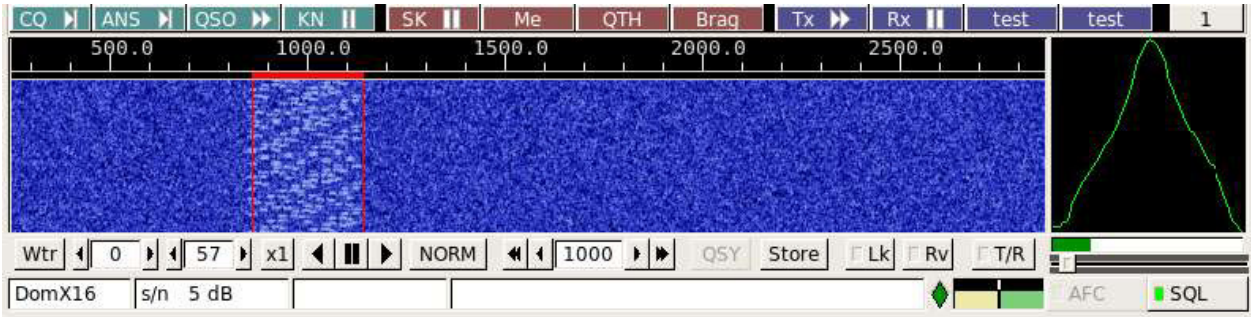


Figure 5.4: DominoEX

The text displayed in the status area is the secondary text being sent by the transmitting station. When the keyboard buffer is empty the dominoEX modem transmits text from the secondary text buffer. Your secondary text buffer can be edited on the DominoEX configuration tab.

The digiscope display represents the tone pairs moving through the tone filters. You can also use an alternate digiscope display (left click on the digiscope display area).

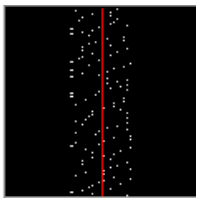


Figure 5.5: DominoEX

In this display mode the red line represents the center of the multiple tone bins that are in the detector. The dots will be blurry if the AFC is not locked on and become very distinct when AFC lock has been achieved. The tone dots will move from bottom to top (opposite the direction of the waterfall).

This is the same signal mistuned:

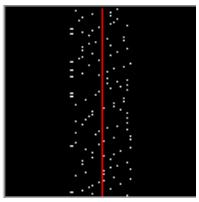


Figure 5.6: DominoEX Mistuned

and with the signal badly mistuned:

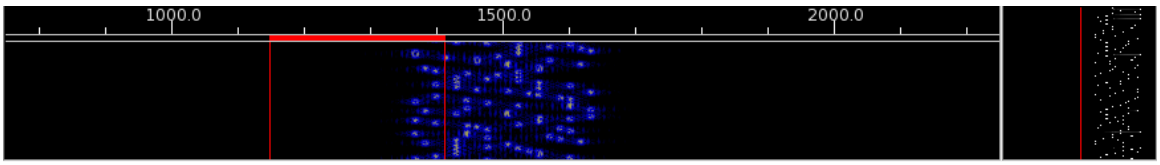


Figure 5.7: DominoEX Mistuned

See DominoEX Configuration.

Return to Top of Page

Return to Main Page

5.4 Hellschreiber

5.4.1 Hellschreiber modem

All Hellschreiber modes are based on character scanning, reproducing characters in a similar way to a dot-matrix printer. This technique uses a digital transmission, yet allows the received result to be interpreted by eye, a similar concept to the reception of Morse by ear. The character is scanned upwards, then left to right. There are typically 14 pixels (transmitted dot elements) per column (although single pixels are never transmitted) and up to seven columns per character including inter-character space.

These remarkably simple modes are easy to use, easy to tune, and although not especially sensitive, are entirely suited to HF/VHF since they use no sync and the eye can discern the text even in high levels of noise. fldigi can operate in the following :

5.4.2 Hellschreiber modes

Mode	Symbol Rate	Typing Speed	Duty Cycle	Bandwidth
Feld-Hell	122.5 baud	~ 2.5 cps (25 wpm)	~ 22%	350 Hz
Slow Hell	14 baud	~ 0.28 cps (2.8 wpm)	~ 22%	40 Hz
Feld-Hell X5	612.5 baud	~ 12.5 cps (125 wpm)	~ 22%	1750 Hz
Feld-Hell X9	1102.5 baud	~ 22.5 cps (225 wpm)	~ 22%	3150 Hz
FSK-Hell	245 baud	~ 2.5 cps (25 wpm)	~ 80%	490 Hz
FSK-Hell 105	105 baud	~ 2.5 cps (25 wpm)	~ 80%	210 Hz
Hell 80	245 baud	~ 5.0 cps (50 wpm)	100%	800 Hz

5.4.3 Hellschreiber Waterfall

Feld-Hell look like this when being received by fldigi:

Figure 5.8: Feld-Hell, Slow Hell, Feld-Hell X5, and Feld-Hell X9 are all pulse

Feld-Hell seems to be the most commonly used and use can usually be found on 80 and 40 meters at the high end of the digital sub bands. Extreme linearity is required in the transmit path in order to control the bandwidth of the transmitted signal. Feld-Hell X5, Feld-Hell X9 and Hell 80 should probably not be used on HF in the US. They can be used on VHF and UHF.

Return to Top of Page

Return to Main Page

5.5 MFSK

MFSK16 and MFSK8 are multi-frequency shift keyed (MFSK) modes with low symbol rate. A single carrier of constant amplitude is stepped (between 16 or 32 tone frequencies respectively) in a constant phase manner. As a result, no unwanted sidebands are generated, and no special amplifier linearity requirements are necessary. The tones selected are set by the transmitted (4 or 5 bit) bit pattern and a gray-code table.

The mode has full-time Forward Error Correction, so it is very robust. Tuning must be very accurate, and the software will not tolerate differences between transmit and receive frequency. The mode was designed for long path HF DX, and due to its great sensitivity is one of the best for long distance QSOs and skeds. MFSK8 has improved sensitivity, but is very difficult to tune, and suffers more from Doppler. It is useful as the band fades out.

MFSK-32 and MFSK-64 are high baud rate and wide bandwidth modes designed for use on VHF and UHF. These are very useful for send large documents or files when some transmission errors are can be tolerated.

This is an example of properly tuned MFSK16 signal with a s/n of approximately 9 dB.

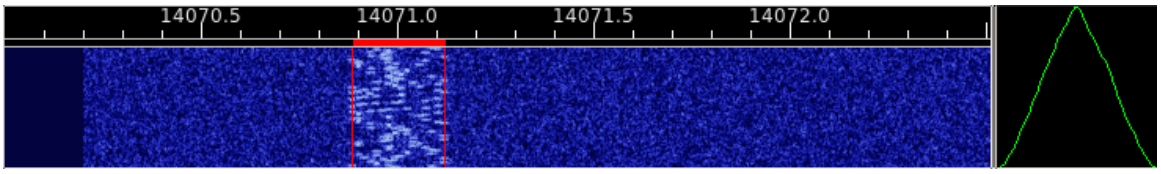


Figure 5.9: MFSK16 signal

The same signal viewed with the waterfall expanded to the x2 factor.

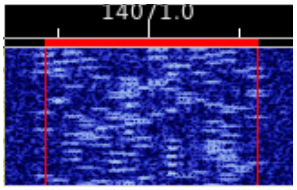


Figure 5.10: MFSK16 signal

5.5.1 MFSK Picture Mode

Fldigi can send and receive images using all MFSK baud rates. When operating with other modem programs you should limit sending pictures to the MFSK-16 baud rate. The program can send and receive MFSK images in both black and white and in 24 bit color. The transmission mode for MFSKpic is similar to FAX.

Reception of an MFSKpic transmission is fully automatic. The MFSKpic transmission has a preamble sent which will be visible on the text screen. The preamble reads as "Pic:WWWxHHH;" or "Pic:WWWxHHHC;" for b/w or color respectively. The WWW and HHH are numbers specifying the width and height of the picture in pixels.

The successful reception of a MFSKpic is highly dependent on s/n conditions. The data is transmitted as an FM modulated signal and is subject to burst and phase noise on the transmission path. It can provide excellent photo transmission on a really good path.

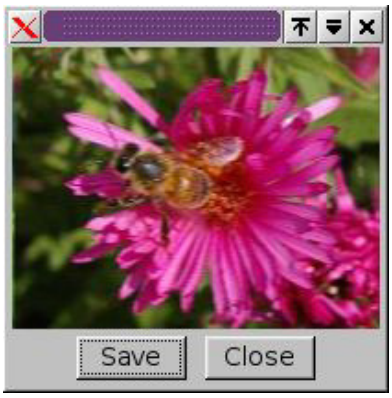


Figure 5.11: Received MFSK Image

This is an example of a photo received on a bench test. The received image is an exact replica of the transmitted image. The color depth is a full 24 bits.

Images should be carefully selected for size before beginning a transmission. To calculate the transmit time for an image use the following formula:

Time(sec) = W $*$ H / 1000 for black and white

Time(sec) = W $*$ H $*$ 3 / 1000 for color

Where the W and H are the dimensions of the photo in pixels. A 200 x 200 image will take 120 seconds to transmit in color and 40 seconds to transmit in b/w. The symbol rate for this mode is 1000 data bytes per second. The color image consists of 3 bytes; red, blue and green for each pixel.

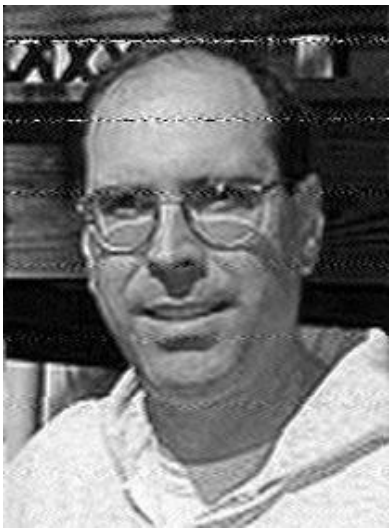


Figure 5.12: Picture received from K0OG

This is an example of a picture received live on 80 meters (thanks K0OG)

Received images are saved in the default folder $HOME/.fldigi/images (Linux) or <defaultpath>/fldigi.files/images (Windows).

5.5.2 Transmitting an Image

Figure 5.13: Xmit Picture Dialog box

You can only transmit an image while in the MFSK-16 mode. The image can be prepared for transmission while in the receive mode. Right click in the transmit text box and select "Send Image" from the popup menu. This will open up the transmit image dialog which will be blank to start.

Press the "Load" button and a file selection dialog will allow you to select a suitable image for transmit. The file selection dialog also has a preview capability so you will see what the image looks like.

You may also open a window manager file browser and drag and drop an image to the center part of the ***Send image*** dialog.

The "X1" button is a three-way toggle that allows you to transmit an image file in

X1 - normal and compatible with other modem programs

X2 - double speed, and

X4 - quadruple speed. X2 and X4 are fldigi specific image modes.

Figure 5.14: Xmit Picture Dialog Box with Image

The Send image dialog after the image was drag and dropped onto the dialog.

The properties box said this image was 120 x 119 24 bit color. So it should take 42.8 seconds to transmit in full color. You can send a color or a b/w image in either color mode or b/w mode. If you transmit a color image in b/w the program will convert the image before transmitting. If you transmit a b/w image as full color you are in effect transmitting redundant information, but it can be done. I selected the "XmtClr" button for a trial run. Pressing either the "XmtClr" or "XmtGry" will put the program and the transceiver into the transmit mode if it was in the receive mode. The image is cleared and then repainted as the transmission proceeds. You see the same image progression

that the receiving station should see. The main display also displays the % completion on the status bar. Hold the mouse over either the XmtClr or the XmtGry button and the tooltip will tell you the transmit time for this image.

You may abort the transmission at any time by pressing the "Abort Xmt" button. That will return you to the text mode for MFSK. You will then have to toggle the T/R button if you want to return to receive.

Figure 5.15: Received MFSK Image

The receiving program decodes the "Pic:110x119C;" as a color picture 110 wide by 119 high. Here is shown being received on a computer running Vista Home Premium.

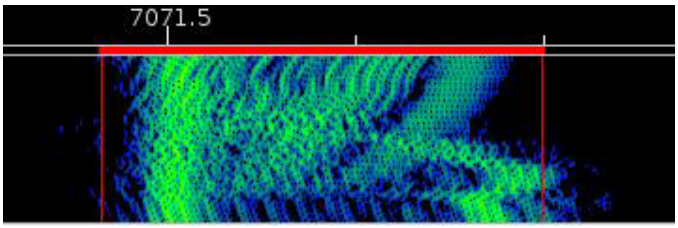


Figure 5.16: Waterfall of a MFSK Image

This is what the waterfall will look like during the reception of an MFSK-16 image.

The actual spectrum signature will vary with the image bytes being transmitted. The waterfall scale is in the x4 mode and the above photo was being transmitted in 24 bit color for this screenshot. The waterfall clearly shows that the image transmission is within the bandwidth occupied by MFSK-16.

5.5.3 Picture with a slant

If either the send, receive or both ends of the transmission are using an uncalibrated sound card whose sampling rate is not an exact multiple of 8000 Hz the resulting picture at the receive end will appear slanted. The degree of slant is directly related to the accumulation of the frequency error at both ends of the transfer. Stations wishing to send and receive MFSK pic's should calibrate their sound card. The WWV calibration mode is used to measure and set the parts per million (ppm) correction factor for the sound card.

Your sound system may be fully corrected, but the sending station may have an uncorrected sound card. You can usually correct for small errors in the following way. After the full picture is received move the mouse to bottom left or right corner of the slanted images (the corner that clearly visible). Then left click on that corner. The program will correct for the slant. The correction will not be perfect but it may help to make the image more viewable.

Return to Top of Page

Return to Main Page

5.6 MT63

MT63 is an Orthogonal Frequency Division Multiplexed mode consisting of 64 parallel carriers each carrying part of the transmitted signal. The tones are differential BPSK modulated. MT63 employs a unique highly redundant Forward Error Correction system which contributes to it robustness in the face of interference and facing. The tones have synchronous symbols, and are raised cosine moduled. This mode requires a very linear transmitter. Over-driving leads to excessive bandwidth and poorer reception.

The mode is very tolerant of tuning and fldigi will handle as much as 100 Hz of mistuning. This is very important since MT63 is often used in very low Signal to Noise ratios. There are three standard modes:

Mode	Symbol Rate	Typing Speed	Bandwidth
MT63-500	5.0 baud	5.0 cps (50 wpm)	500 Hz
MT63-1000	10.0 baud	10.0 cps (100 wpm)	1000 Hz
MT63-2000	20 baud	20.0 cps (200 wpm)	2000 Hz

In addition there are two interleaver options (short and long) which can be set on the MT63 configuration tab. The default calling mode is MT63-1000. If the short interleaver is used then one can expect some compromise in robustness. The long interleaver results in somewhat excessive latency (delay between overs) for keyboard chatting. MT63-1000 with the long interleaver has a latency of 12.8 seconds.

You can change from receive to transmit immediately upon seeing the other stations signal disappear from the waterfall. You do not need to wait until the receive text completes. Any remaining data in the interleaver will be flushed and the associated receive text printed quickly to the Rx pane. Tx will commence right after the buffer is flushed.

MT63 may be operated in the default fixed audio frequency mode. In this mode you are not allowed to randomly place of the signal on the waterfall. Your transmit signal, and also the received signal should be centered at 750 Hz for MT63-500, 1000 Hz for MT63-1000, and 1500 Hz for MT63-2000. If you click on the waterfall to move the tracking point it will be restored to the required position.

The default mode, MT63-1000, looks like this on fldigi's waterfall.

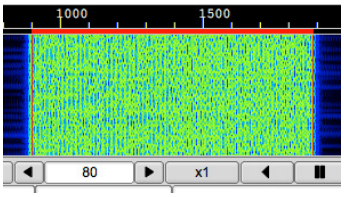


Figure 5.17: MT63-1000

You can also elect to operate the MT63 modem in a "manual tune" mode (MT63 configuration tab). The manual tune allows you to place both the Rx and the Tx signal to be anywhere within the confines of your SSB bandwidth. This screen shot shows this capability:

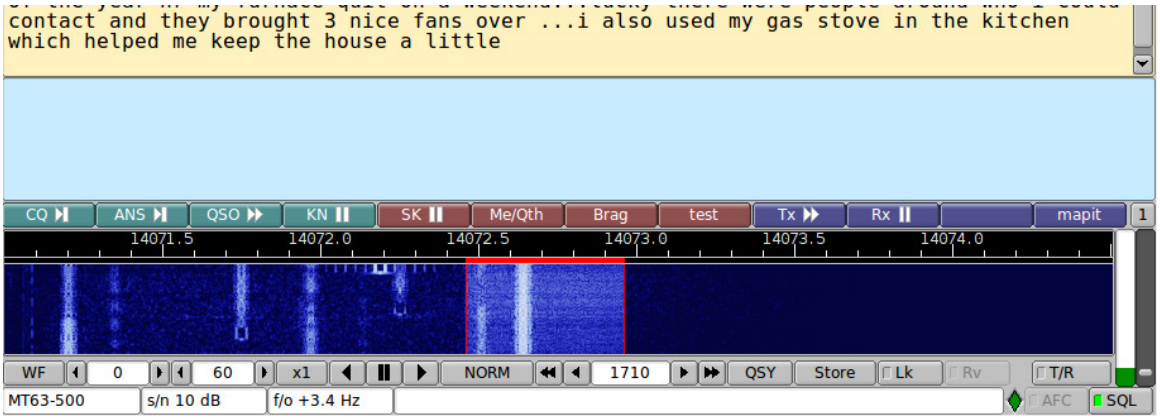


Figure 5.18: MT63-500 with QRM

This view also demonstrates how immune MT63 is to interference. The multiple PSK31 signals that appear on top of the MT63 signal did not degrade the decoder. MT63 is usually used above 14073 MHz to avoid the possibility of this type of mode conflict.

Edited excerpts from Pawel Jalocha's official mt63 code release

The MT63 modem is intended for amateur radio as a conversation (RTTY like) mode where one station transmits and one or more other stations can listen. In short, the modem transmits 64 tones in its baudrate specific bandwidth. The differential bipolar phase modulation is used to encode 10 bits of information per second on each tone. The user data in the form of 7-bit ASCII characters is encoded as a set of 64-point Walsh functions. The bits are interleaved over 32 symbols (3.2 seconds) to provide resistance against both pulse and frequency selective noise or fading. The character rate equals to the symbols rate thus the modem can transmit 10 7-bit characters per second.

This modem can as well run in two other modes obtained by simple time scaling, the possible modes are summarized here:

Bandwidth	Symbol Rate	Character Rate	Interleave / Char.
500 Hz	5 baud	5 char / sec	6.4 or 12.8 sec
1000 Hz	10 baud	10 char / sec	3.2 or 6.4 sec
2000 Hz	20 baud	20 char / sec	1.6 or 3.2 sec

For each mode the interleave factor can be doubled thus each character becomes spread over twice as long period of time.

The MT63 modem is made for single side band operation. The audio generated by the modem (sound card output) is applied to the SSB modulator. On the receiver side, the output of the SSB demodulator is put into the sound card input. The envelope of the MT63 signal is not constant as in other multi-tone systems - it is rather noise-like. One must be careful not to overdrive the transmitter.

The receiver of the MT63 is self-tuning and self-synchronizing thus the radio operator is only required to tune into the signal with +/- 100 Hz accuracy. The modem will tell the actual frequency offset after it is synchronized. The operator **should not** try to correct this offset unless he is able to tune the radio receiver very slowly, because MT63 as a low rate phase modulated system cannot tolerate sudden frequency changes.

The MT63 is a synchronous system and it relies on the sampling rate to be the same at the receiver and the transmitter. At least the sampling rates should not be different by more that 10^{-4} .

If you have calibrated your sound card to WWV, then you will meet this requirement.

Return to Top of Page

Return to Main Page

5.7 NAVTEX and SITOR-B

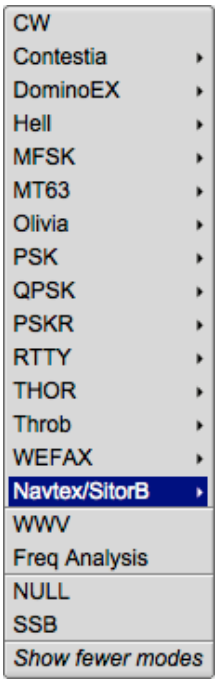


Figure 5.19: Op Mode Select

NAVTEX (Navigational Telex) is an international automated service for delivery of meteorological and marine safety information to ships. These broadcasts are sent with the SITOR collective B-mode (Also known as AMTOR-B or AMTOR-FEC), using the CCIR 476 character set. SITOR-B is also used in amateur radio, where it is known as AMTOR-B or AMTOR-FEC.

It transmits at 100 baud FSK modulation with a frequency shift of 170 Hz. The frequencies are:

- 518 kHz : International frequency, always in English.
- 490 kHz : Regional transmission, in local languages (Not used in the United States).
- 4209.5 kHz : Marine Safety Information.

The implementation reflects this structure: The Navtex modem is a specialization of the SITOR-B modem. Fldigi supports both modes. We will specify when their behaviors are different.

5.7.1 Transmitting a text

Transmitting a text is done with the usual GUI. The only difference between the SITOR-B and Navtex modes, is that data (Whether sent with the GUI or with XML/RPC) are untouched in SITOR-B. In Navtex, on the contrary:

- They are preceded by a phasing signal of several seconds.
- A "ZCZC B_1 B_2 B_3 B_4 " preamble is sent.
- Then the original message is transmitted.
- ... followed by the "NNNN" terminator,
- ... and another phasing signal.

5.7.2 Receiving a text

The only difference between the SitorB and Navtex modes, is that messages are processed (Parsed and optionally stored) only in Navtex mode. Here is a typical transmission showing:

- The end of a previous message : "NNNN"
- The preamble of a new message: "EA85": 85th *navigational warning* ('A') of the Niton station in England ('E').
- The message itself, without the terminator (Which should come soon).

The modem has some flexibility when interpreting messages, and is able to deal with missing or incomplete preamble and terminator.

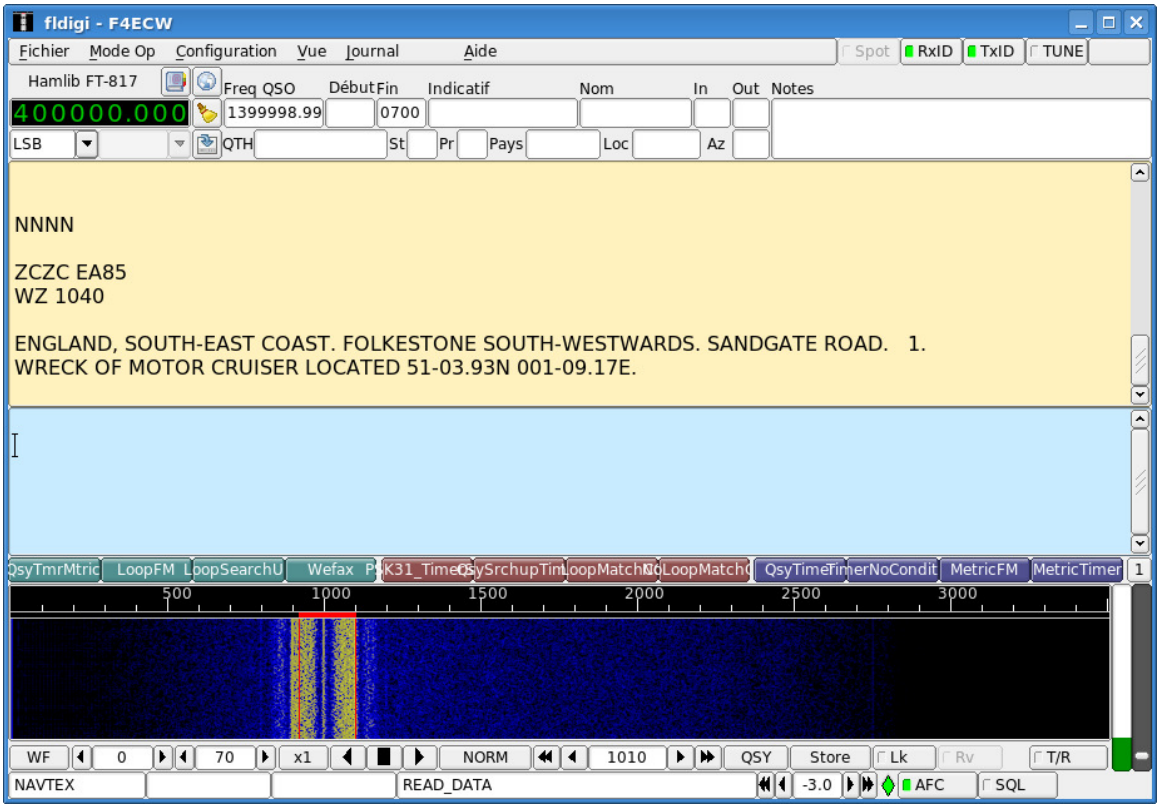


Figure 5.20: Navtex Reception

5.7.3 AFC: Automatic Frequency Control

As shown in the reception screen hard-copy, it is possible to tick "AFC". The consequence is that the frequency is continuously monitored.

5.7.4 Logging Navtex data to ADIF files

Navtex messages are delimited with the usual separators ZCZC and NNNN. Their format is:

ZCZC B_1 B_2 B_3 B_4

(message text ...)

NNNN

These four characters are:

- B_1 : The station origin, used for lookups in the Navtex stations file. The same character is associated to several stations. Therefore, we use other criteria such as the frequency and the distance to the receiver, to eliminate the ambiguity. It is therefore important to specify correctly your Maidenhead locator.

- B_2 : The subject indicator, used by the receiver to identify the message class, such as ice reports or meteorological messages.

- B_3 B_4 : A serial number between 00 and 99.

5.7.5 Configuration

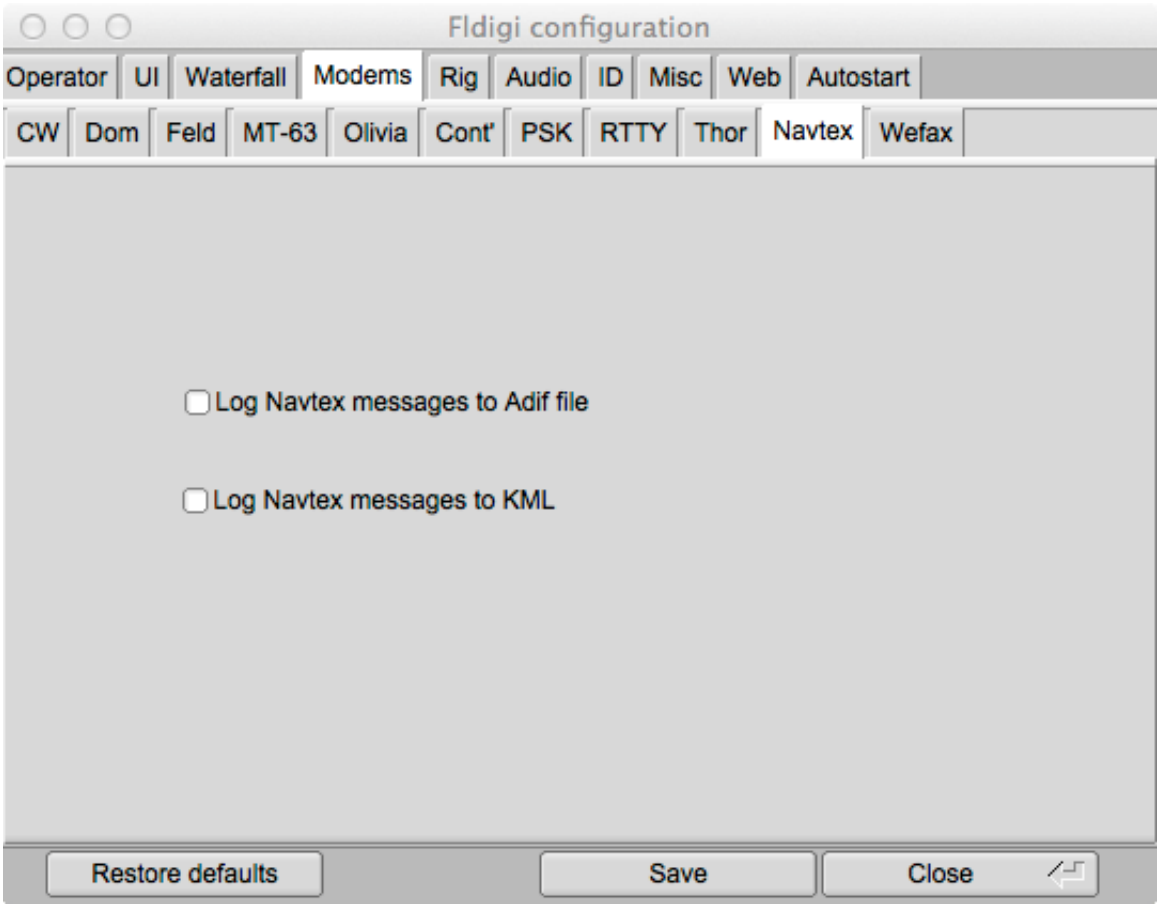


Figure 5.21: Navtex Configuration

As for any modem, there is a specific configuration tab for Navtex which allows:

- To specify whether messages should be stored in the current Adif log file.

- To choose the Text file containing stations definitions.

5.7.6 ADIF journal file

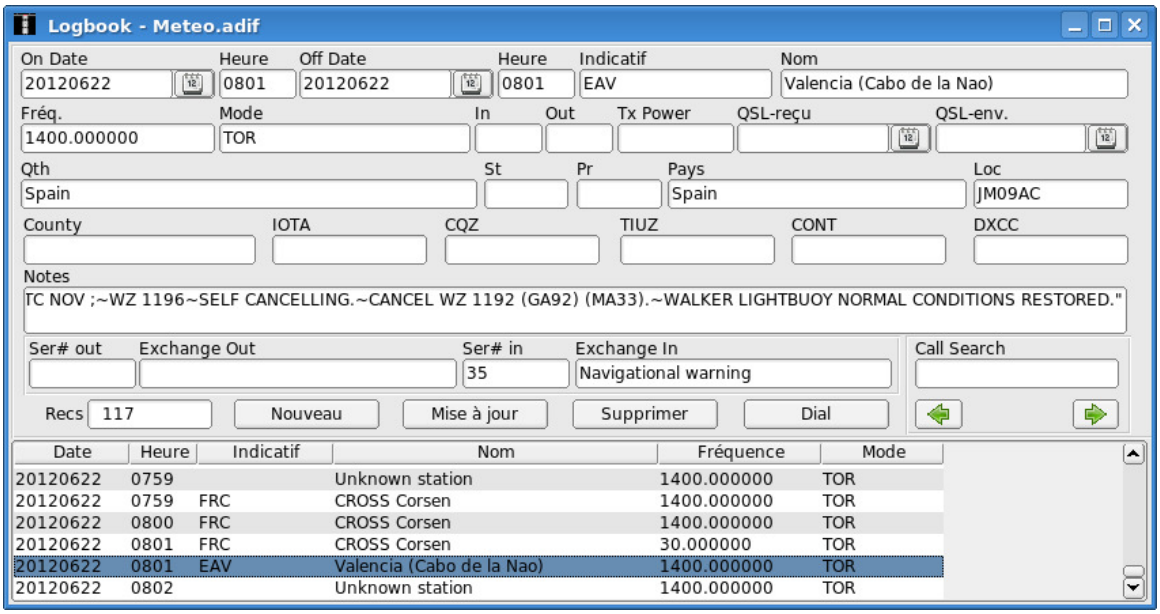


Figure 5.22: Logging Contacts

Navtex messages can be logged to the ADIF file. This is done with other extra data:

- Date and time of the contact.
- Frequency and mode.
- Station name, country, Maidenhead locator and callsign deduced from the message header and the Navtex stations file (See below). The locator is calculated using the station coordinates.
- The message content itself. Note that carriage-returns which cannot be displayed on a single line, are transformed into a tilda "∼".

5.7.7 Stations file

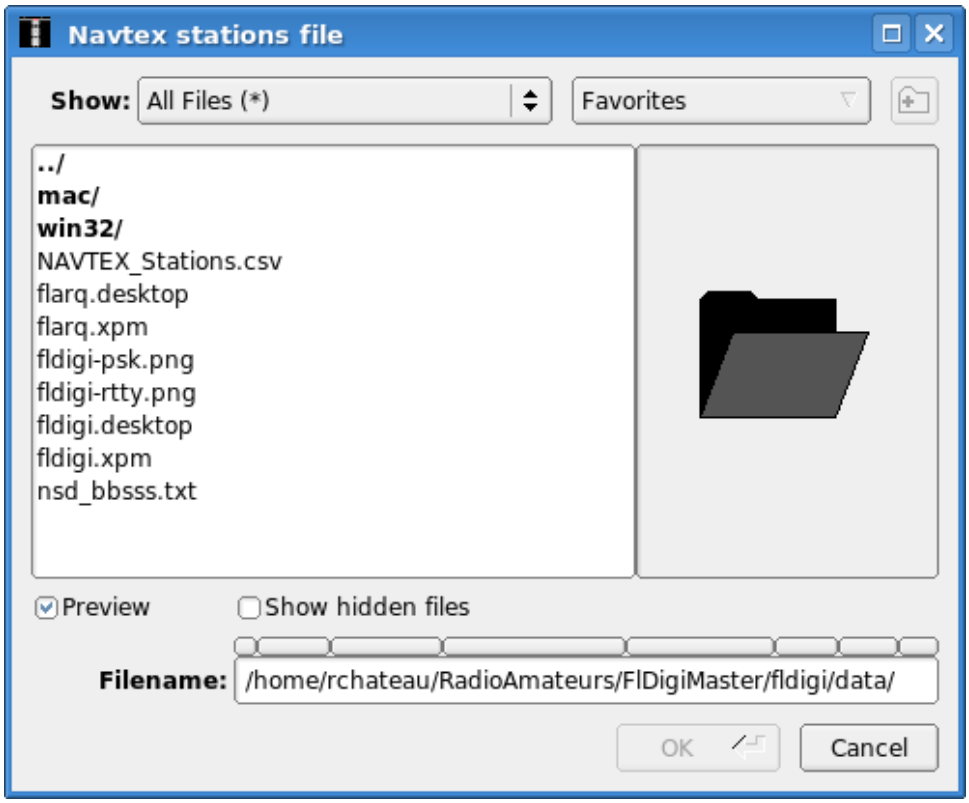


Figure 5.23: Choosing Stations File

The emitter station is extracted from the Navtex message, and used to extract latitude, longitude, station name and other characteristics from a file containing well-known station, data/NAVTEX_Stations.csv. It is possible to edit this text file, or to choose another one.

5.7.8 XML/RPC functions

Two XML/RPC functions are create:

navtex.get_message	Returns next Navtex/SitorB message with a max delay in seconds passed as an integer parameter. Empty string if timeout.
navtex.send_message	Send a Navtex/SitorB message, passed as a string. Returns an empty string if OK otherwise an error message

Return to Top of Page

Return to Main Page

5.8 Olivia

fldigi can operate on the following Olivia modes without special setup by the operator:

Mode	Symbol Rate	Typing Speed	Bandwidth
Olivia 8-250	31.25 baud	1.46 cps (14.6 wpm)	250 Hz
Olivia 8-500	62.5 baud	2.92 cps (29.2 wpm)	500 Hz
Olivia 16-500	31.25 baud	1.95 cps (19.5 wpm)	500 Hz
Olivia 32-1000	31.25 baud	2.44 cps (24.4 wpm)	1000 Hz

Unusual combinations of symbol rate and bandwidth can be selected using the Olivia configuration tab.

These are unconnected, simplex chat modes with full time Forward Error Correction. Olivia is a very robust mode with low error rates, but the penalty can be an annoyingly slow transfer of information. If you are a one finger typist then Olivia is your cup of tea. The tones are spaced the same as the baud rate, for example 31.25 Hz for the default baud rates. The default calling mode is 32-1000. It has the following appearance on fldigi's waterfall:

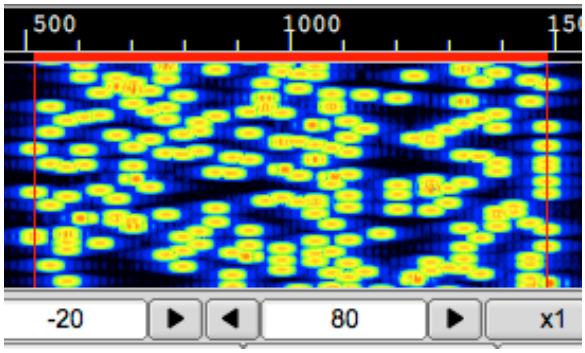


Figure 5.24: Olivia 32/1000

Excerpts from the web pages of `Gary, WB8ROL`

Oliva Mode is a little different than PSK, RTTY, and many other digital modes. Below are tips on how to maximize your use of this mode.

Disable your software squelch or turn it down as low as you can

Generally turn your squelch setting in your software off or set it as low as it will go. You will see some "garbage" letters get printed out if there is NO Olivia signal present but it doesn't harm anything. When an Olivia signal is there it will start decoding it and print out the text without garbage at that time. It doesn't do much good to use a digital mode like Olivia that can decode signals -14 db below the noise lever IF you squelch it AT the noise level! It would be like getting a pair of high power binoculars and using them only in a 10x10 room with no windows.

Be Patient!

When you call CQ on this mode be patient and wait at least 45-60 seconds before you put out another call. When the other person who hears your CQ clicks on the waterfall it may take 4-20 seconds or even longer before they might actually start decoding your signal. That varies a lot depending on the software they are using AND value they have their Sync Integration Period set to.

The Sync Integration Period setting determines how "deep" the Olivia decoding algorithm searches in the noise to get the signal. A higher settings takes longer BUT usually decodes with more accuracy - at least to a point. However, a higher setting (since it does more work and takes longer) will increase the delay factor. So, when you finish your CQ and your transmitter switches to receive - the station listening to you (depending on his Sync Integration Periods setting) MAY NOT finish decoding your CQ for another 4-20 seconds. The same applies during a QSO when you pass it back to the other guy for his turn – be patient if he doesn't come back right away because his software may still be decoding your signal long after you stopped transmitting.

It DOES NOT PAY to be impatient on this mode and send SHORT CQ's or NOT wait at least 45-60 seconds between CQ's. Generally a a 2x2 CQ sent at least 2 or 3 times is going to work much better for you than a short one. Below is the normal CQ I use though on real fast Olivia formats (like 500/4) I will do a 3x3 and send it 3 times.

```
CQ CQ de WB8ROL WB8ROL

CQ CQ de WB8ROL WB8ROL
```

```
CQ CQ de WB8ROL WB8ROL pse K
```

Don't set your Sync Integration Period setting TOO high

If you set your Sync Integration Period too high it MAY take minutes before your software will start decoding a signal AND there is no or little benefit to doing that past a certain point. I usually set mine so that the delay factor is abut 15-20 seconds. I can time this delay factor by sending a very short test and then when it is done and the software switches back to receive - time the number of seconds before you see random garbage start appearing on the screen (assuming you have your SQUELCH OFF). For the standard Olivia modes like 2000/64, 1000/32, 500/16, 250/8, and 125/4 that usually means my Sync Integration Period is set between 3-5 most of the time. If I use the faster formats I set it higher often between 6-10. As long as my delay factor is approx. 15-20 seconds. Any higher than that and I don't see any real improvement in the quality of the decoding. But play with your own settings and see what does best for you. If you leave it always on one setting, though, and use standard and non standard formats of Olivia you are short changing yourself.

Generally keep your Search (Tune Margin) setting to about 8

The setting of 8 is usually good for most situations and this setting is usually not all that critical. However, under a few band conditions it might (or might not) help to temporarily adjust this. If you find other Olivia signals very very close to you - almost adjacent or even overlapping it might help to reduce this setting to 4 or even 2. This setting determines how far, either side of your center frequency, Olivia will search for a signal to decode. If you reduce this when another Olivia signal is close or overlapping it may keep it from locking onto the other signal instead of yours. Also .... if you are trying to decode an extremely weak signal and can't even tell exactly WHERE to click on the waterfall because the trace is too faint or non existent then it might help to increase this setting to 16 or 32 temporarily. Then it would perhaps decode the signal even if you were OFF his center frequency by a large margin.

If the slow speed of Olivia bothers you some ...

If you find yourself wanting things to go a little faster then start using more (ham) common abbreviations like "hw" for how and "ur" for your. Don't waste time sending words like "the" and "and" all the time. An example : The weather here is nice and sunny today and the high will get to 85 degrees — instead send : Wx nice + sunny - high 85 deg – No need to spell out everything and use superfluous words like the, and, many others. And why use words like HERE and TODAY in the above context when the other station already knows you are telling the weather for YOUR QTH for TODAY. You aren't writing a novel, an article, or in a spelling bee. Also after you establish the QSO don't send BOTH calls all the time at the beginning and end of every transmission. After the QSO is in progress come back to the station like this : .. de WB8ROL – instead of : W9ZZZ de WB8ROL – and when you sent it back to the other guy send : BTU - de WB8ROL KN – That will help speed things up too. You don't need to send the other stations call sign continually to fulfill your legal obligation to identify your own station.

Don't be afraid to switch to a NON standard Olivia format if conditions warrant it.

If signals are real strong and you prefer to be sending and receiving at a faster speed - don't be afraid to ask the other station if they would like to speed things up and switch to another Olivia format - even a non-standard one. If you, for instance, were talking to me on 500/16 Olivia format and we both had very strong signals and not much QR-M, QRN, etc. then ask me if I would like to go to 500/8 format or even 500/4 format. 500/16 format is approximately 20wpm while 500/8 is close to 30wpm and 500/4 close to 40wpm. If you do end up switching to the faster modes you may also want to increase your Sync Integration Period setting substantially too - to maintain the best quality decoding. If not, you might get more errors in the decoded text. And if the band conditions become worse - go back to the original format AND remember to reset your Sync Integration Period setting or the delay in decoding will be way too long! Also, if the band starts getting real crowded and say, for example, you were on 500/16 mode - you might suggest to the other station to switch to 250/4 mode (increase Sync Integration Period setting too) to save space and be a "good neighbor" to all the other operators nearby. 250/4 is the SAME speed as 500/16 and nearly as sensitive with the correct settings.

Return to Top of Page

Return to Main Page

5.9 PSK - BPSK, BPSKR, QPSK, and Multi-Channel Modems

5.9.1 BPSK and QPSK modems

PSK are narrow band low symbol rate modes using either single or multiple carrier differential Binary Phase Shift Kying, BPSK, or Quadrature Phase Shift Keying, QPSK.

PSK63FEC and the PSKxxxR modes are forward error correcting modes. PSK63FEC is compatible with the Multi-Psk mode of the same name. The PSKxxxR, or robust, modes use both forward error correction and interleaving to achieve about 4 dB s/n improvement over standard PSK. These modes are use primarily by the PskMail user community. They are the invention of John Douyere, VK2ETA, a member of the fldigi development team.

In addition to the binary phase shift keying the signal is 100% raised-cosine amplitude modulated at the symbol rate. This reduces the power to zero at the phase change. Because of this amplitude modulation, the signal bandwidth is relatively narrow. Synchronization at the receiver is straight forward because it can be recovered from the amplitude information. Differential PSK is used to provide continuous phase changes when idle (to maintain sync), and by allowing the receiver to measure phase difference from symbol to symbol, to reduce the effects of ionospheric Doppler phase changes which modulate the signal. The slower modes are more affected by Doppler, and the QPSK modes are particularly affected.

With no interleaver and limited coding length, the QPSK mode Forward Error Correction coding gain is limited, and under burst noise conditions on HF the performance is usually worse than the BPSK option at the same baud rate. In general the narrow-band BPSK modes work well on a quiet single-hop path, but give poor performance in most other conditions.

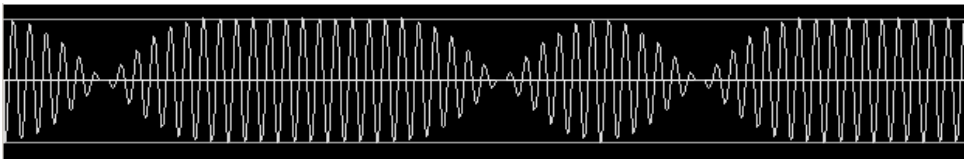


Figure 5.25: PSK63 signal transmitting text data - oscilloscope view

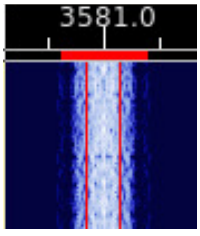


Figure 5.26: PSK63 signal transmitting text data - waterfall view

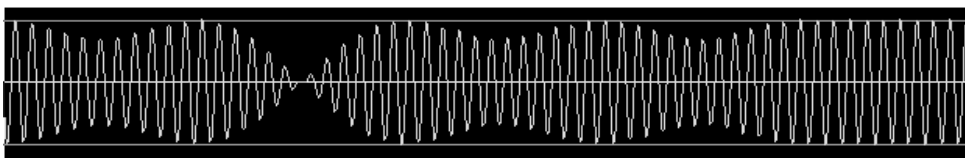


Figure 5.27: QPSK63 signal transmitting text data - oscilloscope - waterfall view

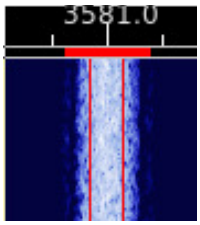


Figure 5.28: QPSK63 signal transmitting text data - waterfall view

The two oscilloscope views above clearly show the combined phase and amplitude modulation of these modes.

With these modes, a very linear transmitter is required. Over-driven operation results in excessive bandwidth, poorer reception and difficult tuning. Overdrive usually occurs by having the audio signal much too large. These are very sensitive modes and usually very little power is required. QRP operation of 80, 40, 30 and 20 meters can provide nearly 100% copy over multi-hop paths. In many instances PSK can provide better decoding than CW.

Setting up for a good clean on air signal that will receive the accolades of your QSO partners is easy. Follow the instructions on using the Tune button and you will have a clean on signal.

Good reception of PSK signals requires that the demodulator be phase locked to the incoming signal. Fldigi has both a fast acquire / slow tracking AFC system. Place the red bandwidth bar (see above) so that it overlies the desired signal and then press the left mouse button. The signal should quickly lock on a decoding should commence immediately. It is almost impossible to visually tell whether a BPSK or QPSK signal is being received. Under very high s/n you might be able to hear the difference, but that is even difficult for most operators. If you are not able to decode a signal that looks like a BPSK and the bandwidth of the signal matches the baud rate then it might be a QPSK signal. Just change mode a try reacquiring the signal.

5.9.2 Multi-Channel PSK modems

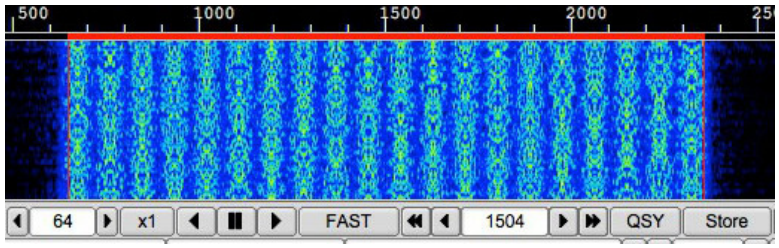


Figure 5.29: PSK63R20C signal transmitting text data - waterfall view

Return to Top of Page

Return to Main Page

5.10 RTTY

The RTTY modulator and demodulator have been extensively changed with version 3.21.67. The new design was a cooperative effort of Stefan, DO2SMF, and Dave, W1HKJ with extensive testing performed by Ed, W3NR, and Dick, AA5VU. Chen, W7AY, was a silent contributor to the design by virtue of his excellent technical papers on RTTY modulation and demodulation, which he so generously placed in the public domain.

fldigi can operate on a wide range of RTTY symbol rates and bandwidths. The selection of symbol rate and bandwidth is made on the RTTY configuration tab. The three most common in amateur radio use can be selected from the mode menu. These are

Mode	Symbol Rate	Typing Speed	Bandwidth
RTTY 45	45.45 baud	6.0 cps (60 wpm)	270 Hz
RTTY 50	50.0 baud	6.6 cps (66 wpm)	280 Hz
RTTY 75	75.0 baud	10.0 cps (100 wpm)	370 Hz

These modes were a result of mechanical and electrical designs of the early TTY machines. The 45.45 baud and 75 baud machines were for the US / Canadian market and used 60 Hz synchronous motors. The 50 baud machines were for the European market and used 50 Hz synchronous motors.

fldigi can encode and decode many other symbol rates and bandwidths. "Custom" combinations are set up on the RTTY configuration tab. You probably will never have to do that unless you like experimenting with unusual RTTY modes.

5.10.1 RTTY modulator

All of the modem signals that fldigi produces are audio signals. That includes the RTTY signal. fldigi can encode and decode an RTTY signal that is anywhere within the passband of the sideband transceiver. It is not limited to the traditional tone pairs around 2100 Hz. Each of the generated Mark / Space signals are on-off-keyed (OOK), bandwidth limited signals. The resultant waveform is not an FM type signal of constant amplitude. Therefore the **transmit audio and RF amplifiers must be linear**, just like the requirement for PSK signals. There are performance gains using this approach. The principal being a reduction in inter symbol interference which gives much improved performance by the receiver. The waterfall, time domain, and spectrum signatures of the transmitted signal look like this:

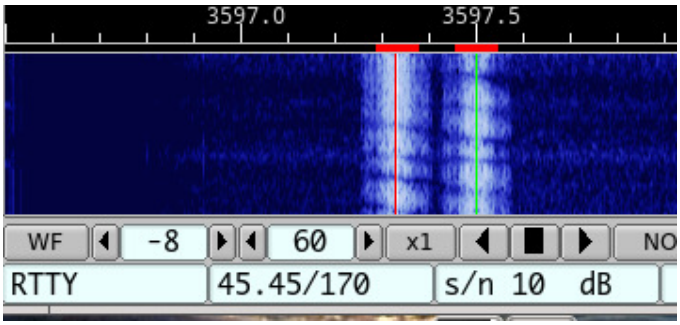


Figure 5.30: W1AW on air signal

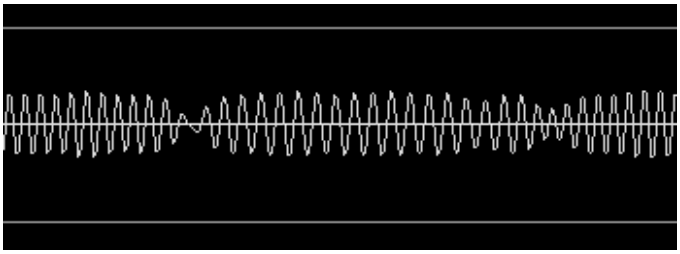


Figure 5.31: AFSK signal

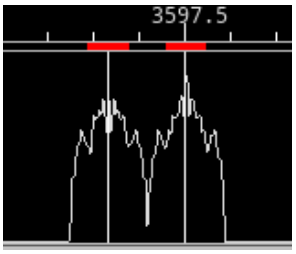


Figure 5.32: Spectrum

You must operate your transceiver in the USB mode for the fldigi RTTY signal to be the correct polarity. If your transceiver is set to LSB then use the fldigi "Rev" button to reverse the sense of the mark and space signals.

You must maintain transmitter LINEARITY in the AUDIO AMPLIFIERs. Do not think that you can improve performance by over driving the audio input. A good operating procedure for most transceivers is the set the audio level to the level for which there is just barely a hint of ALC. Then reduce the input to 80% of that power level. Over driving an AFSK signal is as disastrous as over driving a PSK signal. This is an actual on air signal that was being over driven (but not on purpose):

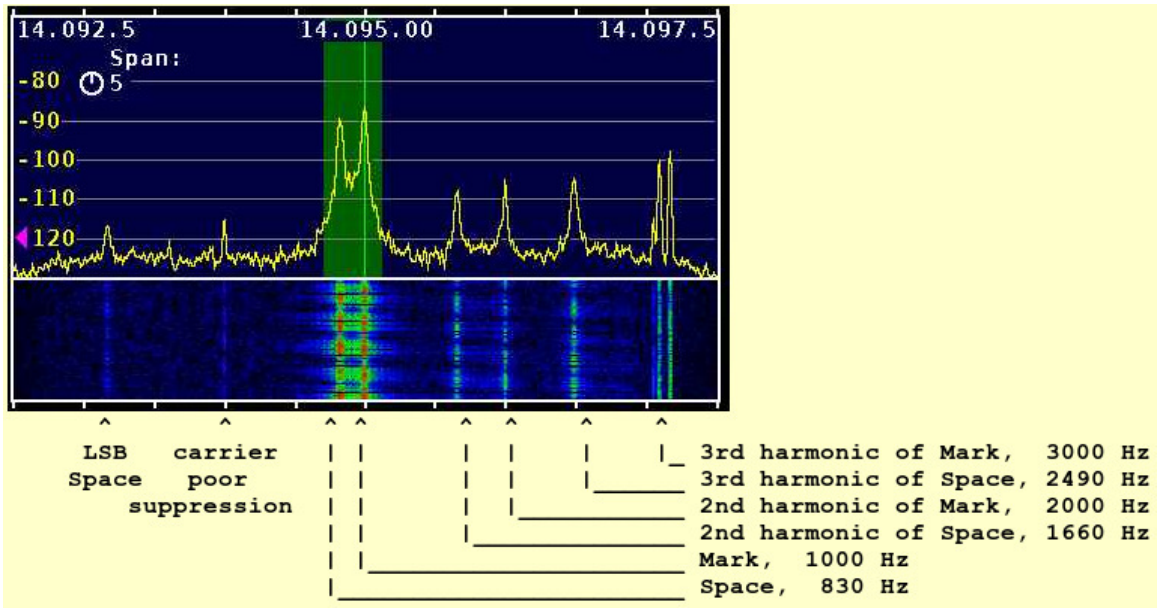


Figure 5.33: Overdriven RTTY Signal

Joe also performed a series of tests on an Icom 706 mkIIg transceiver. The results of those tests are very enlightening.

"Two views - the 2 KHz span and a 10 KHz span. The 10 KHz span shows the one failing of the IC-706mkIIg and other rigs with analog modulation - opposite sideband and carrier leakage. This one isn't too bad but one can see carrier at -50 dBc and opposite sideband at -55 dBc +/-. I do use a high audio frequency to minimize harmonic issues.

For fun I've attached versions at 70 W in 10K, 5K, and 2K spans. The narrow spans clearly show the benefits of reducing the audio until output power drops 1.5 dB.

Audio was connected to the IC-706mkIIg via the "DATA" jack rather than the mic connector or "Mod In" pin of the ACC jack. Using this input avoids several potential problems - including the constant swapping between mic and digital connections and remembering to turn off the compressor when switching to digital operation. The "Data" input is also 6 dB less sensitive than "Mod in" making it that much less likely that one will significantly over drive the the transceiver and create distortion in the audio stages ahead of the modulator".

The green area is 400 Hz wide.

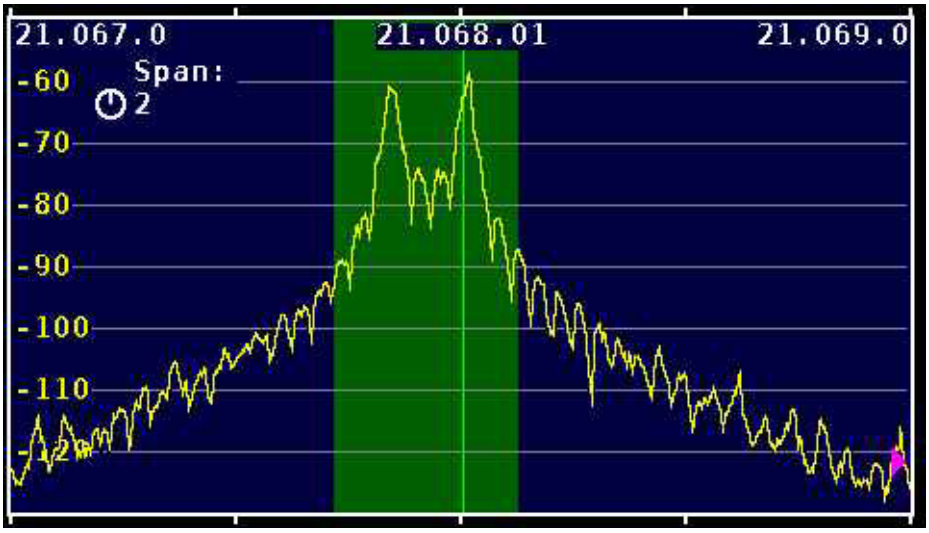


Figure 5.34: Image A

Transceiver operated in FSK mode

Power: 100 W

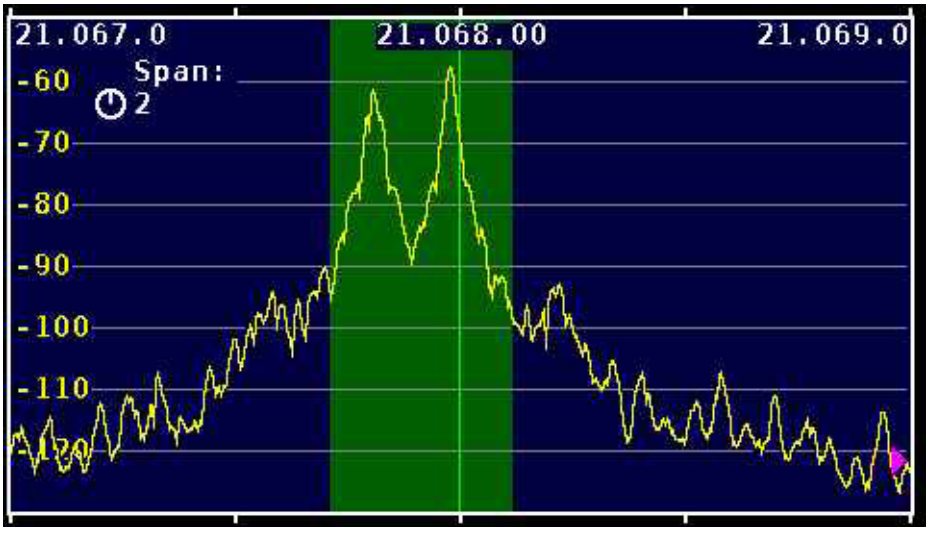


Figure 5.35: Image B

Transceiver in USB, fldigi AFSK audio drive

Space at 2125, Mark at 2295 Hz

Power: 100 Watts, ALC just extinguished

Figure 5.36: Image C

Transceiver in USB, fldigi AFSK audio drive

Space at 2125, Mark at 2295 Hz

Power: reduced to 80 Watts (-1 dB)

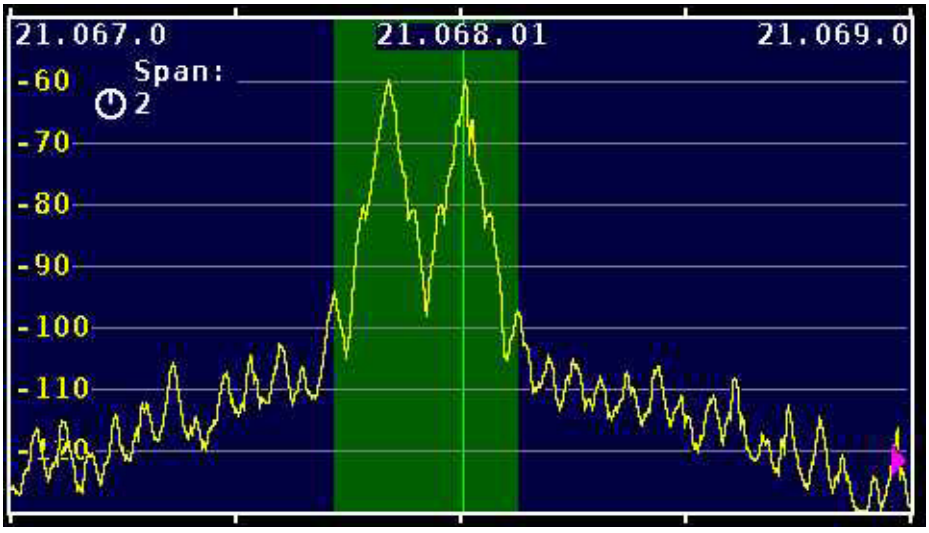


Figure 5.37: Image D - 2 K span

Transceiver in USB, fldigi AFSK audio drive

Space at 2125, Mark at 2295 Hz

Power: reduced to 70 Watts (-1.5 dB)

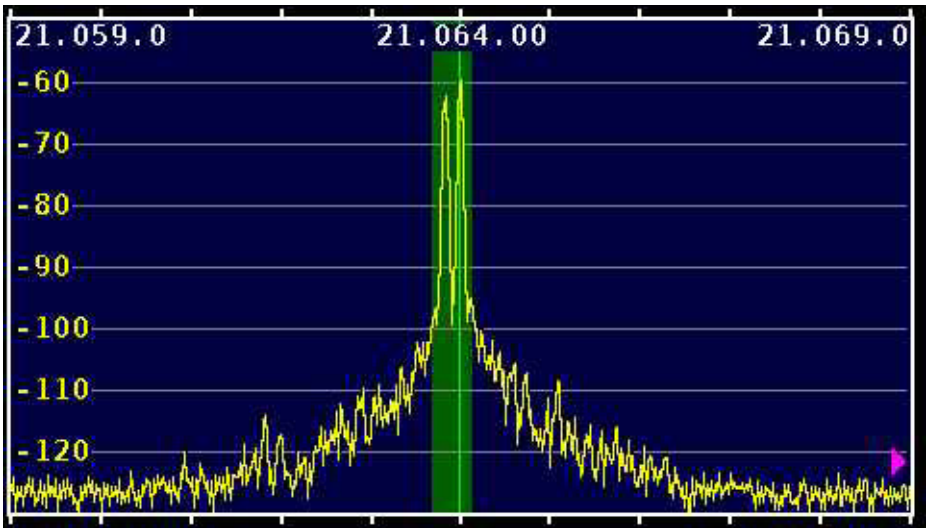


Figure 5.38: Image F - 10 K span

Transceiver in USB, fldigi AFSK audio drive

Space at 830 Hz, Mark at 1000 Hz

Power: 70 Watts. The LSB leakage is clearly

seen at approximately -55 dB

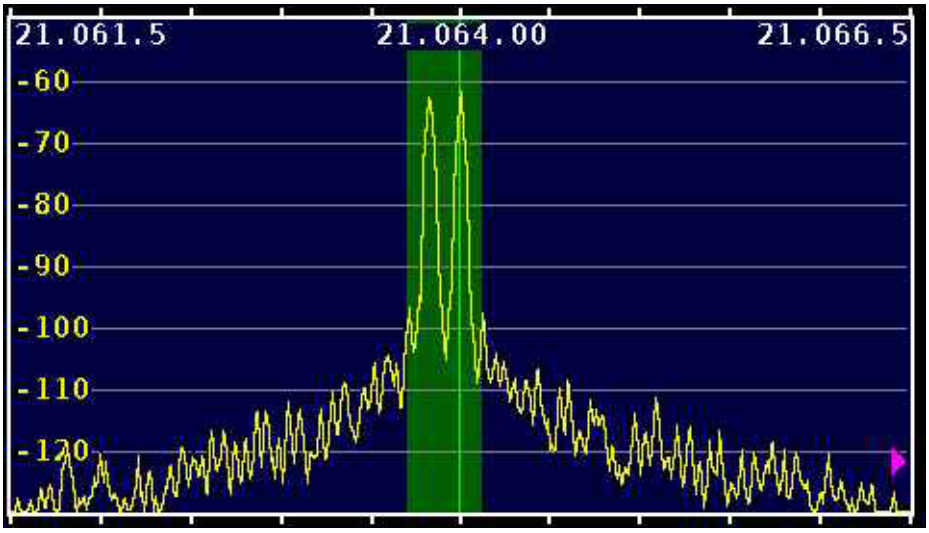


Figure 5.39: Image G - 5 K span

Transceiver in USB, fldigi AFSK audio drive

Space at 830 Hz, Mark at 1000 Hz

Power: 70 Watts. The LSB leakage is clearly

seen at approximately -55 dB.

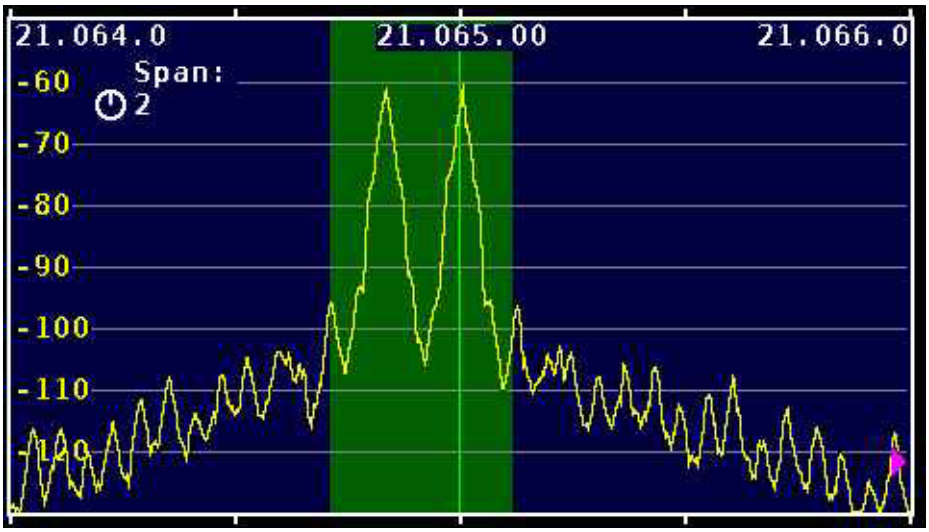


Figure 5.40: Image H - 2 K span

Transceiver in USB, fldigi AFSK audio drive

Space at 830 Hz, Mark at 1000 Hz

Stop Bits set to 2 vice 1.5

Power: 70 Watts. Compare this to image D

5.10.2 RTTY demodulator

Fldigi's demodulator uses a design based on theoretical work published by Kok Chen, W7AY, http://www.-w7ay.net/site/Technical/ATC/.

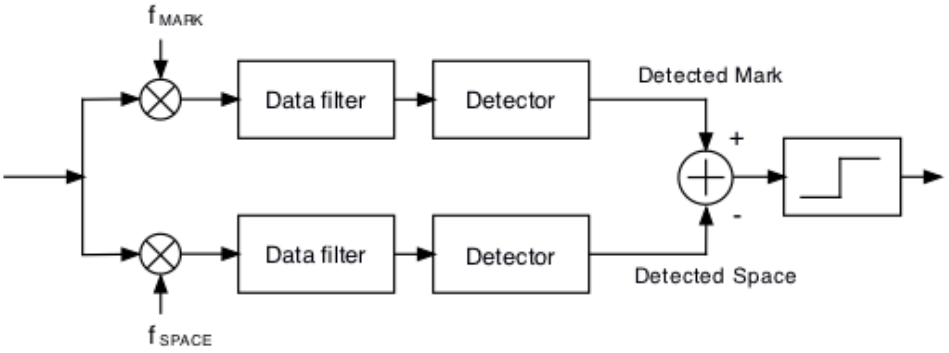


Figure 5.41: Demodulator

The mark and space signals are converted to base band and then filtered in a low pass filter. Each filter is a variant of Chen's enhanced Nyquist filter. It is implemented using a Fast Overlap-and-Add Fourier Transform.

Each time the baud rate is selected the program must "rebuild" the digital RTTY filter. The filter parameters are optimized for the baud rate.

The detector is an optimized Automatic Threshold Correcting (ATC) type described in Chen's paper.

To start decoding a signal simply left click on the signal and the AFC should lock on to the signal.

The digiscope display will extinguish when the Rx signal level falls below the squelch setting.

It is possible to use fldigi to generate the keying waveform for use with an FSK type of transmitter. See Pseudo FSK for adescription of how this can be accomplished.

See RTTY Configuration Page

Return to Top of Page

Return to Main Page

5.11 Thor

Thor is a new forward error correcting incremental frequency shift keyed communications mode. It was developed specifically to meet the needs of ARQ transfers in the HF spectrum. It is particularly well suited under conditions of atmospheric static noise. Thor borrows from two current modem technologies, MFSK and DominoEX.

Thor emits a distinctive double rising tone sequence at the beginning of each transmission. It is used to flush the receive decoder and also provides a visual and audible clue to its being used.

The modem code for Thor uses a wide band multiple frequency detector that can lock on and detect the incoming signal even when badly mistuned. Frequency domain oversampling is used to allow proper tone detection without the need for AFC. The AFC control does not alter the decoder in any way.

The waterfall and digiscope will appear as:

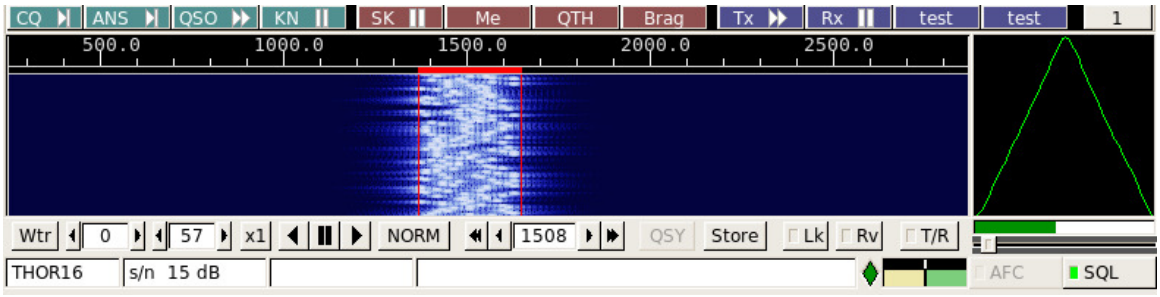


Figure 5.42: Thor

The text displayed in the status area is the secondary text being sent by the transmitting station. When the keyboard buffer is empty the Thor modem transmits text from the secondary text buffer. Your secondary text buffer can be edited on the Thor configuration tab.

The digiscope display is similar to the DominoEX display and represents the tone pairs moving through the tone filters. You can also use an alternate digiscope display (left click on the digiscope display area).

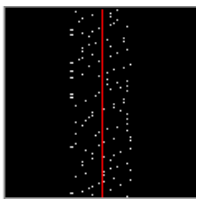


Figure 5.43: DominoEx

In this display mode the red line represents the center of the multiple tone bins that are in the detector. The dots will be blurry if the AFC is not locked on and become very distinct when AFC lock has been achieved. The tone dots will move from bottom to top (opposite the direction of the waterfall).

This is the same signal mistuned:

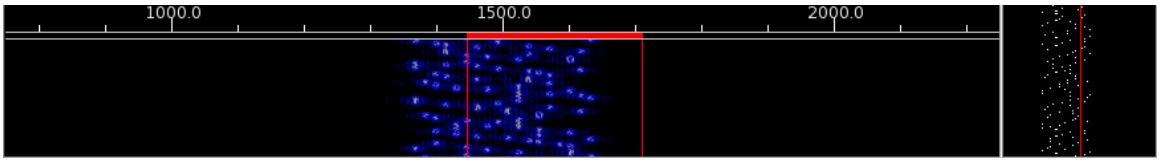


Figure 5.44: DominoEx Mistuned

and with the signal badly mistuned:

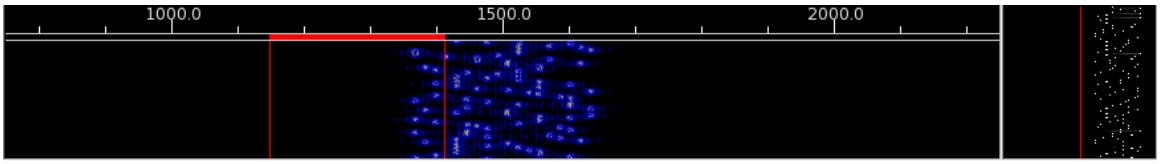


Figure 5.45: DominoEx Mistuned 2

Return to Top of Page

Return to Main Page

5.12 Throb

The THROB family of modes use two tones at a time. These tones are also amplitude modulated and can be a single tone for some symbols.

The mode has no Forward Error Correction, and is difficult to tune. It is fairly sensitive and moderately robust. Keyboard-to-keyboard QSOs are reasonably fast.

Tuning must be very accurate, and the software will not tolerate differences between transmit and receive frequency.

The amplitude modulation component of THROB is a raised cosine AM modulation of each symbol. This combined with two tones transmitted at the same time, means that a very linear transmitter is required. It also gives the mode its very unique sound. You will never mistake Throb for any other mode.

For THROB, nine tones are used, spaced 8 or 16 Hz. For THROBX, 11 tones are used, spaced 7.8125 or 15.625 Hz.

Fldigi supports the following:

5.12.1 Throb baud rates and tone spacings

Mode	Symbol Rate	Typing Speed	Bandwidth
THROB1	1.0 baud	1.0 cps (10 wpm)	72 Hz
THROB2	2.0 baud	2.0 cps (20 wpm)	72 Hz
THROB4	4.0 baud	4.0 cps (40 wpm)	144 Hz
THROBX1	1.0 baud	1.0 cps (10 wpm)	94 Hz
THROBX2	2.0 baud	2.0 cps (20 wpm)	94 Hz
THROBX4	4.0 baud	4.0 cps (40 wpm)	188 Hz

Return to Top of Page

Return to Main Page

5.13 TUNE - Transmit a single tone carrier.

5.13.1 TUNE Mode

Too often you see an overdriven signals on the digital sub-bands; multiple audio sidebands on PSK, splatter from overdriven MFSK and RTTY. There is absolutely no reason for a transceiver driven by fldigi to exhibit this type of performance. You can set up your computer / transceiver for good solid performance without excessive drive.

The "TUNE" button generates a continuous single frequency audio signal at the exact frequency to which the waterfall cursor has been set. The peak amplitude of this signal is the peak amplitude of every modem signal generated by fldigi. None will exceed this value, even the simultaneous multi-tone modes like Throb. Every modern SSB transmitter uses some automatic level control ALC for preventing overdrive for SSB voice. A little overdrive on a voice channel can be tolerated to a degree. In fact, that is what an analog RF compressor does, overdrive and then subsequent filtering. But you absolutely cannot tolerate that with the digital modes. Here is the way to set up your transceiver for a clean signal. I recommend starting out with a dummy load, but an "off hour" for a band might work just as well if you do not have a dummy load.

- For Windows users
  - Set your sound card output level to the minimum on the Windows mixer
- For Linux users
  - Set your PCM level to about 80%
  - Set your Transmit Level control for minimum output level.
- Enable the "Tune" mode in fldigi ... you do have CAT or PTT set up ...right?
- Make sure your transceiver's speech compression control is OFF
- Slowly bring up the Mixer audio out until your rig's ALC just starts to function (a light blinking or a meter showing this condition).
- Reduce the Mixer audio output until the ALC is disabled.
- You are now transmitting at maximum output power without distortion.

You can use any level below this and be assured that your output signal will be clean. All digital signals that fldigi generates will be limited to this peak-to-peak voltage. You should always use the minimum power necessary to maintain good comms, remember that even if you are clean at 100 W you signal will be so strong among the QRP signals that it will overpower the AGC on many receivers that are working another digital station within the same SSB bandwidth that you are on. You will appreciate this the first time that you are working a weak PSK DX station and someone blasts through and captures your AGC.

You should try the the above adjustments at different audio frequencies. Transceivers that achieve the SSB filtering with crystal or mechanical filters will have a considerable amount of variation across the passband of the filter. This will show up as a varying amount of ALC that is dependent on the audio frequency. Once you are comfortable with the process you can very quickly repeat the "Tune" and set the power for the frequency to which the waterfall is set.

Return to Top of Page

Return to Main Page

5.14 WEFAX

This modem is able to receive and transmit HF-Fax images, traditionally used for weather reports.

More technical information is available on the wikipedia article Radiofax.

Two modes are implemented IOC=576 or 288. The focus is made on black-and-white images, color mode is still experimental.

Many frequencies are available at http://www.hffax.com/ for example.

When entering any Wefax mode, the reception window opens, and optionally the transmit window. it is always possible to bypass this with the menu bar:

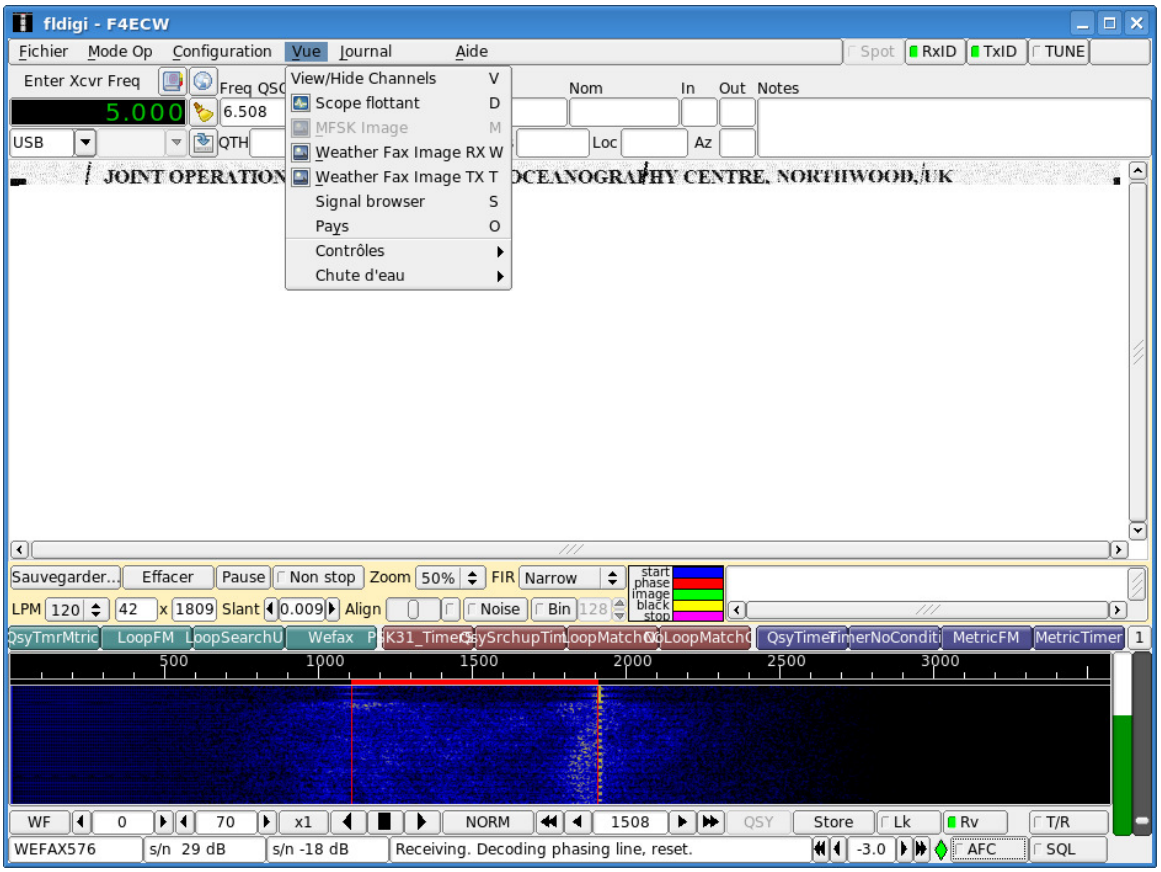


Figure 5.46: WEFAX

5.14.1 Configuration.

As for any modem, weather fax has its own tab in the configuration window.

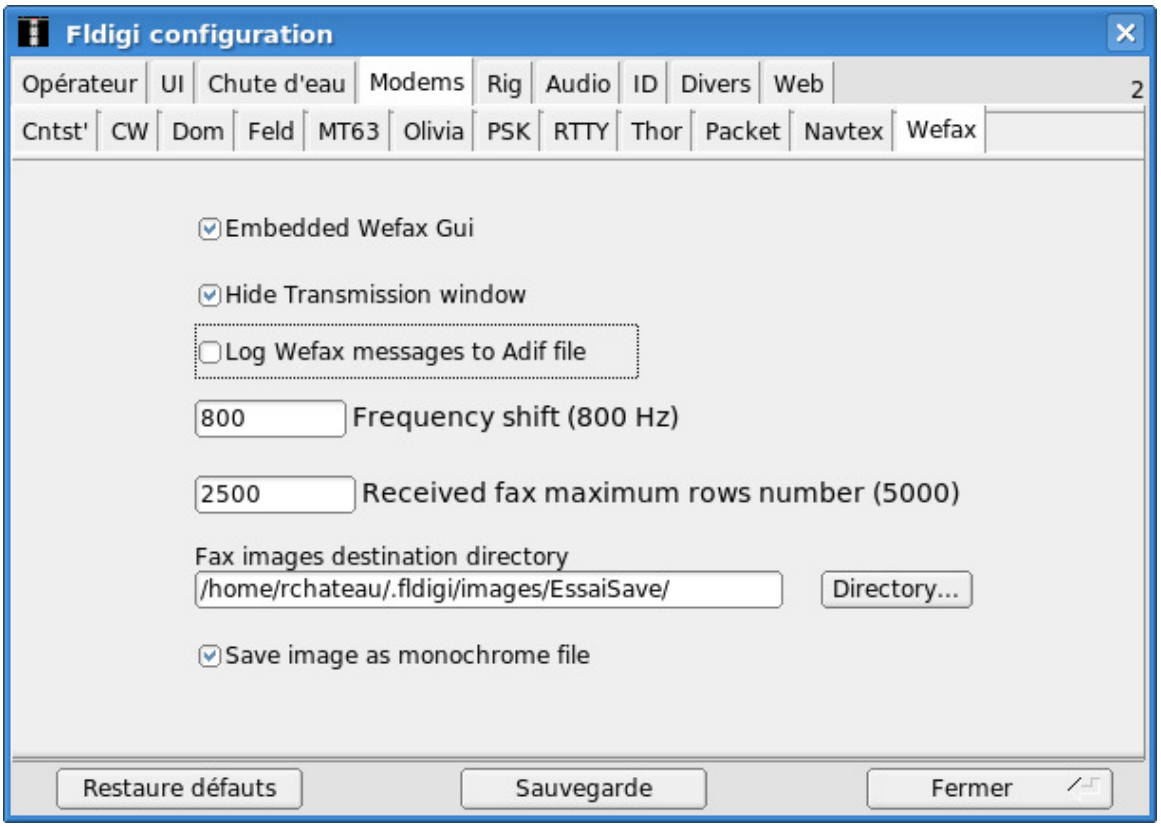


Figure 5.47: WEFAX Configuration

5.14.1.1 Embedded vs floating display mode

There are two display modes for this modem:

- Embedded mode: This is the default mode, the normal receipt and transmit windows gets graphic (Like Hellschreiber mode).

- Floating mode: There are two separate windows for transmission and reception. This was the only available mode until fldigi 3.21.49.

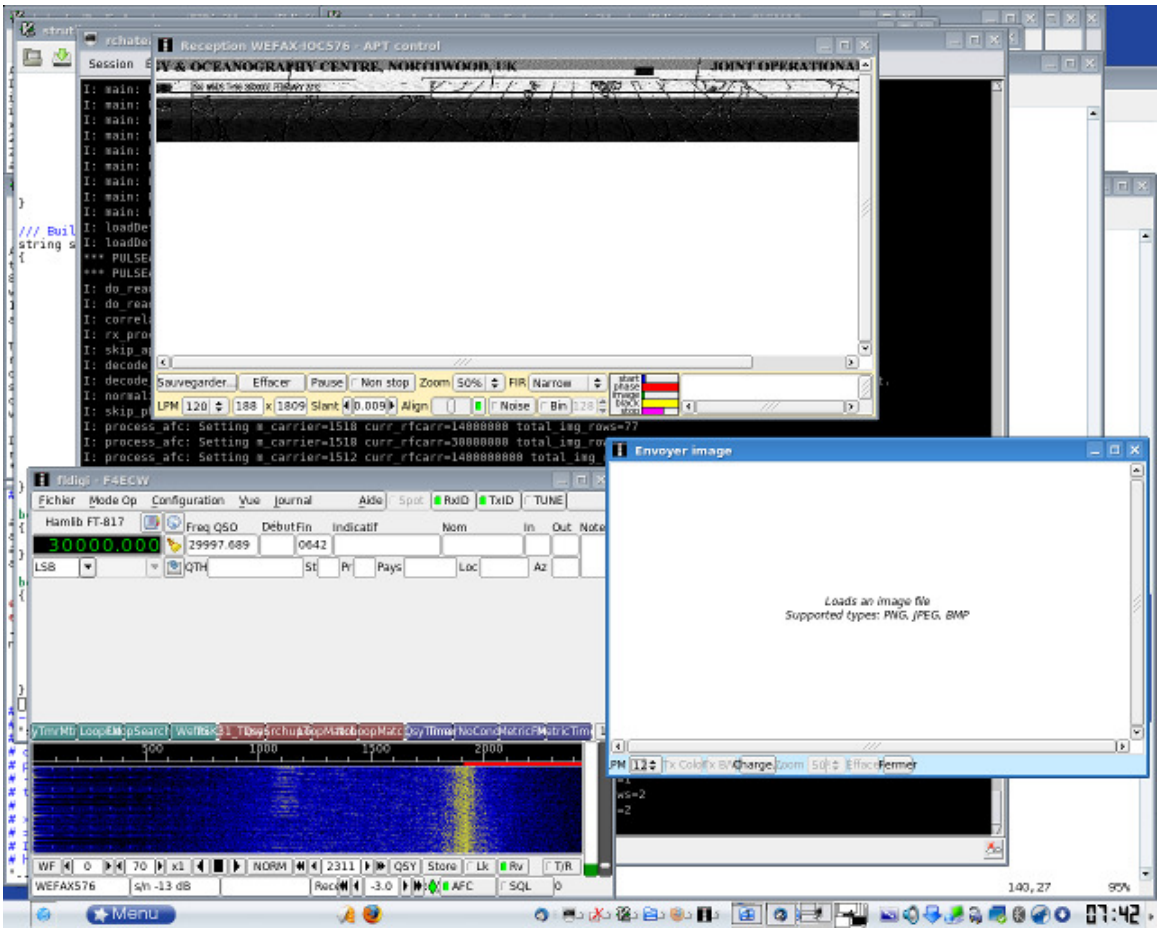


Figure 5.48: Detached WEFAX Window

5.14.1.2 Hide transmission window.

This option, allows to closed by default, the transmission window, when entering Wefax mode. In embedded mode, this means that the entire window is used for reception (Most common mode). In floating mode, this implies that the transmission window is not opened by default.However, it is always possible to manually open or close the transmission window at any time.

5.14.1.3 Logging messages to ADIF file

Each time an image is saved, it is possible to log this event, with the frequency and reception time, to the Adif file. This option is disabled by default.

5.14.1.4 Frequency shift adjustment

The default frequency is 800 Hz. However, it is possible to adjust for example to 850 Hz for Deutsche Wetter Dienst.

5.14.1.5 Maximum rows number

It non-continuous (Non non-stop) reception mode, an image is automatically saved when it has more than this number of lines (Default 2500 lines). Once this number of rows is reached, the image is saved and a new image is read with the same parameters. This feature has two applications:

- In automatic mode (APT control), if an image end is not detected, we can guarantee that the result will take no more than, for example, the size of two faxes. Typical faxes have about 1300 lines, so the max lines parameters can be tuned to, for example, 200 lines.

- In manual mode, where images are read continuously, this cuts the received images into chunks of equal size.

5.14.1.6 Destination directory for saved images

Received images are saved in the default folder

`$HOME/.fldigi/images (Linux) or <defaultpath>/fldigi.files/images`

(Windows).

Additionally, they can be saved manually, at any time, using the button 'Save'. The PNG images received some extra text comments which can be displayed, for example, with GIMP.

5.14.1.7 Monochrome images

Fax images are monochromes and are saved as such by default. However, it is possible to bypass this parameter and save them as color RGB images.

5.14.2 Transmitting an Image

To open the transmit window, you must of course select one of the two Wefax modems, and then right-click on the transmit (blue) window:

Then, the transmit window just opens. This is the same logic as sending MFSK images.

Then, you must open an image file using the button "Load". The image is then displayed, for example like that:

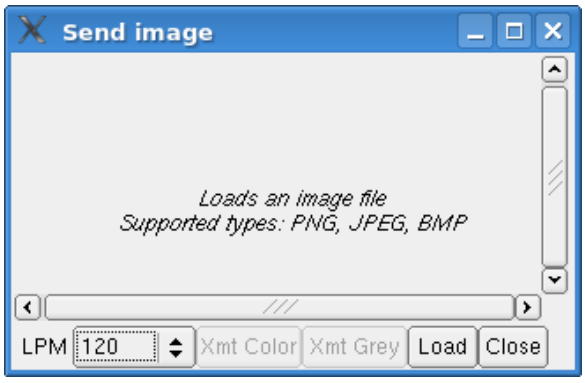


Figure 5.49: Transmitting

Now, to start the transmission, you just need to click "Tx B/W" for black-and-white images, etc... During transmission, image reception is paused. The window will display each image line as it is sent. Please be patient, this may take a while. You might note that FlDigi status line displays the estimation transmit time, and the current stage (Start, phasing etc...); Color transmission ('*Tx Color*') is intentionally disabled at the moment.

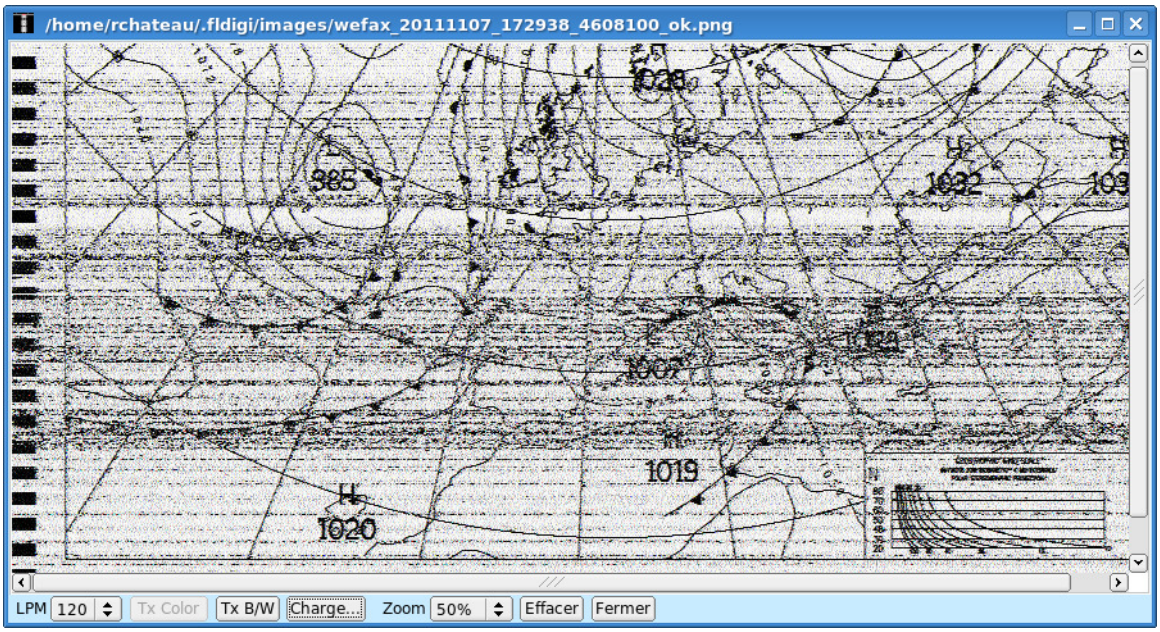


Figure 5.50: Sending

5.14.3 Receiving an image

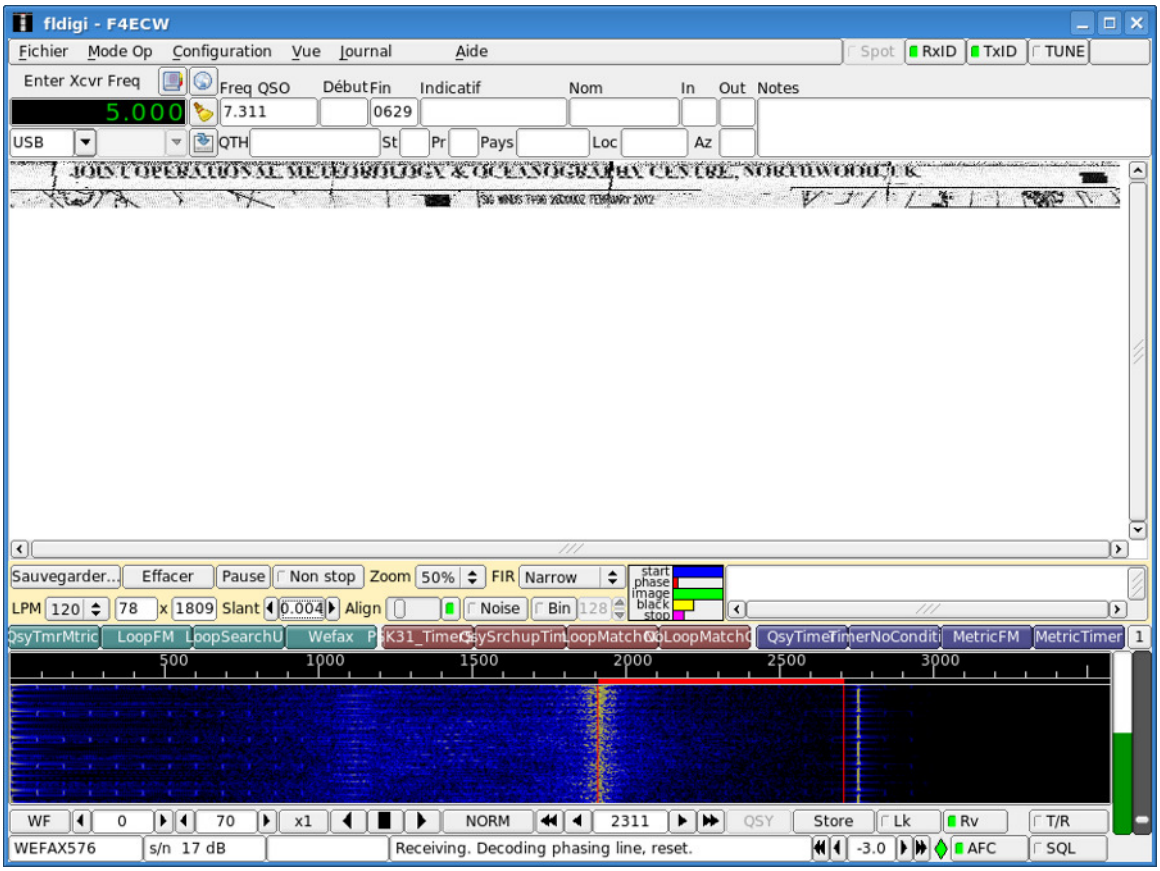


Figure 5.51: Reception

To enter reception mode, one can click the "View" menu tab, and select "Weather Fax Image".

At this time, the reception window opens. A big blank picture is visible, this is where the received image will be displayed. Several controls are available:

- Save: This allows to save the current image as a PNG file at any moment.
- Non-stop: At startup, the modem goes into automatic mode, and the text "Abort" is displayed. When clicking "Abort", this blanks the image and resets the APT detection. When "Manual mode'" is clicked, no APT detection is done. The Automatic/Manual mode is displayed in the reception window label.
- Pause/Resume: At any moment, the image reception can be paused and resumed (State is displayed in the reception window label).
- Zoom: This allows to zoom in/out the image.
- FIR: This allows to select an input FIR (Finite Impulse Response) filter. Practically, the narrow filter (Default value) gives the best results. The selected value is saved in the configuration file.
- Skip APT/Skip phasing: When in automatic mode, this allows to skip detection steps. This is often necessary when the signal is not very good.

When receiving an image, either in manual or automatic mode (APT control), other controls are displayed:

- Line: The number of the line currently received.
- Width: The image width in pixels. This is usually 1809, if LPM is 120.
- LPM: Lines per minute: Typically 120, can be 60, 90, 120 or 240. This is detected in automatic mode, but can be manually adjusted.
- Slant: This is used to adjust the slant of the image due to a clock inaccuracy. This value is saved in the configuration file, so it is not needed to reenter it each time.
- Center: This is used to manually adjust the horizontal center of the image, if it could not be detected in the phasing step.
- Auto: When this button is set, the image will be automatically centered. This process starts after some hundredth of lines are received, by shifting left and right the image. It takes some time to converge.

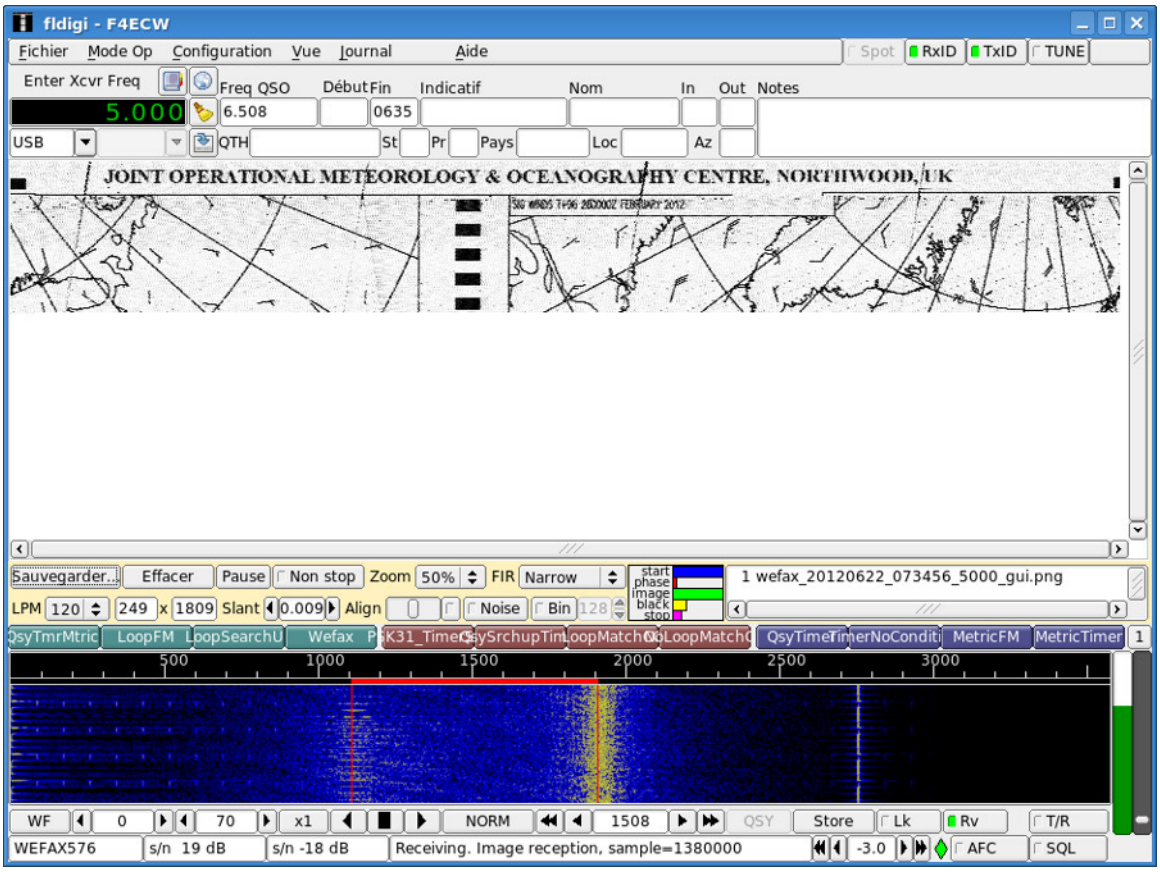


Figure 5.52: Received WEFAX File

5.14.3.1 APT control reception mode

This mode uses the APT start and STOP frequencies to detect the beginning and end of an image. Additionally, it tries to detect the phasing signal - a wide black band - to detect the center of the image. This method is helped by the estimation of the signal power of these frequencies.

5.14.3.2 Manual reception mode (Non-Stop)

In this mode, the image is continuously read and displayed. When the maximum number of lines is reached, the image is saved and blanked, and the line counter returns to one.

5.14.3.3 Input FIR filters

There are three input Finite Impulse Response filters available. Here are their frequency characteristics:

Narrow filter, the default one, give the better results.

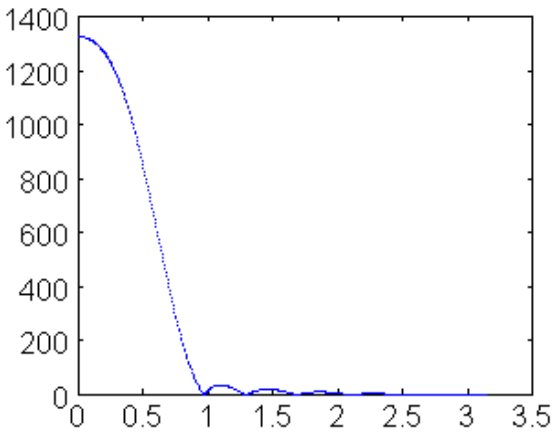


Figure 5.53: Narrow filter response

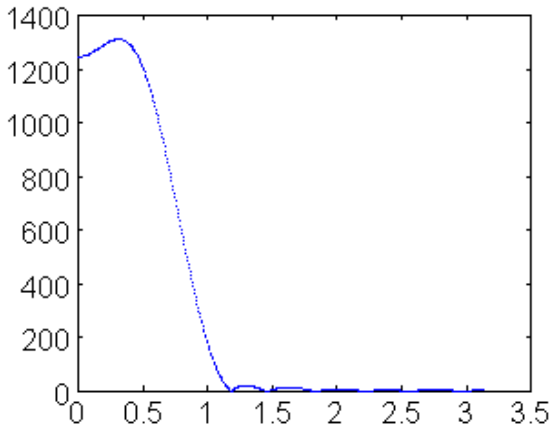


Figure 5.54: Middle filter response

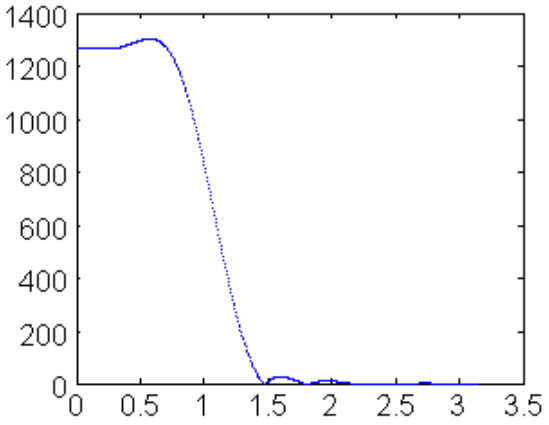


Figure 5.55: Wide filter response

5.14.3.4 Centering an image

If the phasing was not automatically detected, the modem could not deduce the beginning of an image. The result is an image which is horizontally shifted. To correct this, one can use the "Center" slider.

5.14.3.5 Picture with a slant

If either the send, receive or both ends of the transmission are using an uncalibrated sound card whose sampling rate is not an exact multiple of the sample rate the resulting picture at the receive end will appear slanted. The degree of slant is directly related to the accumulation of the frequency error at both ends of the transfer. Stations wishing to receive Weather fax pictures should calibrate their sound card.

The WWV calibration mode is used to measure and set the parts per million (ppm) correction factor for the sound card.

Your sound system may be fully corrected, but the sending station may have an uncorrected sound card. You can usually correct for small errors during reception by using the slant slider. Its value (Typically between - 0.005 and 0.005) will be stored in fldigi configuration parameters.

5.14.3.6 Automatic centering.

If the phasing signal could not be used for centering the image, the program waits for a string image signal anyway to go into reception mode, but it sets an internal flag allowing to automatically center the image. This feature can be freely enabled and disabled at any moment. it works by detecting a wide vertical band of about hundred pixels, where the sum of the contrast as the lowest among the complete image width.

That is: It computes for each row and each pixel, the absolute value of the horizontal derivate. It then sums these derivatives pixel-wise, row by row. Then, it computes an average of about hundred pixels along this single row. The column which has the lowest averaged contrast is considered to be the image margin, which is then shifted on the left of the window.

This method takes some time to stabilize, because at the beginning, there are many areas of the image, without details. It gets stable at the end, when only the margin stays with few contrasted details.

5.14.3.7 Image detection based on signal power

The APT control - inherited from the Hamfax signal, does not work very well when the image is noised. On the other hand, fldigi provides ways to evaluate the signal power on a given bandwith. This is used because APT control relies on the emission on specific frequencies.

Therefore, in the APT start and phasing loops, when check for the presence of strong signals associated to APT control. This information is used to take a decision when the traditionally method does not detect anything.

These two methods are interchangeable but used together for better detection.

5.14.3.8 AFC: Automatic Frequency Control

This option controls the frequency on the complete spectrum width. After several hunderads of lines have been correctly loaded (That is, with a high line-to-line correlation), the AFC locks until the frequency or mode is manually changed.

5.14.3.9 Noise elimination

This option eliminates short-lived noise, individual pixels with a different value from their horizontal neighbors. They are modified using median values. This is based on the fact that no line should be narrower than two pixels, otherwise such an image would not be broadcast, because not correctly readable. It is therefore impossible to have one single pixel, simultaneously very different from its left and right neighbors.

5.14.3.10 Binary images

Faxes can be stored as binary or grey level images. The cutoff level between black and white (Default 128) can be adjusted. No information is lost until the image is saved, therefore this level can be freely modified until image end. The purpose of this option is to save disk space.

5.14.3.11 Displaying received files

Each time the end of an image is detected, an image file is created and its name appears on the file list of the reception window. By clicking on a file name, it is displayed in the transmit window.

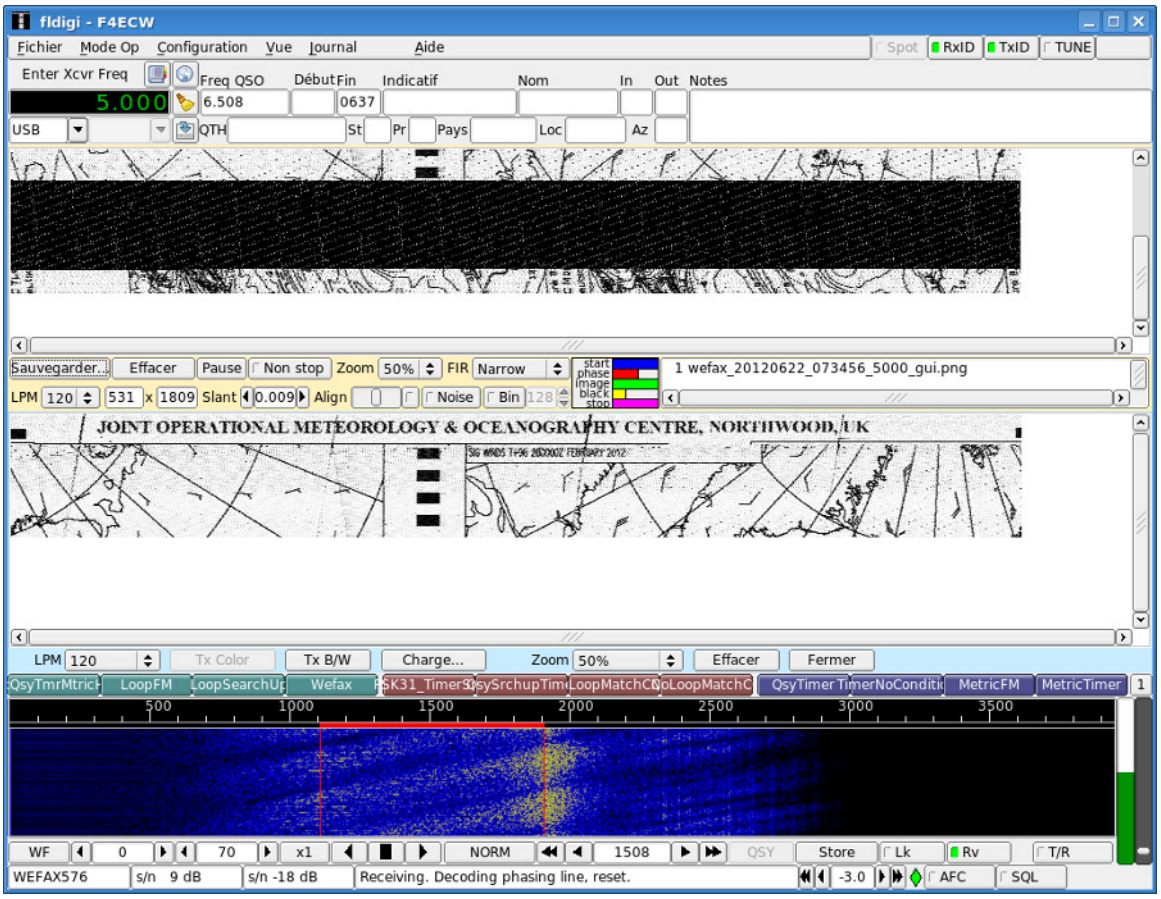


Figure 5.56: Received Image

Return to Top of Page

Return to Main Page

5.15 WWV transmit mode

The WWV mode is normally used to measure the offset of the sound card oscillator. (see WWV ppm measurement).

It can also be used to transmit time tick signals similar to the format that is used by WWV. The WWV modem sends a time tick which is a 200 msec wide pulse at a once per second rate when the T/R button is enabled. This pulse is shaped with a 4 msec raised cosine shape on the leading and trailing edges to reduce key clicks. The accuracy of the transmitted time tick is solely dependent on the accuracy with which the WWV ppm measurement was performed.

The purpose of the WWV time tick transmission is to allow other stations to calibrate their sound cards against your calibrated system. This can be used to align all systems in a VHF/UHF net for example. It is only necessary for one of the net members to be able to calibrate his or her sound card against WWV. The other's would then be calibrated by proxy use of the WWV time tick transmit mode. This can even be used in the case where no member has access to a HF transceiver. The "master" station would set the Rx and Tx ppm settings to zero. It would then transmit the time tick signal for the other stations to calibrate their sound cards against the master sound card. Having all of the stations calibrated in this way will insure that the modem decoders will give maximum performance. Here is an example of an advanced macro that will send a CW announcement, 2 minutes of time ticks and end with another CW announcement.

```
<MODEM:CW>
<!GOFREQ:1000>
<!WPM:24>
QRZ QRZ de <MYCALL> <MYCALL>
2 minute time tick cal run follows
<IDLE:2>
<!MODEM:WWV><!IDLE:120>
<!MODEM:CW><!IDLE:2>
end of time tick run
de <MYCALL> k
<TX><RX>
```

Return to Top of Page

Return to Main Page

5.16 Mode Table

5.16.1 PSK

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
PSK31	31.25	50	80.00%	31	1-PSK		31HG1B	1	
PSK63	62.5	100	80.00%	63	1-PSK		63HG1B	2	
PSK63F-EC	62.5	55	80.00%	63	1-PSK		63HG1B	22	
PSK63R-C4	63	220	80.00%	330	4-PSKR		330HG1-BC	263	1
PSK63R-C5	63	275	80.00%	416	5-PSKR		416HG1-BC	263	2
PSK63R-C10	63	550	80.00%	850	10-PSKR		850HG1-BC	263	3
PSK63R-C20	63	1100	80.00%	1725	20-PSKR		1725H-G1BC	263	4
PSK63R-C32	63	1760	80.00%	2775	32-PSKR		2775H-G1BC	263	5

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
PSK125	125	200	80.00%	125	1-PSK		125HG1B	4	
PSK125R	125	110	80.00%	125	1-PSKR		125HG1B	183	
PSK125-RC4	125	352	80.00%	650	3-PSKR		650HG1B	10	
PSK125-RC5	125	440	80.00%	825	4-PSKR		700HG1-BC	11	
PSK125-RC10	125	1100	80.00%	1700	10-PSKR		1700H-G1BC	12	
PSK125-RC16	125	1760	80.00%	2750	16-PSKR		2750H-G1BC	13	

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
PSK250	250	400	80.00%	250	1-PSK		250HG1B	126	
PSK250R	250	220	80.00%	250	1-PSKR		250HG1B	186	
PSK250-C6	250	2400	80.00%	2000	6-PSK		2000H-G1B	263	63
PSK250-RC2	250	440	80.00%	600	2-PSKR		600HG1-BC	263	20
PSK250-RC3	250	660	80.00%	950	3-PSKR		950HG1-BC	263	21
PSK250-RC5	250	1100	80.00%	1650	5-PSKR		1760H-G1BC	263	22
PSK250-RC6	250	1320	80.00%	2000	6-PSKR		2000H-G1BC	263	65
PSK250-RC7	250	1540	80.00%	2350	7-PSKR		2350H-G1BC	263	23

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
PSK500	500	800	80.00%	500	1-PSK		500HG1B	173	
PSK500-C2	500	1600	80.00%	1200	2-PSK		1200H-G1B	263	27
PSK500-C4	500	3200	80.00%	2600	4-PSK		2600H-G1B	263	28
PSK500R	500	440	80.00%	500	1-PSKR		500-HG1-BC	187	
PSK500-RC2	500	880	80.00%	1400	2-PSKR		1400H-G1BC	263	24
PSK500-RC3	500	1320	80.00%	1900	3-PSKR		1900H-G1BC	263	25
PSK500-RC4	500	1760	80.00%	2600	4-PSKR		2600H-G1BC	263	26

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
PSK800-C2	800	2300	80.00%	1400	2-PSK		2300HG-B1	263	57
PSK800-RC2	800	1280	80.00%	1400	2-PSKR		800HG-B1C	263	54

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
PSK1000	1000	1600	80.00%	1800	1-PSK		1800H-G1B	263	50
PSK1000-R	1000	880	80.00%	1800	1-PSKR		1800H-G1B	263	51
PSK1000-C2	1000	3200	80.00%	3600	2-PSK		3600H-G1BC	263	52
PSK1000-RC2	1000	1760	80.00%	3600	2-PSKR		3600H-G1BC	263	53

5.16.2 QPSK

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
QPSK31	31.25	50	80.00%	31	1-QPSK		31H0G1B	110	
QPSK63	62.5	100	80.00%	63	1-QPSK		63HG1B	3	
QPSK125	125	200	80.00%	125	1-PQSK		125HG1B	5	
QPSK250	250	400	80.00%	250	1-PQSK		250HG1B	127	
QPSK500	500	800	80.00%	500	1-PQSK		500HG1B		

5.16.3 Contestia

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
CONTESTIA-4-250	62.5	40	100.00%	250	4-FSK	-10 dB	250HF1B	55	
CONTSTIA-8-250	31.25	30	100.00%	250	8-FSK	-13 dB	250HF1B	49	
CONTESTIA-4-500	125	78	100.00%	500	4-FSK	-8 dB	500HF1B	54	
CONTESTIA-8-500	62.5	60	100.00%	500	8-FSK	-10 dB	500HF1B	52	
CONTESTIA-16-500	31.25	30	100.00%	500	16-FSK	-12 dB	500HF1B	50	
CONTESTIA-8-1000	125	117	100.00%	1000	8-FSK	-5 dB	1K00F1B	117	
CONTESTIA-16-1000	62.5	78	100.00%	1000	16-FSK	-9 dB	1K00F1B	53	
CONTESTIA-32-1000	31.25	48	100.00%	1000	32-FSK	-12 dB	1K00F1B	51	

5.16.4 DominoEX

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
DominoEX4	3.9	29	100.00%	173			173HF1B	84	
DominoEX5	5.4	44	100.00%	244			244HF1B	85	
DominoEX8	7.8	58	100.00%	346			346HF1B	86	
DominoEX11	10.8	80	100.00%	262			262HF1B	87	
DominoEX16	15.6	115	100.00%	355			355HF1B	88	
DominoEX22	21.5	160	100.00%	524			524HF1B	90	
DominoEX44	43	312	100.00%	1600			1600HF1B	263	45
DominoEX88	86	614	100.00%	1600			1600HF1B	263	46

5.16.5 MFSK

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
MFSK4	3.9	18	100.00%	154	32-FSK		154HF1B		
MFSK8	7.8	36	100.00%	316	32-FSK		316HF1B	60	
MFSK11	10.8	40	100.00%	218	16-FSK		218HF1B	148	
MFSK16	15.6	58	100.00%	316	16-FSK		316HF1B	57	
MFSK22	21.5	80	100.00%	435	16-FSK		435HF1B	152	
MFSK31	31.3	55	100.00%	330	8-FSK		330HF1B		
MFSK32	31.3	120	100.00%	630	16-FSK		630HF1B	147	
MFSK64	63	240	100.00%	1260	16-FSK		1260HF1B	263	30
MFSK128	125	480	100.00%	1920			1920HF1B	263	31
MFSK64L	63	240	100.00%	1260	16-FSK		1260HF1B	263	30
MFSK128L	125	480	100.00%	1920			1920HF1B	263	31

5.16.6 MT-63

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
MT63-500	5	50	80.00%	500	64 x 2-PSK		500HJ2D-EN	9	
MT63-1000	10	100	80.00%	1000	64 x 2-PSK		1K00J2D-EN	12	
MT63-2000	20	200	80.00%	2000	64 x 2-PSK		2K00J2D-EN	15	

5.16.7 Olivia

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
OLIVIA-4-250	63	20	100.00%	250	4-FSK	-12 dB	250HF1B	75	
OLIVIA-8-250	31	15	100.00%	250	8-FSK	-14 dB	250HF1B	69	
OLIVIA-4-500	125	40	100.00%	500	4-FSK	-10 dB	500HF1B	74	
OLIVIA-8-500	63	30	100.00%	500	8-FSK	-11 dB	500HF1B	72	
OLIVIA-16-500	31	20	100.00%	500	16-FSK	-13 dB	500HF1B	70	
OLIVIA-8-1000	125	58	100.00%	1000	8-FSK	- 7 dB	1K00F1B	116	
OLIVIA-16-1000	63	40	100.00%	1000	16-FSK	-10 dB	1K00F1B	73	
OLIVIA-32-1000	31	24	100.00%	1000	32-FSK	-12 dB	1K00F1B	71	

5.16.8 RTTY

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
RTTY 45	45	60	100.00%	270			270HF1B	39	
RTTY 50	50	66	100.00%	270			270HF1B	40	
RTTY 75	75	100	100.00%	370			370HF1B	41	

5.16.9 THOR

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
THOR4	3.9	14	100.00%	173			173HF1B	136	
THOR5	5.4	22	100.00%	244			244HF1B	139	
THOR8	7.8	28	100.00%	346			346HF1B	137	
THOR11	10.8	40	100.00%	262			262HF1B	143	
THOR16	15.6	58	100.00%	355			355HF1B	138	
THOR22	21.5	78	100.00%	524			524HF1B	145	
THOR25-X4	24.3	88	100.00%	1800	4x tone spacing, 2 sec interleave		1800HF1-B	263	40
THOR50-X1	48.6	176	100.00%	900	1 sec interleave		900HF1B	263	41
THOR50-X2	48.5	176	100.00%	1800	2x tone spacing, 1 sec interleave		1800HF1-B	263	42

THOR100	97	352	100.00%	1800	0.5 sec interleave		1800HF1-B	263	43

5.16.10 THROB

Mode	Baud	WPM	Duty Cycle	BW (Hz)	Modulation	S/N	ITU	RSID-1	RSID-2
THROB1	1	10	80.00%	72	1/2 of 9-FSK		72H0F1B	43	
THROB2	2	20	80.00%	72	1/2 of 9-FSK		72H0F1B	44	
THROB4	4	30	80.00%	144	1/2 of 9-FSK		144HF1B	45	
THROB-X1	1	10	80.00%	94	2 of 11-FSK		94H0F1B	46	
THROB-X2	2	20	80.00%	94	2 of 11-FSK		94H0F1B	47	
THROB-X4	4	40	80.00%	188	2 of 11-FSK		188HF1B	146	

Return to Top of Page

Return to Main Page

Chapter 6

Operating

6.1 Contest - How To

Fldigi supports a basic contesting format. Select the menu item View/Contest fields to see how the QSO entry entry fields change for contest data. You will see that fldigi has fields to support received and sent contest numbers as well as generic contest exchange information.

6.1.1 Contest Fields

Figure 6.1: Contest Fields

The serial number out (#Out) is automatically initialized and updated by the built-in serial number generator. You can enter the appropriate exchange information via the keyboard or mouse. Text in the Rx pane can be selected by the usual left-click-swipe of highlighting. Then right click anywhere after highlighting the desired text and a popup menu will appear allowing you to select the destination QSO field. Make your selection and the info is placed in the correct text box. Note that the popup menu changes with the QSO logging view and also with a change in "Quick entry". A full description is found in the description of operating the Logbook. The important thing to note for contest operation is that the Call and Serial # are single word captures. The Xchg capture can be either single word or multiple word (mark / right click). If the Xchg field has text contents then the new capture is appended to end of the current text in that field. That means you can point to the word representing the field, right click and select from the menu. You do not need to highlight the text for the word capture. You can very rapidly fill in the serial number and the exchange data (even if multi value) by simply pointing and right clicking on the desired word.

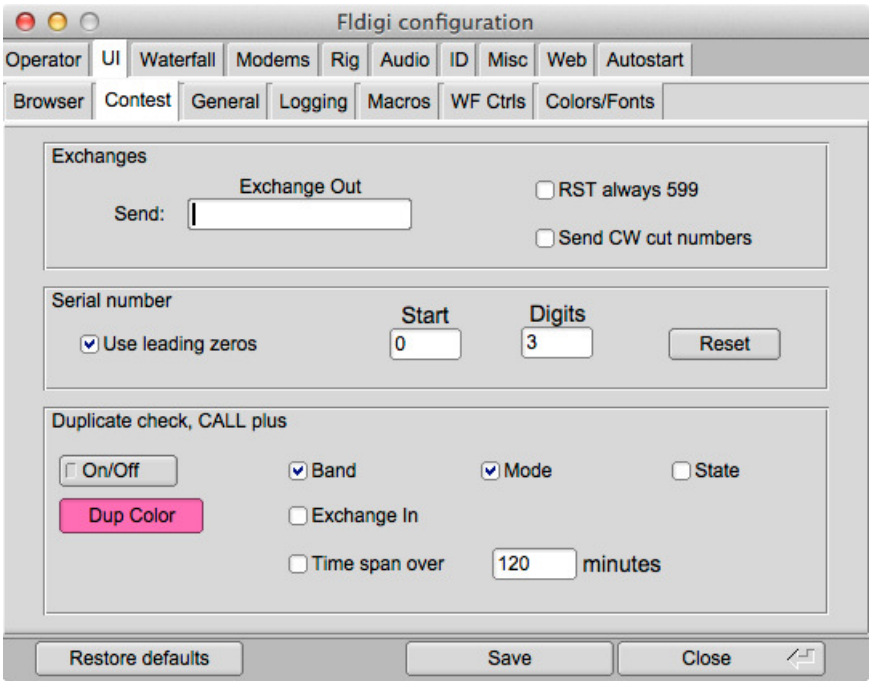


Figure 6.2: UI Contest

To set up fldigi for contesting you will need to open configure contest. the 1st row contains what info you want sent with the appropriate macro tag. ie...if the contest requires RST and name you would fill in the Exchange Out box with your name. The contents of this field are accessed from a macro with the <XOUT> tag. You will also need to check the RST always 599 box as this is the de-facto signal report in contests.

If you are participating in a CW contest you may want to select the "Send CW cut numbers", cut numbers is the norm for a CW contest. The cut numbers will send N for 9 and T for zero.

The next box contains the needed requirements to use serial numbers for a contest. You will always want to use leading zeros, start with 1 and use 3 digits. Press reset to initialize the #Out QSO field to the Start number.

Check the appropriate fields for determining if this is a duplicate call. If a duplicate is detected the Call entry will be highlighted as shown in the "Dup Color" button. Pressing this button opens a color selector so you may customize the color. There are many choices to alert you to a duplicate contact. The duplicate is based on the logical AND of all of the fields to be checked. The DUPE is cleared when you press the clear QSO log button (the brush icon).

After you have filled in all the required information, make sure you save and close.

Remember YOU MUST click the Reset button in the Serial number panel for the serial number counter to be initialized. You should also press the QSO clear button (broom) in the QSO entry widget for the other changes to take effect.

It would be best to create a new log for each contest. You create a new log by selecting the menu item File/Logs/-New logbook. The default new log name will be newlog.adif on Linux and newlog.adi on Windows. You can rename the new log file now or later by using the system file manager or when you save the log. The import/export feature of fldigi will allow you to export the log into your everyday logging software or the built-in fldigi logbook.

6.1.2 Restarting a contest session

You might have closed down fldigi in the middle of a contest, everyone needs a break now and then. You then start fldigi and want to continue the contest. Here are the steps to insure that you continue operations with no glitches.

- Load your macro file that contains your contest macros (more on that below)
- Select the menu item View/Contest fields
- Select the menu item View/Logbook
- Make sure you have the contest logbook open ... if not then this is the time to open that logbook database.

  Select the menu item "File/Logs/Open logbook..." and find your log data file.
- Look at the last record and check the serial number sent. Enter that number plus one in the Start entry on the config contest tab (see above).
- Press the Reset button in that panel.

You are ready to keep on contesting

6.1.3 Remembering a contact

If you are copying a potential contact but you are not being heard you can save fldigi's modem state using one of two methods

1. double click the signal on the waterfall
2. right click on the Rx panel and select "Insert marker"

A line of text will be inserted at the end of the Rx text buffer. It will appear similar to this:

<<2008-12-30T10:06Z BPSK-31 @ 3580000+0781>>

The date-time, the mode, the transceiver operating frequency and the audio offset will be recorded. The text line is in blue and behaves in a way that you might expect a url reference to behave in a web browser window. Work a few more contacts (even on a different band or frequency) and then scroll the Rx pane to that special divider. Left click on the line of text and fldigi will restore the transceiver to its frequency, change the mode to the saved mode and put

the waterfall cursor at the audio offset frequency. Changing the transceiver frequency will only work if you are using CAT control of your transceiver. If you are not using CAT control the mode and waterfall cursor will still be restored.

There is no limit to the number of divider lines that can be inserted into the Rx pane. They will all be removed when the Rx pane is cleared.

6.1.4 Saving the entire session

Select the menu item "File/Logs/Log all RX/TX text". If this toggle menu is checked your entire session of received and sent text will be saved to a file in the fldigi default files folder. It will be given a name synonymous with the date and time is is started, ie: fldigi20081230.log. You can review this log by selecting the menu item "File/Show config" which will open your OS default file explorer to the fldigi files folder. The file is an ASCII text file.

The format of the daily log is shown in Working Logs.

6.1.5 Contesting Macro Tips

OK, now we have fldigi setup for basic contesting, lets move on to some ideas on macros to use. I tend to make generic one size fits all macros. I recommend that you make a new macro file, mine is named contest.mdf, this will give you 48 macros to use based on the type of contest you are entering. Take a good look at the examples I have listed, you will notice there are no commas, hyphens or other extraneous items. I have seen just about every example of a poorly thought out macro there is or has ever been dreamed up. Classic examples are:

- w3nr you are 599 in Alabama your serial number is 001-001-001 how copy ??
- hello ed thanks for the call you are 599-599-001-001-001 QTH Alabama back to you

The list goes on and on. Just think, you have to try and capture the exchange, try it and you will see what I mean.

When you enter a contest you have to decide whether you are going to sit on one frequency and call CQ (Run) or are you going to tune the band looking for stations to work (S&P). So lets set up some macros that should cover both cases.

Several new macro tags have been created to facilitate contesting, these include the following tags.

<LOG>	add QSO data to the logbook & clear the QSO data fields
<CNTR>	insert current contest serial number into the text stream
<INCR>	increment contest serial number
<DECR>	decrement contest serial number
<XOUT>	contest exchange
<QSOTIME>	current log time in Zulu HHMM format
<LDT>	local date time
<ILDT>	LDT in iso-8601 format
<ZDT>	Zulu date time
<IZDT>	ZDT in iso-8601 format
<QSOTIME>	actual time of execution of the macro ... useful where exact times are used to match contest log submissions

<SAVEXCHG>	save entire contents of the expanded macro text to the "Exchange Out" field in the logbook
<XBEG>	mark the beginning of a text string that is to be saved to the "Exchange Out" field in the logbook
<XEND>	mark the end of the text string that is to be saved to the "Exchange Out" field in the logbook note: <SAVEXCHG> and the <XBEG>...<XEND> macro tags are mutually exclusive <XBEG>...<XEND> is given priority if both all three are specified in a single macro

See Macros for additional information on editing and using the fldigi macro system.

6.1.6 RUN Macros

We need just a few, starting with a CQ macro - Put this in the F1 key definition

```
<TX>
cq test de <MYCALL> <MYCALL> cq k
<RX>
```

Notice that I left 2 spaces between my call and 3 spaces at the end before the k. This will make it easier for a station to grab my call and the k on the end eliminates garbage characters before my macro finishes. The tx/rx are on separate lines as I want to be sure my macro is on a line by itself and not mixed in with screen garbage.

Now the exchange macro - Put this in the F2 key definition

```
<TX>
<CALL> 599  <CNTR> <CNTR>  <X1>  <X1> <CALL>  k
<RX>
```

Why do I have his call at the beginning as well as the end, to make sure I have copied his call correctly. You will also see that I have not as yet logged the contact,why, well are you sure he does not need to correct his call or ask for a repeat.

You are asked to repeat the exchange, you can just re-send the exchange macro, this verifies all of the information. Now he sends you his info and if you have copied it correctly you need a TU macro. - Put this in the F3 key definition.

```
<TX>
qsl tu qrz test  <MYCALL>  k
<RX><LOG><INCR>
```

Here we have done all the necessary items to complete the exchange. Notice that I did not log the contact until after everything was correct. I have fldigi set to clear on save, so when the <LOG> part of the macro executes the QSO area is cleared.

Thats the end of my RUN macro setup, told you it was rather simplistic and generic.

6.1.7 S & P Macros

I rarely if ever use S&P, but there are times I need to, especially if my QSO rate drops while running. Again the macros are very generic with only the needed info. If band conditions warrant you may want to send your call 3 times. Put this in the F5 key definition

```
<TX>
<MYCALL>  <MYCALL>  k
<RX>
```

Why just my call ?? Well I assume the other guy already knows his call !

The exchange macro is basically the same as the RUN macro. Put this one in the F6 key definition

```
<TX>
599  <CNTR>  <CNTR>  <X1>  <X1>    k
<RX>
```

As you see I have not as yet logged the QSO or incremented the serial number. This is the final S&P macro. Put this one in the F7 key definition.

```
<LOG><INCR>
```

Now this is the most important macro you will ever need.....trust me. Put it where you won't fail to find it. How about F9 ?

```
<TX>
agn  agn  k
<RX>
```

You will see that it is used many times during a contest, especially with weak stations and heavy QRN/QRM.

<QSOTIME>

- time sent in Tx stream
- repeat execution of <QSOTIME> before a <LOG> macro or a save to log button press will resend the original time
- <LOG> macro or a save-to-log button press appends the QSOTIME to the STX_STRING field in the adif log record and clears the QSOTIME.

<XBEG>

- use at end of a contest exchange to save the entire exchange string in STX_STRING
- usurps QSOTIME if both are contained in same macro text, ie: "<RST> <CNTR> <QSOTIME><SAVEXCHG>" will send an exchange as 599 024 1125 if RST = 599, Counter = 024 and time of execution is 1125
- repeats the same as <QSOTIME >
- <LOG> macro or a a save-to-log button press saves the associated macro text (after expansion). QSOTIME and the saved exchange text are cleared after the save occurs.

An example of the SAVEXCHG macro tag

```
<RST> <CNTR> <XOUT> <QSOTIME><SAVEXCHG>
```

Where RST = 599, CNTR = 0125, XOUT = AL, QSOTIME = 1433

Will save this string to the *Exchange Out* field in the logbook: "599 0125 AL 1433"

Please note that you should not include any text or macro tags that are not to be a part of *Exchange Out*. If your macro had this:

```
<TX><CALL> UR <RST> <CNTR> <XOUT> <QSOTIME> de <MYCALL> k<RX><SAVEXCHG>
```

Where CALL = W3NR, MYCALL = W1HKJ

the saved *Exchange Out* field would contain: "W3NR UR 599 0125 AL 1433 de W1HKJ k"

Probably not what you want. Use separate function keys for the "<TX>CALL ..." and the "de <MYCALL> k<RX>" or use the next set of macro tags

```
<XBEG>...<XEND>
```

These two macro tags are delimiters for capturing the transmitted exchange data, for example:

```
<TX><CALL> de <MYCALL> QSL <XBEG><RST> <CNTR> <QSOTIME><XEND> K<RX>
```

Will place the expanded <RST> <CNTR> <QSOTIME> into the *Exchange Out* field of the logbook when the contact is saved. This is much better illustrated with a screen shot. This one shows the macro editor contents, the logbook entry in *Exchange Out*, and the transmit text buffer.

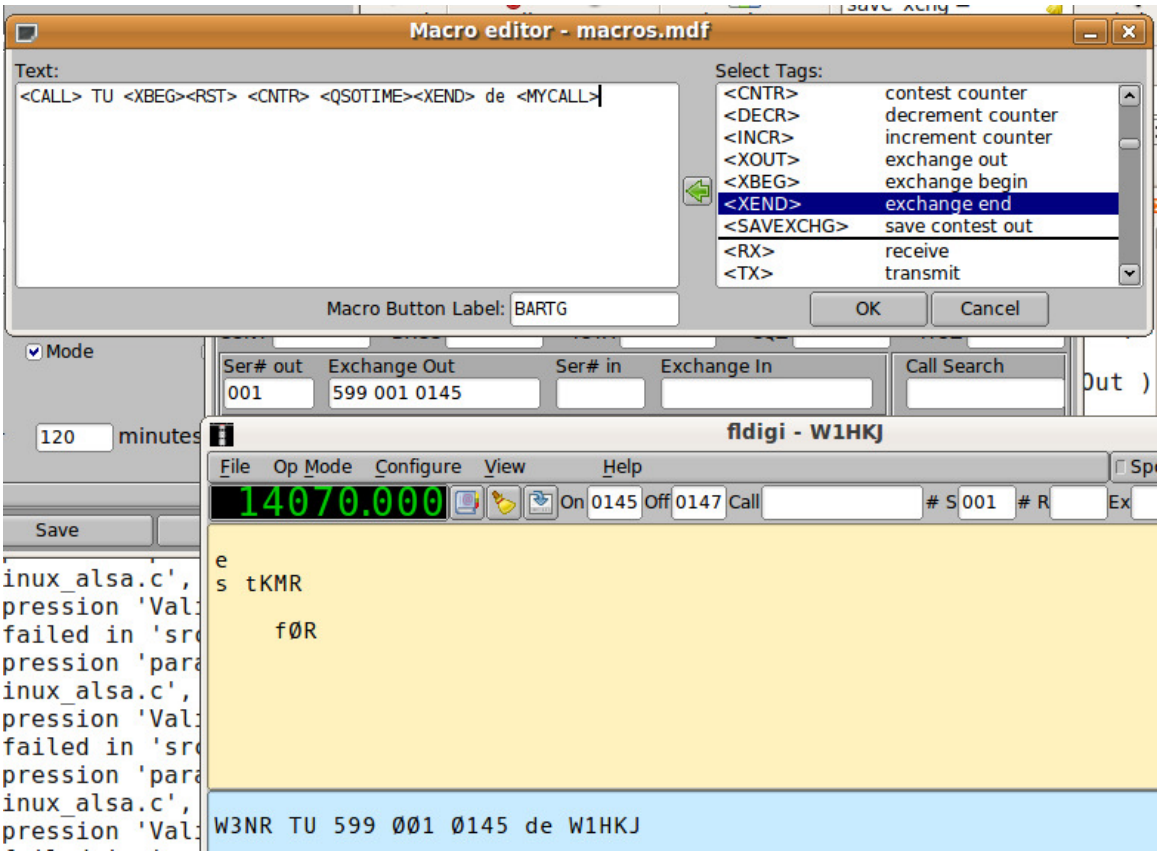


Figure 6.3: Exchange Begin-End

Return to Top of Page

Return to Main Page

6.2 CW Keying

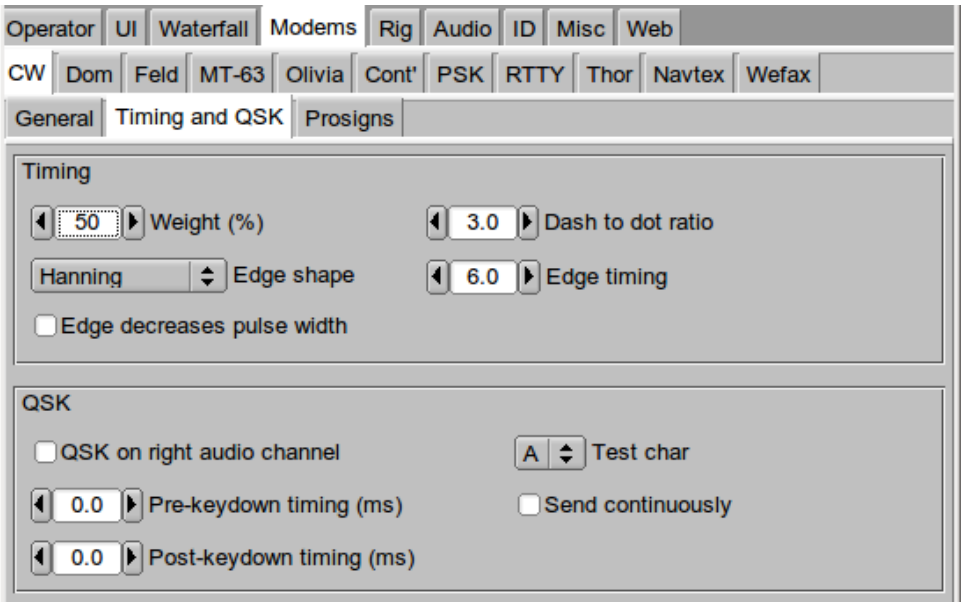


Figure 6.4: Timing and QSK

Click on the Configure menu item to open the *fldigi* configuration dialog. Click on the Modems tab and then on the QSK tab.

1. Set the Pre Timing and Post Timing to zero.

2. Activate the QSK feature by clicking "QSK on right channel".

Click the CW tab and adjust the CW settings to your preference. Use the Test char and the Send continuous controls to ease the adjustment process.

FLdigi is now ready to generate a 1600 hertz CW tone on the right channel of the stereo audio out of your sound card.

The left channel will be the normal raised cosine shaped CW wave form that you may use for your side tone.

The following circuit may be used to take the FLdigi QSK OUT signal from the right channel of your SOUND CARD to key your transmitter or a QSK circuit.

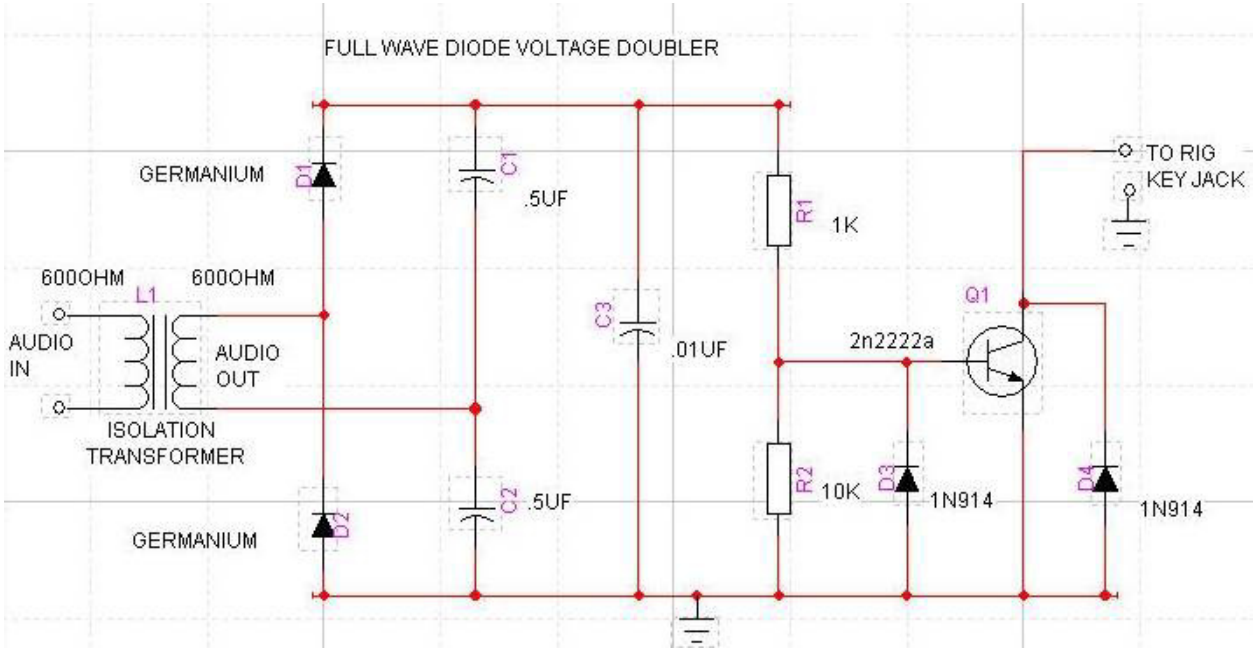


Figure 6.5: CW Keying Circuit

NOTE: L1 - Radio Shack has two items that may be used for this isolation transformer.

- Catalog # 270-054
- Catalog # 273-1374

Attach an audio cable from the Rt. Channel out of the your computer's SOUND CARD to the input of this QSK INTERFACE CIRCUIT (input of L1).

Attach another cable from the output of this circuit to your Rig's Keying Jack.mall>

Every CW tone that is generated by FLdigi is rectified by this FULL WAVE VOLTAGE DOUBLER circuit. The resultant voltage turns the Q1 transistor on and "grounds" the collector, which takes the RIG'S CW KEYING JACK to ground and "keys" the transmitter.

You can adjust the start and stop timing of the QSK circuit relative to the CW waveform with the "pre" and "post" settings.

Return to Top of Page

Return to Main Page

6.3 Digiscope Display

Fldigi provides several different views of the decoded signal with its waterfall, text and a scope displays. The scope display is either a separate moveable, resizable dialog that is opened from the "View/Digiscope" menu item or a docked scope.

6.3.1 CW

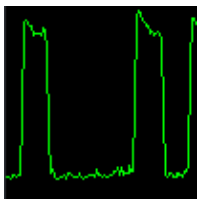


Figure 6.6: Digiscope CW

The CW signal will consist of the time domain amplitude detected signal. The horizontal timing is dependent on CW speed, so that the display will appear similar independent of CW speed.

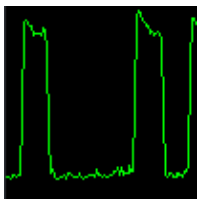


Figure 6.7: Digiscope CW

6.3.2 DominoEX / Thor

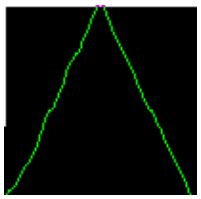


Figure 6.8: Digiscope DominoEX & Thor

DominoEX and Thor have two alternate views available on the digiscope display. You can toggle between the views by left clicking on the digiscope display area. The triangular view shows data propogation through the interleave filter. As signal s/n degrades this display will become more wavy.

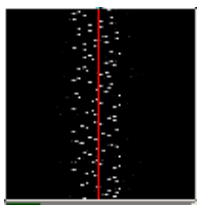


Figure 6.9: Digiscope DominoEX & Thor

The second view is the decoded data stream viewed in the frequency domain. The dots will be very distinct when the signal is fully acquired and decoding properly. It will be fuzzy when the decoder is not locked or there is interference present.

6.3.3 MFSK

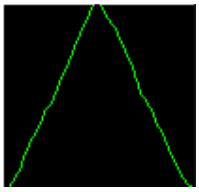
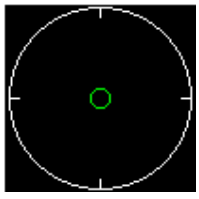


Figure 6.10: Digiscope MSFK

This is what you expect to see for all of the MFSK type modes. The number of steps in the slant lines will change with the various modes, but they will all have the same general appearance. If the signal is mistuned the sloped lines will become bowed and distorted.

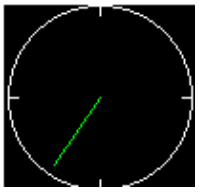
6.3.4 PSK

The digiscope display just to the right of the waterfall displays signal quality in various formats. The display for PSK modes is the vector scope:



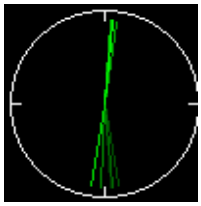
(a) The display with no signal or below squelch level. If the SQL is off this display will be random vectors driven by noise.

(b) The display with a normal psk31 signal. The vector flips between 0 and 6 o'clock

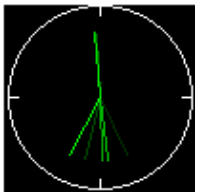


(c) AFC off and receive carrier set below the center of the received signal

(d) AFC off and receive carrier set above the center of the received signal.



(e) AFC enabled, Fading History Display Mode Selected (left click on scope)



(f) AFC enabled, Fading History / Amplitude Display Mode Selected (2nd left click on scope)

Figure 6.11: PSK Digiscopes

You can see the effect of mistuning by slewing the carrier carrier control moving from low to high over the signal . You must do this with AFC off. Engage the AFC and the vectors will immediately snap to vertical positions.

You can alter the appearance of the phase vectors by left clicking on the digiscope display. One click will give you a history of phase vectors that fade with time. A second click will give you a history of phase vectors that both fade with time and are amplitude significant. The third click returns you to the original phase vector display.

The effect is the same with QPSK signals except you will see 4 vectors that are 90 degrees from each other.

6.3.5 RTTY

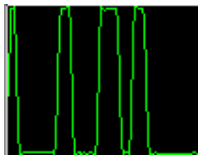


Figure 6.12: Digiscope RTTY

The signal can be viewed in two different ways on the digiscope. This is the time domain representation of the detected FSK signal. The two yellow lines represent the MARK and SPACE frequencies. This display is for Baudot, 45.45 baud, 182 Hz shift. If the transmitting station were transmitting at 200 Hz shift the signal extremes would lie above and below the yellow lines. Try tuning across the RTTY signal with the AFC disabled. You will see the signal move above and below the yellow lines as you tune. Then enable the AFC and the signal should rapidly move into the center region of the display. This signal was about 3 - 6 dB above the noise floor. It looked marginal on the waterfall but still gave good copy.

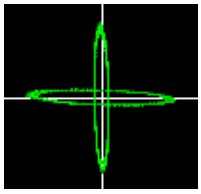


Figure 6.13: Digiscope RTTY

This is the other digiscope display for RTTY. You obtain this view by left clicking anywhere in the digiscope display window. You can toggle back and forth between these views. The MARK / SPACE frequencies are represented by the quadrature ellipses. When the RTTY signal is properly tuned in the lines will be in quadrature and aligned as shown. Tune across the RTTY signal and the MARK/SPACE lines will rotate around the center. If the sending station is using a shift that is smaller than you have the decoder setting then the two lines will close toward the NW/SE quadrants. If the sending station is using a shift that is greater than the decoder setting then the two lines will close toward the NE/SW quadrants.

Return to Top of Page

Return to Main Page

6.4 Digiscope Display - WWV mode

The WWV mode is used to measure the offset of the sound card oscillator. It does this by comparing the timing loop for the sound card measurements against the clock tick signal that is transmitted by WWV and WWVH. The sampling rate for the sound card should be set to "native". The sound card samples the signal and returns the values in 512 blocks. This block sampling is what sets the basic timing mechanism for the thread that reads the sound card, sends data to the waterfall, and sends data to the modem signal processing functions. A process of filtering is used that simultaneously reduces the sampling rate. Most modern soundcards will use 44100 or 48000 as the native smampling rate. That sample rate in down converted to 1000 using a decimation in time type FIR. The resulting signal is then power detected and further filtered with a filter called a moving average filter. The moving average is very good at detecting the edge of a pulse such as the 1 second tick transmitted by WWV. This output is then displayed in a manner very similar to a FAX signal. Each scan line represents the received signal over a 1 second interval. The bright white line is the time tick. You can see a very slight slope from left to right as the signal goes from top to bottom of the display.

Open the configure dialog box to the "SndCrd" tab. You are going to be adjusting the "Rx corr Rate" while you observe the effect of this control on the slope of the time tick line.

Tune in WWV or WWVH on 2.5, 5.0, 10.0 or 15.0 MHz in the AM mode. This seems to give the best signal view. Select the WWV modem and allow the data to begin to accumulate in the digiscope display. When you can clearly see the bright tick line, move the cursor to the bottom of the line and left click at that position. That will resync the digiscope display and put the ensuing tick marks at the center line red graticule.

Then right click anywhere in the digiscope display. That changes the zoom level to show more detail regarding the slope of the time tick line. The zoom level increases by a factor of 5. Right clicking again restores the original zoom level. I recommend making the adjustments to the Rx corr Rate control in the x5 zoom level.

If the slope of the time tick line is positive you will need to apply a negative value to the Rx corr Rate. If it is negative then a positive correction is needed.

Start with a correction of 0 ppm and observe the slope. Try a value of 1000 ppm and observe the slope. Again, try a -1000 ppm correction and observe the slope. The following are some observations made on 10 MHz WWV, DCF-77 and RWM under less than ideal conditions.

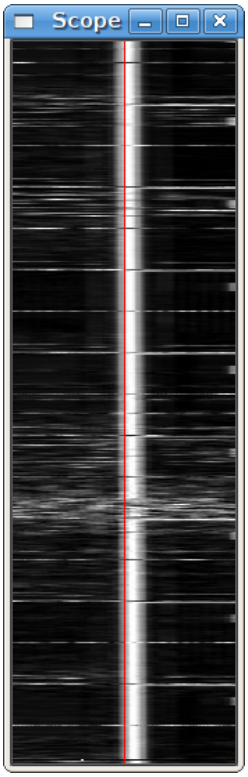
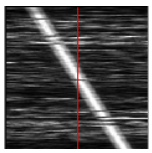
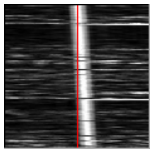


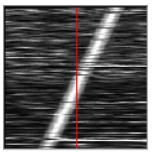
Figure 6.14: WWV corrected 20 minute trace 5x scale



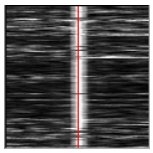
(a) 0 ppm WWV 5x scale



(b) +1000 ppm WWV 5x scale

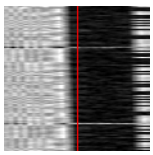


(c) -1000 ppm WWV 5x scale

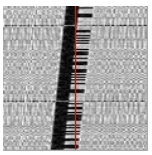


(d) +120 ppm WWV 5x scale

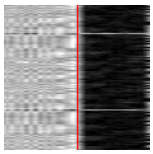
(e) 0 ppm DCF-77 1x scale



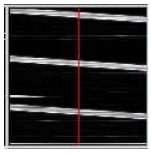
(f) 0 ppm DCF-77 5x scale



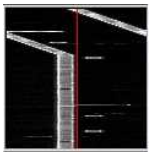
(g) +1000 ppm DCF-77 1x scale



(h) +65 ppm DCF-77 5x scale



(i) RWM uncorrected 1x scale



(j) RWM +25361 ppm 1x scale

(k) RWM +25361 ppm 5x scale

Figure 6.15: PSK Digiscopes

You can see that my sound card requires a positive correction since the slope is negative with a 0 ppm entry. The required correction of +120 ppm was determined by guessing the needed correction to be close to 1/10 of the -1000 ppm slope and then adjusting for a steady track along the red graticule. The DCF-77 images were provided by Walter, DL8FCL. The RWM images were provided by Andy G3TDJ.

You can left click on the tick line anytime you want to recenter the signal. That will aid in making your visual observation.

When you are finished, the Rx corr Rate entry is the correct one for your sound card. Save the configuration for future fldigi use.

Andy also provided information on the RWM transmissions:

RWM details extracted from http://www.irkutsk.com/radio/tis.htm

Station RWM - Main characteristics

Location: Russia, Moscow

55 degr. 44' North , 38 degr. 12' East

Standard frequencies : 4996, 9996 and 14996 kHz

Radiated power: 5kW on 4996 and 9996 kHz; 8kW on 14996 kHz

Period of operation: 24 hours per day, except 08.00-16.00 msk for maintenance as below:

on 4996 kHz : 1st wednesday of the 1st month of quater;

on 9996 kHz : 2nd wednesday of the 1st month of the quater;

on 14996 kHz : 3rd wednesday of each odd month;

Coverage: 20 degr. - 120 degr. East

35 degr. - 75 degr. North

Time signals A1X are given every second of 100 ms duration with a frequency of 1 Hz. Minute pip is extended to 500 ms.

Hourly transmission schedule

m:s - m:s

00:00 - 07:55 – MON signals (no modulation)

08:00 - 09:00 – transmitter is signed off

09:00 - 10:00 – station's identification is sent by Morse Code

10:00 - 19:55 – A1X signals and identification of DUT1+dUT1

20:00 - 29:55 – DXXXW signals

30:00 - 37:55 – N0N signals (no modulation)

38:00 - 39:00 – transmitter is signed off

39:00 - 40:00 – station's identification is sent by Morse Code

40:00 - 49:55 – A1X signals and identification of DUT1+dUT1

50:00 - 59:55 – DXXXW signals

See Transmitting Simulated WWV Timing Tone

Return to Top of Page

Return to Main Page

6.5 DXCC List etal

Fldigi uses several data files that are not included with the distribution. These must be downloaded from the list maintenance web sites for the most current data. These lists include:

List Data	List Name	Web source
DXCC	cty.dat	http://www.-country-files.com/cty/
LOTW	lotw1.txt	http://www.hb9bza.-net/lotw/lotw1.txt
EQSL	AGMemberList.txt	http://www.eqsl.cc/QS-Lcard/DownloadedFiles/-AGMemberList.txt

These files should be downloaded and placed in the fldigi files directory. The most convenient way to open the fldigi files directory is via the menu item "File / Show config".

The DXCC list browser is shown by selecting the menu item "View / Countries".

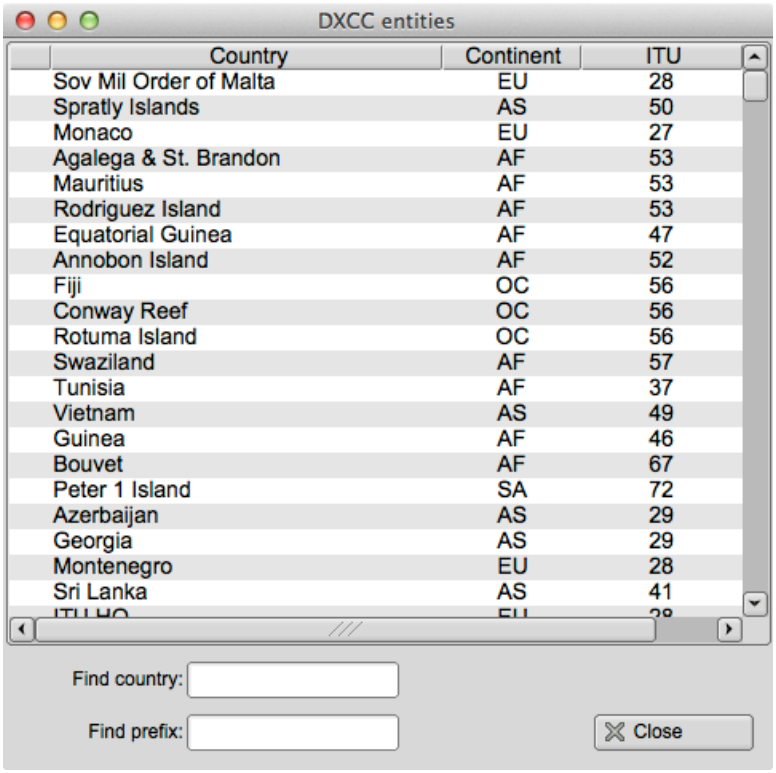


Figure 6.16: DXCC List

You can sort the list by Country, Continent, ITU or CQ zone by clicking on the various column headers.

Return to Top of Page

Return to Main Page

6.6 Frequency Analyzer

Fldigi can be used to accurately measure the frequency of a remote signal that is transmitting a steady carrier.

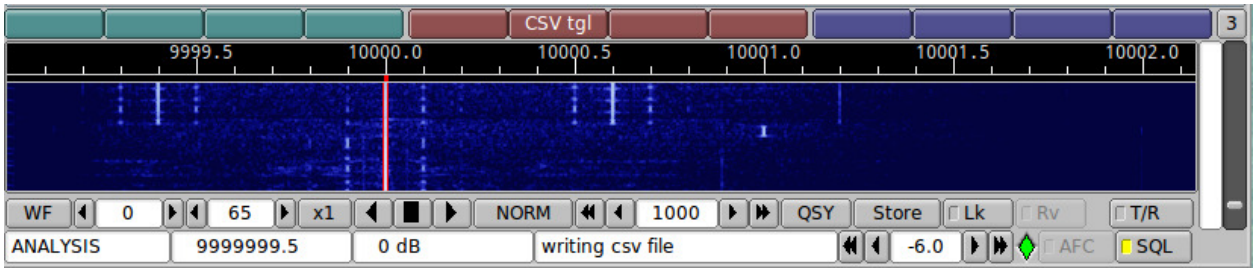


Figure 6.17: Frequency Analyzer

I have set the sound card up using the WWV modem and have it adjusted for the proper PPM offset on receive. I followed the frequency calibration procedure for the FT-950, adjusting the WWV carrier signal for exactly 1000 Hz with the transceiver tuned to 9999.0 MHz. The FT-950 provides the user with a digital adjustment to the reference frequency, Menu 035.

Then fldigi was used in the "Freq Analysis" mode to track the WWV carrier at 10 MHz. In this mode the decoder measures the frequency of the marked signal using DFT-based parameter estimator as described by Tsui and

Reisenfeld in the January 2006 issue of the Jornal of Telecommunications and Information Technoloby, "A highly accurate DFT-based parameter estimator for complex expontials." The digiscope is not used. During the measurement both the estimated signal and it's s/n will be displayed in the two left two status bar report controls. The 3rd status bar control will be blank or indicate that the measurements are being recorded to a file, "writing csv file."

The data file is named "analysis.csv" and is written to the files temp directory. The format is:

Clock	Error -	-1	1	Audio	RF
1	0.06	-1	1	1000.06	10000000.0602
2	0.043	-1	1	1000.043	10000000.0434
3.1	0.04	-1	1	1000.04	10000000.0398
4.1	0.03	-1	1	1000.03	10000000.0298
5.1	0.027	-1	1	1000.027	10000000.0274
6.1	0.022	-1	1	1000.022	10000000.0223
7.2	0.018	-1	1	1000.018	10000000.0177
8.2	0.013	-1	1	1000.013	10000000.0132
9.2	0.009	-1	1	1000.009	10000000.0093
10.2	0.004	-1	1	1000.004	10000000.0040
11.3	0.001	-1	1	1000.001	10000000.0008
12.3	-0.002	-1	1	999.998	9999999.9978
13.3	-0.006	-1	1	999.994	9999999.9944

Clock is in seconds for the start of the recording

Error is in Hz, from the selected waterfall frequency

The +/-1 are simply data point to make it convenient to display a graph in a spreadsheet program.

Audio is the estimated audio tracking frequency in Hz.

RF is the estimated radio frequency tracking frequency in Hz.

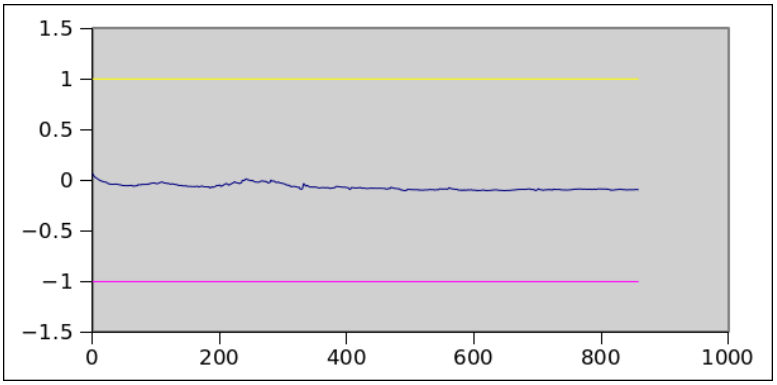


Figure 6.18: Frequency Analysis

The analysis.csv file will be cleared and restarted every time that the waterfall tracking point is changed by the operator, using either the WF cursor left click, or the WF frequency selection control.

A new macro tag has been added to assist the operator to manage the recording:

<CSV:on|off|t>

on - restart file,

off - stop recording,

t - toggle on/off

ARRL frequently announces a frequency measurement test (FMT) which takes place on 160, 80 and 40 meters. This is a chance to test your skills in frequency measurement. You should be able to make a submission to the FMT using this technique. Make corrections to the FMT transmission based upon your WWV measurement. You may have to adjust for other local oscillator effects as well. If you have some good ways to measure and correct for these

I would be glad to share them with the other fldigi users.

Return to Top of Page

Return to Main Page

6.7 Keyboard Operation

The transmit buffer for fldigi is type ahead which means that you can be typing text while the program is sending an earlier part of your transmitted message.

Newly entered text appears in black and text which has been transmitted is changed to red. You can backspace into the red area. When you do and the modem in use supports the BS character it will be sent to the receiving station. If you monitor PSK and MFSK signals you will often find operators backspacing over previously sent text. It's probably just as easy to just send XXX and retype that part of the message, but we have gotten used to word processors, email, etc. that allow us to send perfect (right) text, so we expect our digital modems to do the same. Let's see, what was that prosign often used in CW for oooops.

All of the alpha numeric keys perform as you would expect, entering text into the transmit buffer. There is one very important exception:

The caret "^" symbol. This is used in the macro expansion routine and also used by the transmit buffer evaluator. A ^r puts fldigi into receive mode. So you can enter the ^r (caret followed by the r) at the end of your transmit buffer and when the sent character cursor (red chars) gets to that point the program will clear the text and return to the receive mode.

You can load the transmit buffer with any ASCII Text file of your choice. Merely right click in the buffer window and select from the pop-up menu. You can also short cut to the ^r from this popup.

Many ops (including me) do not like to be tied to a mouse. The fldigi text widget supports some short cuts to make your life easier:

- Pause/Break - a transmit / receive - pause button.
  - if you are in the receive mode and press the Pause/Break key the program will switch to the transmit mode. It will begin transmitting characters at the next point in the transmit buffer following the red (previously sent text). If the buffer only contains unsent text, then it will begin at the first character in the buffer. If the buffer is empty, the program will switch to transmit mode and depending on the mode of operation will send idle characters or nothing at all until characters are entered into the buffer.
  - if you are in the transmit mode and press the Pause/Break key the program will switch to the receive mode. There may be a slight delay for some modes like MFSK, PSK and others that require you to send a postamble at the end of a transmission. The transmit text buffer stays intact, ready for the Pause/Break key to return you to the transmit mode.
  - Think of the Pause/Break key as a software break-in capability.
- Esc -
  - Abort transmission. - immediately returns the program to receive, sending the required postamble for those modes requiring it. The transmit buffer is cleared of all text.
  - Triple press on Esc - terminates the current transmission without sending a postamble - The PANIC button.
- Ctrl-R will append the ^r (return to receive) at the end of the current text buffer.
- Ctrl-T will start transmitting if there is text in the transmit text window.
- Alt/Meta-R will perform the same function as the Pause/Break key
- Tab moves the cursor to the end of the transmitted text (which also pauses tx). A tab press at that position moves the cursor to the character following the last one transmitted. CW operation is slightly different, see the help for CW .
- Ctrl + three digits will insert the ASCII character designated by that entry.

6.7.1 Function Keys

Keys F1 through F12 are used to invoke the macro F1 - F12. You can also just click on the macro key button associated with that function key. There are 4 sets of 12 macros. If you press the numbered button on the macro button bar the next set of macros are referenced by the F1 - F12. A right click on the numbered button provides a reverse rotation through the 4 sets of macro keys. The respective macro set can be made available by pressing the Alt-1, Alt-2, Alt-3 or Alt-4 key combination. Note that this is not Alt-F1 etc.

Return to Top of Page

Return to Main Page

6.8 KML

Figure 6.19: KML Logo

Section data_source Data sources

Keyhole Markup Language

(KML) is an XML file format for geographic visualization in two-dimensional maps such as Google Maps and three-dimensional earth browsers such as Google Earth or Marble.

Fldigi can generate data with geographical locations, which can be used to generate KML data. This list might expand in the future

- The emitting station of a Navtex message.
- The origin of a SYNOP weather report.
- The Maidenhead locator of the user, as entered in fldigi user's profile.

6.8.1 KML generation from Navtex messages

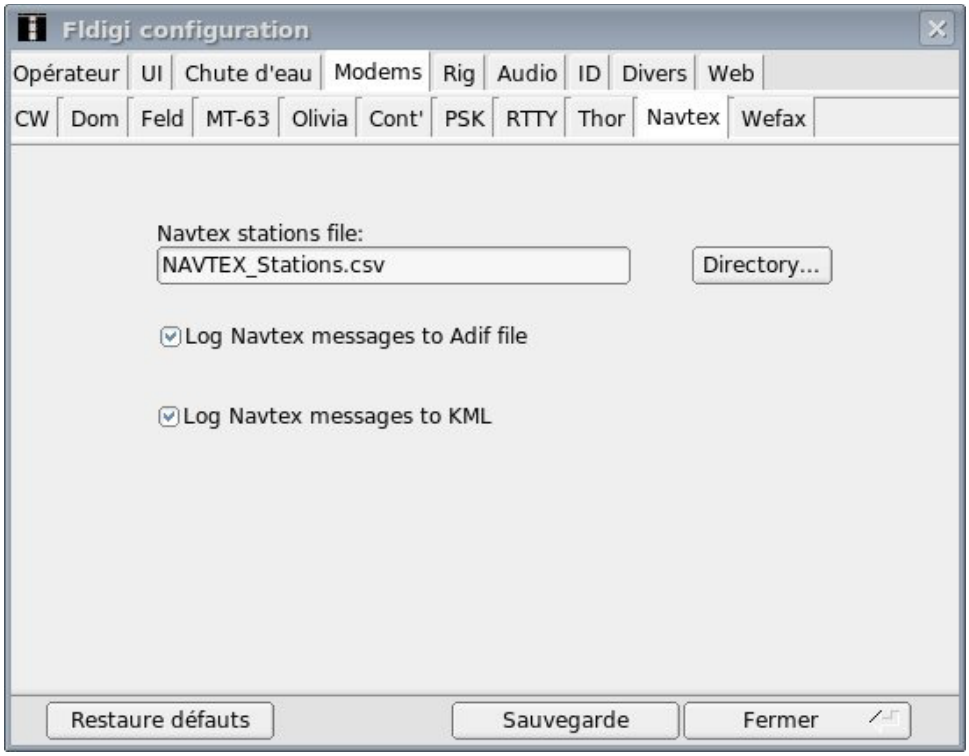


Figure 6.20: Navtex configuration tab with KML option

Each Navtex message comes with the code of the sending station, also called *origin*.

These messages are displayed, in KML files, at the coordinates of the sender. That is: KML placemarks are created or updated with these coordinates. Fldigi parses the Navtex reports, uses the station identifier to make a lookup in the Navtex stations file which contains geographical coordinates. These coordinates are used to create KML placemarks.

More explanation about how station coordinates are used, are given at the Navtex page.

Navtex messages are quite often sent with embedded coordinates of the event they describe (Ship wreck, oil exploration etc...). For example: "`LIGHT BUOY MARKING DANGEROUS WRECK 58-01.2 NORTH 005-27.1 WEST`" or "`THREE MEN OVERBOARD IN PSN 39-07,7N 026-39,2E`" . A future version will parse the content of the message, extracting raw coordinates, and will display a graphic entity at the location of the described event.

6.8.2 KML generation from SYNOP reports

`SYNOP` is a code used for reporting weather information and as such, is used to broadcast meteorological data by radio. One of the most important emitter is `Deutsche Wetterdienst which transmits them in RTTY`, and fldigi is able to decode them and generate KML placemarks at the location of the weather information.

6.8.3 KML files structure

The KML data are made of different files

fldigi.kml	Entry point. Only this one has to be loaded. It never changes.
styles.kml	KML style sheet. Freely changeable by the user, for example to customize the icons.
User.kml	Location of the user based on his/her `Maidenhead locator`.
Synop.kml	Synop weather reports displayed at the location of the WMO station, or ship, or buoy.
Navtex.kml	Navtex reports, displayed at the place of the emitting station. A future version will plot the position of the coordinates indicated in the Navtex messages themselves.

6.8.4 Extended data

When creating a new placemark, written in of the KML data files (`Synop.kml`, `Navtex.kml` etc...) data are sent to the KML module in the form of key-value pairs and are written into two forms:

- HTML content, in the `<description>` tag, surrounded by `CDATA` directives. The HTML format is chosen exclusively for display purpose and might change at any new version.
- Regular `<ExtendedData>` XML tags: These data are internally used by Fldigi to reload the previous session. The format is stable and can be used by external applications. All useful data are saved.

6.8.5 Parameters

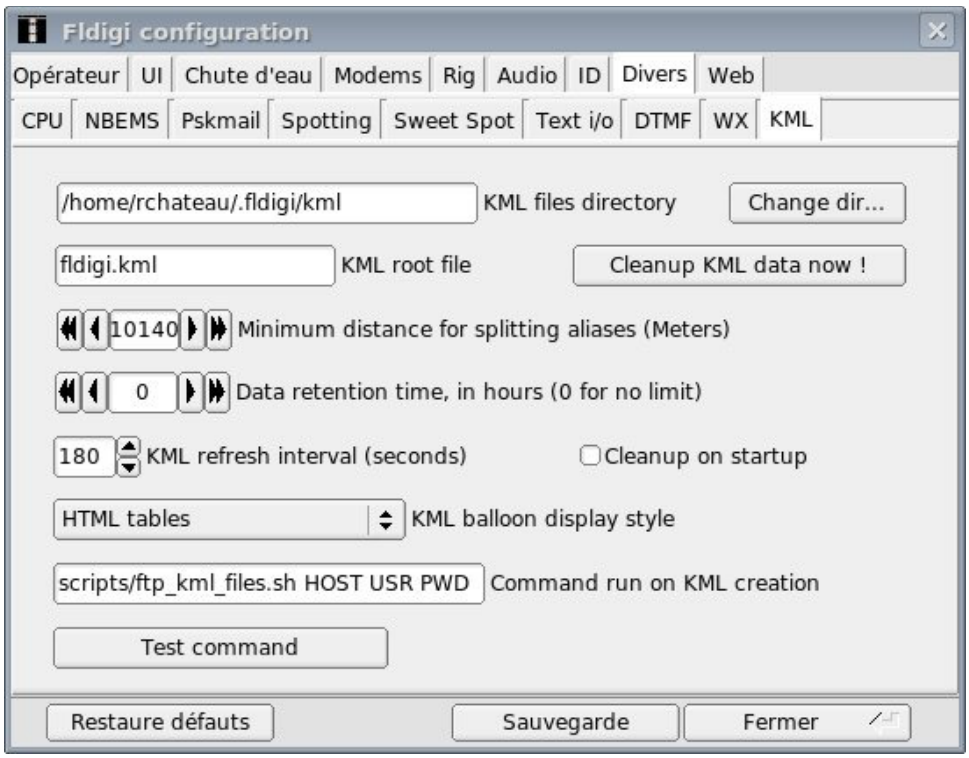


Figure 6.21: KML configuration tab

Fldigi maintains in a internal container, a set of placemarks which are data associated to geographical coordinates, an unique name, a set of key-value pairs and a timestamp. At regular intervals, a thread is woken up to save these

geographical data to a KML file, in a specific directory. At this moment, a process can be started, running an external command. Depending on the type of data, a given file name will be used.

All KML files are accessible from an unique KML filename. Placemarks are identified with an unique name, for example a vessel name, or their WMO identifier. Placemark with a moving position such as ships, can have their path visualized because they still cen be identified in two different reports. These reports can be kept as separate, or they can be merged into a single placemark: This depends on the distance between two placemarks with the same name, compared to the merging distance parameter.

Data can be kept for a given retention time, after which delay they are purged. At startup, former KML data can be reloaded, or cleaned up. Data as key-value pairs associated to a given placemarks can be displayed several ways.

All these parameters are controlled by the KML configuration tab.

6.8.6 Destination directory

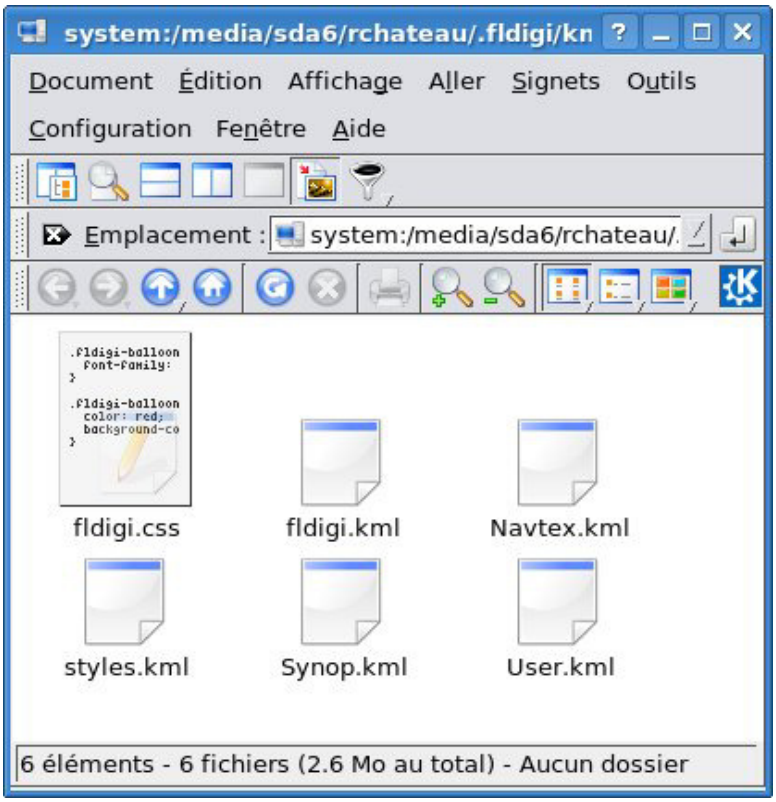


Figure 6.22: Directory of generated KML files

The default destination directory where KML files are saved is a subdirectory called /kml in the fldigi users directory. For example on Linux: `$HOME/.fldigi/kml/` and `<defaultpath>/fldigi.files/kml` on Windows™. This destination can be freely changed.

The file `fldigi.css` is created at installation, and is not changed later. Therefore it is possible to customize it by adding specific icons.

The file `fldigi.kml` is created by fldigi when it is not there, or when the refresh interval is changed.

If this destination directory is accessible from the internet, then it can be published to Google Maps.

Note:

Files updates are atomic. This means that a file is not accessible by a reader until it is completely written and closed. This is achieved by writing into temporary files, which are atomically renamed (POSIX function `rename()`) at the end of operation.

Therefore, the KML destination directory can safely be accessed by one writer and multiple readers. Several sessions of fldigi might also updates different KML files, as long as the main `fldigi.kml` file is not changed.

6.8.7 KML root file

This is the default name of the entry file of the generated KML document, which by default is fldigi.kml. If it does not exist, it is generated with the list of possible source of KML data (Synop, Navtex etc...). If `Google Earth` or `Marble` are installed on your machine, then they are associated to the file extension .kml and you just need to click on fldigi.kml to visualize it. It is automatically refreshed when fldigi adds new Synop weather reports or Navtex messages to it.

6.8.8 KML refresh interval

This delay, in seconds, is used at two places:

- This is the frequency at which new KML files are created, if new data is available
- This is the refresh interval specified in the KML file with tag <refreshInterval>.

This should not be too small, especially if the data files are big, otherwise fldigi will spend most of its time refreshing KML data, and accordingly Google Earth or Marble, reloading them.

6.8.9 Cleanup on startup

By default, at startup, fldigi reloads the existing KML files, extracting the key-value pairs contained in the "Extended-Data" tags. However, it is possible to force fldigi to restart from scratch.

6.8.10 Merging distance

Different reports with the same placemark name can be merged into a single report if their distance is below a given threshold which is the merging distance. Otherwise, separate placemarks are created and joined by a red line, visible in the KML document.

6.8.11 KML balloon display type

Reports are inserted in the KML document one after the other. These description data are visible as *KML balloons*, or when getting placemark properties. If they have the same name and are within the merging distance, they will form a single placemark. The descriptions of each report will be displayed and merged by three possible ways.

6.8.11.1 Plain text

Description are inserted without any HTML formatting. Only special HTML entities such as ampersands are reformatted. This is especially useful if the KML document is later converted to GPX, because many GPS devices are not able to display HTML data.

6.8.11.2 HTML tables

Each description of placemark is transformed into a HTML table labelled with the time stamp of the insertion. Here is an example of two Navtex messages from the same station at different times:

2013-02-14 23:18	
Callsign	OST
Country	Belgium
Locator	JO11JE
Message number	35
Frequency	0
Mode	TOR
Message	191533 UTC NOV ; WZ 1196 SELF CANCELING. CANCEL WZ 1192 (GA92) (MA33). WALKER LIGHTBUOY NORMAL CONDITIONS RESTORED."
2013-02-14 23:13	
Callsign	OST
Country	Belgium
Locator	JO11JE
Message number	35
Frequency	0
Mode	TOR
Message	... etc ...

6.8.11.3 Distinct HTML matrix

For the same KML placemark, the key will typically the same for all reports. More, some data are numeric. This is therefore convenient to group them in matrices:

Here is an example for SYNOP weather data, made of three reports:

	2012-12-16 00:00	2012-12-17 06:00	2012-12-18 00:00
Dewpoint temperature		Undefined	Undefined
Figure		11	
Humidity	Unspecified		
Precipitations		Omitted, no observation	Omitted, no observation
Pressure change	Not specified	Not specified	Not specified
Sea level pressure	994 hPa	1000 hPa	1013 hPa
Ship average speed	0 knots	0 knots	
Ship direction	Calm	Calm	
Station type		Automated station. No observation (No 7WW)	Automated station. No observation (No 7WW)
Temperature	9.5 deg C	9.3 deg C	10.3 deg C
Value		37	
Visibility		4 km	4 km
Wave height	3.6 meters	4.7 meters	
Waves height	3.5 meters	4.5 meters	
Waves period	8 seconds	8 seconds	
Wind direction		265 degrees	275 degrees
Wind speed		33 knots (Estimated)	15 knots (Estimated)

6.8.12 Data Retention Time

Data may be automatically purged based on their time-stamp and a maximum retention time in hours. If the retention time is zero, then data are kept for ever.

6.8.13 Command run on KML creation

This command is executed at regular times, by default 180 seconds, and only if new data was written to any KML files. The first time this command is run, its process id is stored. Next time this command must be run, we check if this process is still running. If yes, no new process is created.

The intention is to handle the same way, programs which should always be running, for example KML visualizers, and on the other hand, one-shot scripts or converters. Typical situations are:

- Starting a program such as `Google Earth` or `Marble`, only once per session.
- They will be automatically restarted if they crash, because their `process identifier` is not present anymore.
- Run as needed conversion programs such as GpsBabel, to another format (`GPX`). Or a `FTP` transfer to a remote platform, for inclusion of KML files in Google Maps.
- Accordingly, do not restart this conversion process as long it is not finished (FTP transfers might take long)

6.8.13.1 Example of commands

6.8.13.1.1 FTP Transfer

A new transfer - and a new process - must be initiated at each KML file save. A script is created for this purpose, and the command can be:

```
fldigi/scripts/ftp_kml_files.sh ftpperso.free.fr MyFtpUserName MyPassword
kml
```

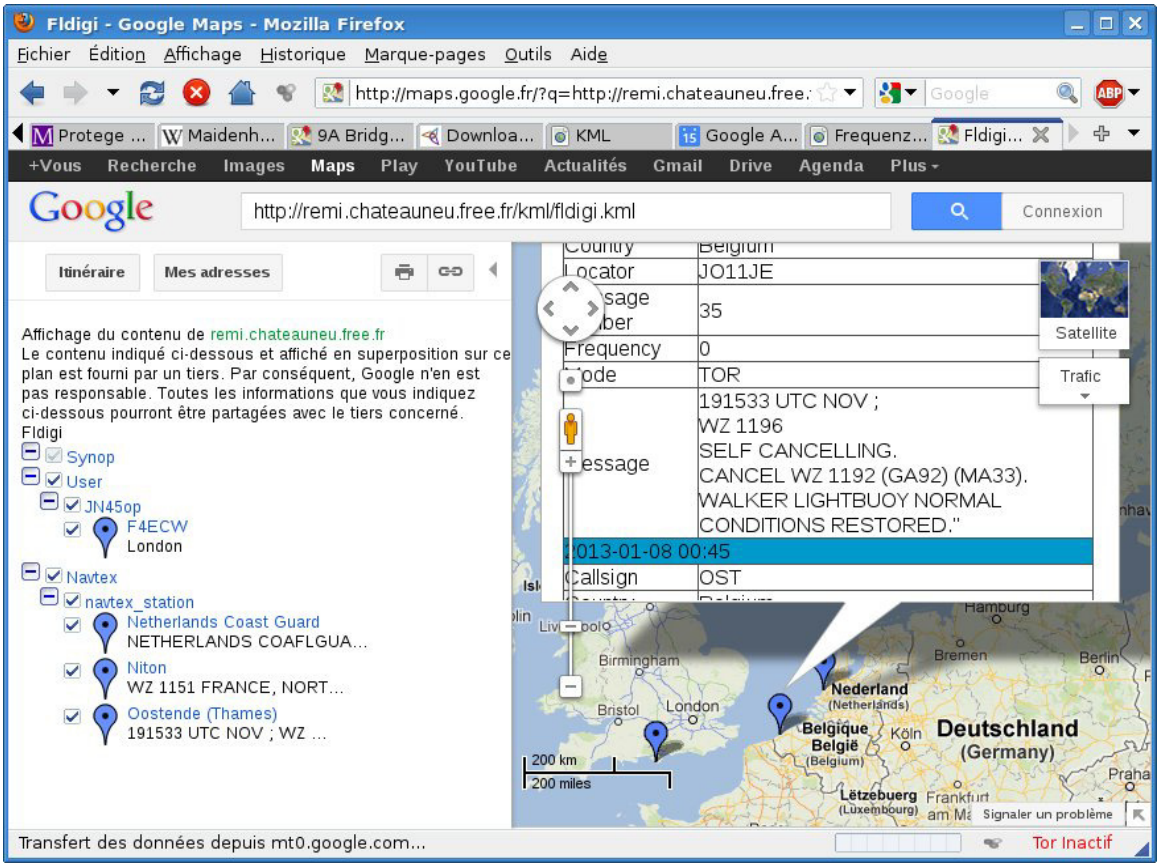


Figure 6.23: KML files displayed in Google Maps

An obvious use is to save these file to a remote machine where they can be accessed with a public URL. This URL can then be given as CGI parameter to `Google Maps` which will display the placemarks on a map. There are `limitations` on the maximum size of KML files which have to be smaller than 10 megabytes.

Note that KML files are for the moment not compressed into KMZ files.

An FTP copy is not necessary if the destination directory for KML files storage is public (That is, accessible from the Internet).

6.8.13.1.2 Launch google-earth

The program will only be launched once, because its process id is still present. The command can be:

```
googleearth $HOME/.fldigi/kml/fldigi.kml
```

It is possible to change the icons by customizing the file `styles.kml`.

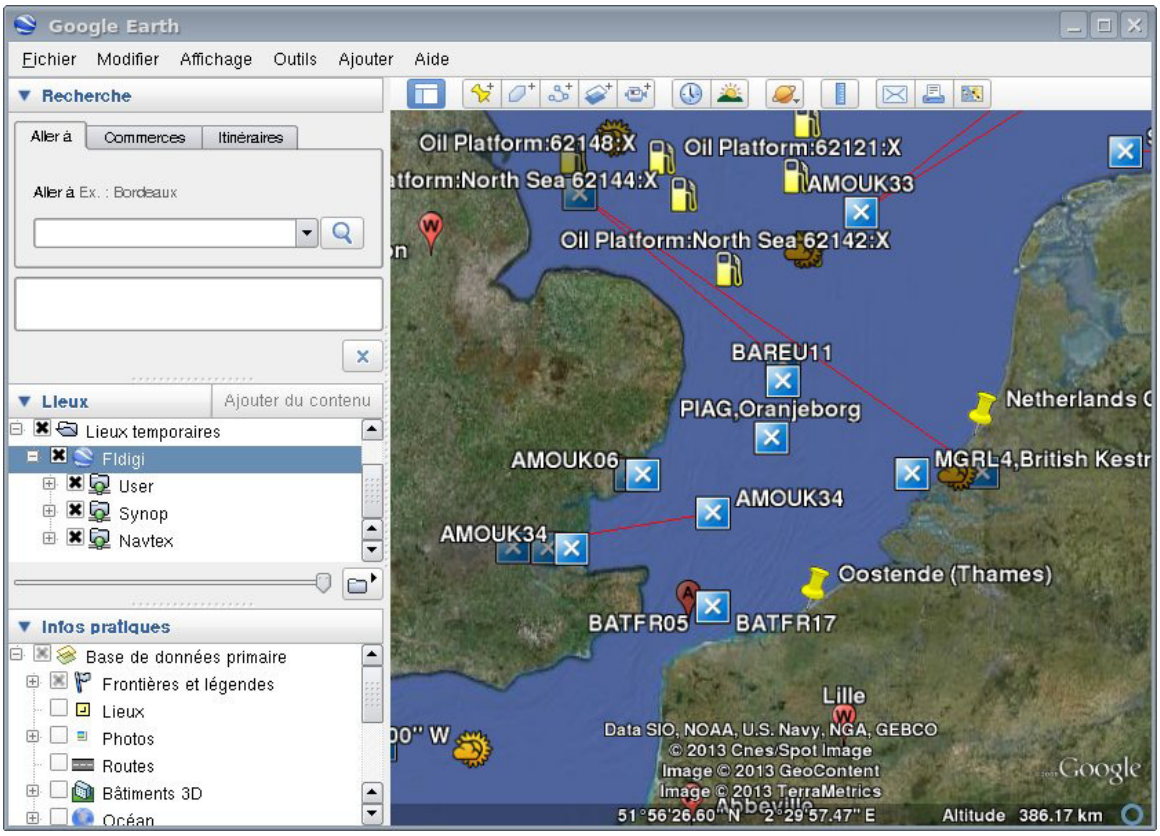


Figure 6.24: Google Earth

6.8.13.1.3 GPS Babel conversion

The command `GpsBabel`, for example, will selectively convert the KML file of Synop reports. It is generally advised to generate plain text description tags in the KML files, because GPS devices might not be able to correctly display HTML data. The command can be:

```
gpsbabel -i kml -f $HOME/.fldigi/kml/Synop.kml -o gpx -F out.gpx
```

The generated files can for example be fed into `Xastir`.

Return to Top of Page

Return to Main Page

6.9 Menus

Figure 6.25: Menu

6.9.1 Menu Heirarchy

6.9.1.1 Files

6.9.1.1.1 Folders

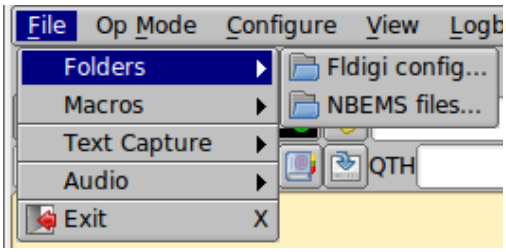


Figure 6.26: Folders

- Fldigi config... - open the OS native file explorer to the folder containing the fldigi operating & data files.
- NBEMS files... - open the OS native file explorer to the folder containing the NBEMS data files

6.9.1.1.2 Macros

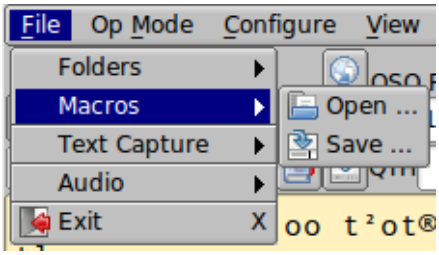


Figure 6.27: Macros

- Open Macros - open a macro definition file ... changes the MACRO keys immediately
- Save Macros - save the current macro definitions to a designated file

Additional information:

Macros

User Interface Configuration - Macros

6.9.1.1.3 Text Capture

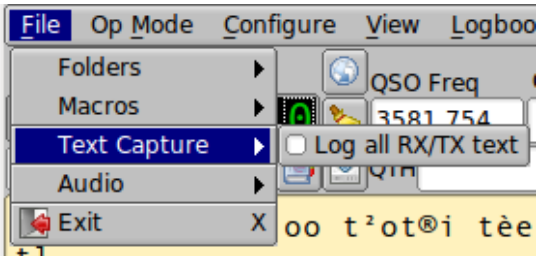


Figure 6.28: Text Capture

Capture all received and transmitted text to a date-time stamped file. Each day's data will be stored in a single file in which the text is appended: fldigiYYYYMMDD.log. This file will be in the "Fldigi config..." folder which can be accessed per the File/Folders menu item. A brief example log of Rx data:

```
--- Logging started at Wed Jan  5 18:42:51 2011 UTC ---
RX 14071955 : PSK31 (2011-01-05 18:42Z): d dx sk S
RX 14071756 : PSK31 (2011-01-05 18:42Z): PSE -lr dACQ CQ de WX1GRS WX1GRS
RX 14071756 : PSK31 (2011-01-05 18:42Z): CQ CQ de WX1GRS WX1GRS
RX 14071756 : PSK31 (2011-01-05 18:42Z): CQ CQ de WX1GRS WX1GRS
RX 14071756 : PSK31 (2011-01-05 18:42Z): PSE K aie =
--- Logging stopped at Wed Jan  5 18:43:04 2011 UTC ---

Each line contains the program state, RX or TX, the frequency, the mode,
the date-time, and the data stream.
```

6.9.1.1.4 Audio

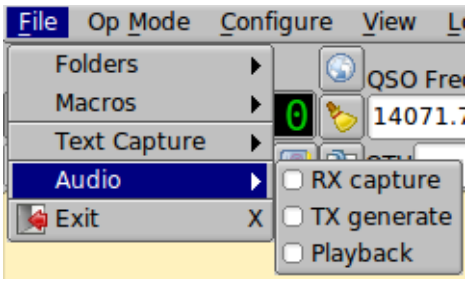


Figure 6.29: Audio

- Rx Capture - allows capturing the incoming audio to a wav file
- Tx Generate - allows capturing the generated tx audio to a wav file
- Playback - playback a previously captured or generated wav file

6.9.1.1.5 Exit

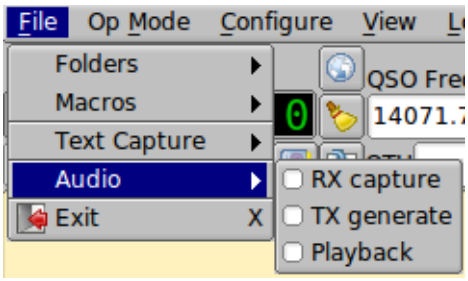


Figure 6.30: Exit

Exit - exit the program closing down the various interfaces in a nice controlled manner.

6.9.1.2 Op Mode

Op Mode - the current operating mode will show as a highlighted menu item.

Figure 6.31: Op Mode

6.9.1.2.1 CW

CW - receive CW 5 to 200 WPM and transmit on any audio frequency using AF CW.

Additional information:

CW

CW Configuration

6.9.1.2.2 Contestia

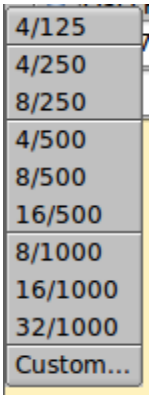


Figure 6.32: Contestia

- Contestia 4 tones, 125 Hz wide
- Contestia 4 tones, 250 Hz wide
- Contestia 8 tones, 250 Hz wide
- Contestia 4 tones, 500 Hz wide
- Contestia 8 tones, 500 Hz wide

- Contestia 16 tones, 500 Hz wide
- Contestia 8 tones, 1000 Hz wide
- Contestia 16 tones, 1000 Hz wide
- Contestia 32 tones, 1000 Hz wide
- Custom ... set on dialog

Additional information:

Contestia

Contestia Configuration

6.9.1.2.3 DominoEX

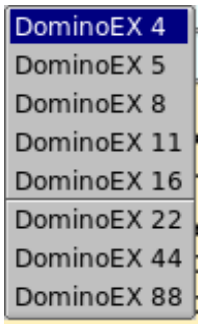


Figure 6.33: DominoEX

- dominoex 4
- dominoex 5
- dominoex 8
- dominoex 11 - the default calling mode for dominoEX
- dominoex 16
- dominoex 22
- dominoex 44
- dominoex 88

Additional information:

DominoEX

DominoEX Configuration

6.9.1.2.4 Hell

Figure 6.34: Hell

- Feld-Hell
- Slow-Hell
- Feld-Hell X5
- Feld-Hell X9
- FSK-Hell (also called FM-Hell by some programs)
- FSK-Hell105
- Hell-80

Additional information:

Hellschreiber

Feld Hell Configuration

6.9.1.2.5 MSFK

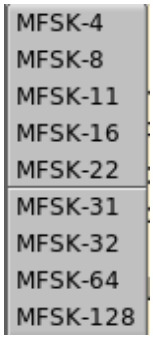


Figure 6.35: MSFK

- MSFK 4
- MSFK 8
- MSFK 11
- MSFK 16
- MSFK 22
- MSFK 31

- MSFK 32
- MSFK 64
- MSFK 64L
- MSFK 128
- MSFK 128L

Additional information:

MFSK

6.9.1.2.6 MT-63

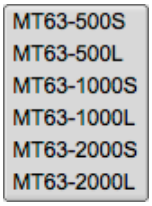


Figure 6.36: MT63

- MT63-500S
- MT63-500L
- MT63-1000S
- MT63-1000S
- MT63-2000S
- MT63-2000L

Additional information:

MT63

MT63 Configuration

6.9.1.2.7 PSK

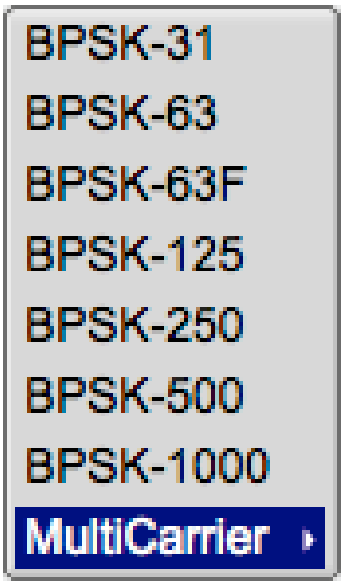


Figure 6.37: PSK

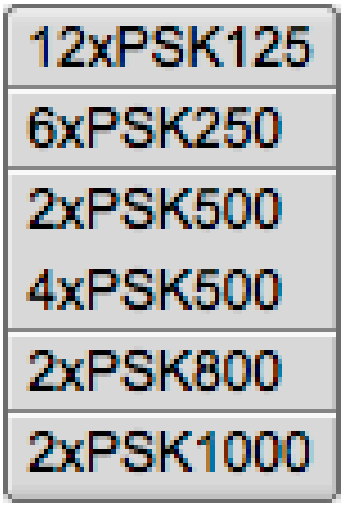


Figure 6.38: Multi Carrier PSK

- psk 31 - phase shift keying - 31.625 baud
- psk 63 - phase shift keying - 63.25 baud
- psk 63F - phase shift keying with FEC - 63.25 baud
- psk 125 - phase shift keying - 126.5 baud
- psk 250 - phase shift keying - 253 baud
- psk 500 - phase shift keying - 506 baud
- psk1000 - phase shift keying - 1012 baud
- MultiCarrier
  - 12 x psk 125 - 12 carrier psk125 - 126.5 baud - bandwidth 2000 Hz

- 6 x psk 250 - 6 carrier psk 250 - 253 baud - bandwidth 2000 Hz
- 2 x psk 500 - 2 carrier psk 500 - 506 baud - bandwidth 1200 Hz
- 4 x psk 500 - 4 carrier psk 500 - 506 baud - bandwidth 2600 Hz
- 2 x psk 800 - 2 carrier psk 800 - ~810 baud - bandwidth 2300 Hz
- 2 x psk 1000 - 2 carrier psk 1000 - 1012 baud - bandwidth 3200 Hz

Additional information:

PSK - BPSK, BPSKR, QPSK, and Multi-Channel Modems

6.9.1.2.8 QPSK

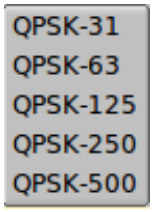


Figure 6.39: QPSK

- qpsk 31 - quadrature phase shift keying - 31.25 baud
- qpsk 63 - quadrature phase shift keying - 63.25 baud
- qpsk 125 - quadrature phase shift keying - 126.5 baud
- qpsk 250 - quadrature phase shift keying - 253 baud
- qpsk 500 - quadrature phase shift keying - 506 baud

Additional information:

PSK - BPSK, BPSKR, QPSK, and Multi-Channel Modems

6.9.1.2.9 PSKR

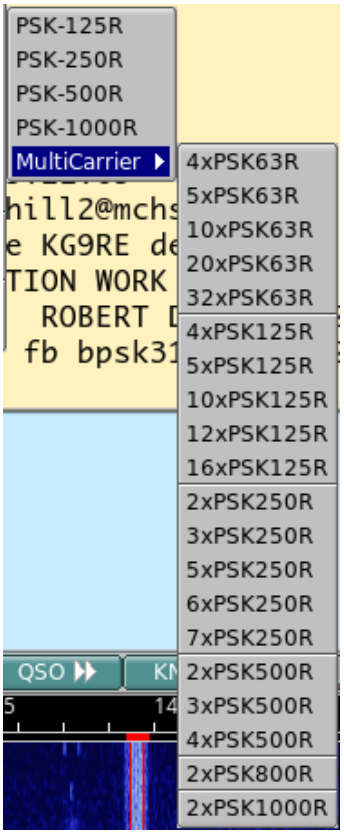


Figure 6.40: PSKR

All PSKR modes are phase shift keying, with FEC and interleaving.

- psk 125R - 126.5 baud
- psk 250R - 253 baud
- psk 500R - 506 baud
- psk 1000R - 880 baud
- MultiCarrier
  - 4 x psk 63R - 4 carrier, bandwidth 330 Hz
  - 5 x psk 63R - 5 carrier, bandwidth 416 Hz
  - 10 x psk 63R - 10 carrier, bandwidth 550 Hz
  - 20 x psk 63R - 20 carrier, bandwidth 1725 Hz
  - 32 x psk 63R - 32 carrier, bandwidth 2775 Hz
  - 4 x psk 125R - 4 carrier, bandwidth 352 Hz
  - 5 x psk 125R - 5 carrier, bandwidth 440 Hz
  - 10 x psk 125R - 10 carrier, bandwidth 1100 Hz
  - 16 x psk 125R - 16 carrier, bandwidth 1760 Hz
  - 2 x psk 250R - 2 carrier, bandwidth 440 Hz
  - 3 x psk 250R - 3 carrier, bandwidth 660 Hz
  - 5 x psk 250R - 5 carrier, bandwidth 1100 Hz

- 6 x psk 250R - 6 carrier, bandwidth 1320 Hz
- 7 x psk 250R - 7 carrier, bandwidth 1540 Hz
- 2 x psk 500R - 2 carrier, bandwidth 880 Hz
- 3 x psk 500R - 3 carrier, bandwidth 1320 Hz
- 4 x psk 500R - 4 carrier, bandwidth 1760 Hz
- 2 x psk 800R - 2 carrier, baud 800, bandwidth 1280 Hz
- 2x psk 1000R - 2 carrier baud 1012, bandwidth 1760 Hz

Additional information:

PSK - BPSK, BPSKR, QPSK, and Multi-Channel Modems

6.9.1.2.10 Olivia

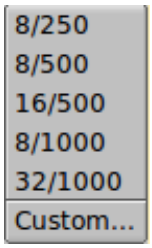


Figure 6.41: Olivia

- 8/250 8tone, 250 Hz wide signal format
- 8/500 8 tone, 500 Hz wide signal format
- 16/500 16 tone, 500 Hz wide signal format
- 32/1000 32 tone, 1000 Hz wide signal format
- Custom - tones and bandwidth configurable on Olivia tab

Additional information:

Olivia

Olivia Configuration

6.9.1.2.11 RTTY

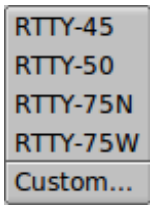


Figure 6.42: RTTY

- RTTY-45 45 Baud Baudot, 170 Hz shift, used primarily in U.S.
- RTTY-50 50 Baud Baudot, 170 Hz shift, used primarily in Europe
- RTTY-75N 75 Baud Baudot, 170 Hz shift

- RTTY-75W 75 Baud Baudot, 800 Hz shift
- Custom - Baud Rate, Baudot/ASCII, Shift etc configurable on RTTY tab

Additional information:

RTTY

RTTY / FSK Configuration

6.9.1.2.12 THOR

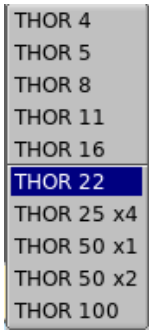


Figure 6.43: THOR

- Thor-4
- Thor-5
- Thor-8
- Thor-16
- Thor-22
- Thor 25 x 4 - 4 x tone spacing, 2 second interleave
- Thor 50 x 1 - 1 second interleave
- Thor 50 x 2 - 2 x tone spacing, 1 second interleave
- Thor 100 - 0.5 second interleave

Additional information:

Thor

Thor Configuration

6.9.1.2.13 THROB

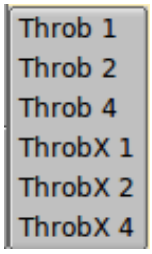


Figure 6.44: throb

- Throb1

- Throb2
- Throb4
- ThrobX-1
- ThrobX-2
- ThrobX-4

Additional information:

Throb

6.9.1.2.14 WEFAX

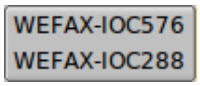


Figure 6.45: WEFAX

- WEFAX-IOC576
- WEFAX-IOC288

Additional information:

WEFAX

6.9.1.2.15 NAVTEX / SITOR

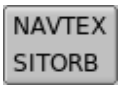


Figure 6.46: NAVTEX / SITOR

- Navtex
- SitorB

Additional information:

NAVTEX and SITOR-B

6.9.1.2.16 SSB, WWV, and Freq Anal

- SSB - fldigi does not transmit, but can be used for rig control, signal frequency analysis and logging
- WWV - special receive only modem used for calibrating sound card
- Freq Anal - used for carrier detection and frequency measurement

Additional information:

Frequency Analyzer

Digiscope Display - WWV mode

6.9.1.3 Configure

Figure 6.47: Configure

See Configuration Page

6.9.1.4 View

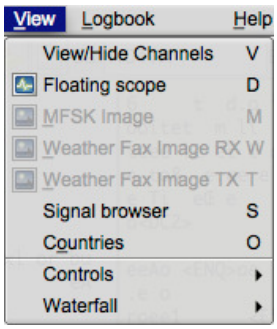


Figure 6.48: View

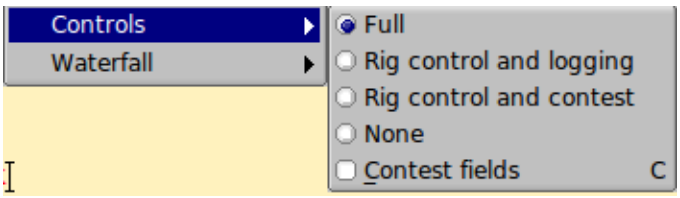


Figure 6.49: View Controls

Figure 6.50: View Waterfall

- View/Hide Channels - Opens / Closes muli-signal viewer, shares space with Rx/Tx panels
- Floating Scope - Opens up a resizable, moveable scope display
- MFSK Image - Opens the MFSK picture image (if being received)
- Weather Fax Image - Opens WEFAX send/receive dialog
- Signal browser - open the psk/rtty viewer dialog to display up to 30 simultaneously decoded psk signals Dialog viewer contains shared data / control with the View/Hide Channel viewer
- Countries - DXCC entities
- Controls
  - Full - show all logbook and rig controls
  - Rig control and logging - minimizes the logging panel for normal QSO entries
  - Rig control and contest - minimizes the logging panel for Contest entries
  - None - removes the Rig Control / Log panel from the main User Interface
  - Contest fields - Display alternate 2nd line in qso logging area; provides access to contest logging fields
- Waterfall
  - Docked scope - toggles the visibility of the docked scope display to the right of the waterfall
  - Min WF Controls - toggles the visibility of various waterfall controls as configured by the user

6.9.1.5 Logbook

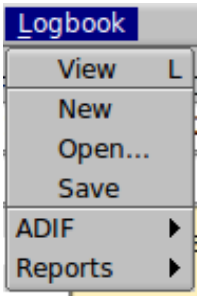


Figure 6.51: Logbook

Figure 6.52: Logbook ADIF

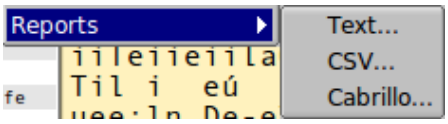


Figure 6.53: Logbook Reports

- View - open the logbook dialog
- New - create a new logbook
- Open - open an existing logbook
- Save - save the current logbook
- ADIF / Merge - merge adif data from an ADIF file
- ADIF / Export - export selected or all logbook records to an ADIF formated file - see Log Exports.
- Reports / Text ... - export selected or all logbook records to a text file suitable for printing
- Reports / CSV ... - export selected or all logbook records to a tab delimited file
- Reports / Cabrillo ... - create a Cabrillo contest report.
  - see Cabrillo Reports.

6.9.1.6 Help

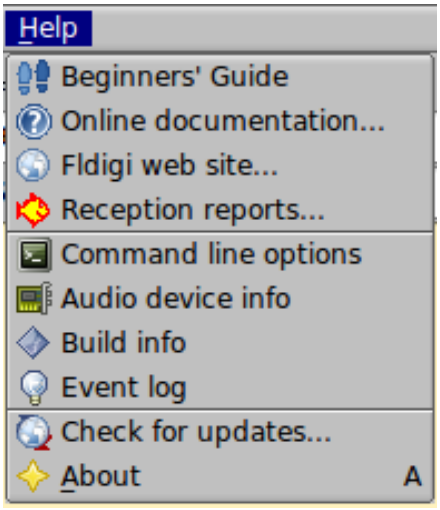


Figure 6.54: Help

- Beginners' Guide
- On line documentation... - open up default browser to the on-line Help site
- Fldigi web site... - open up default browser to the www.w1hkj.com primary web page
- Reception reports... - open up browser to the `http://pskreporter.info` web page preset to your callsign
- Command line options - display a list of all command line switches available to the fldigi user
- Audio device info - displays information about all audio devices detected on the computer system

- Build info - displays all relevant information regarding the compilation and link for the application
  - build info
- Event log - opens a text display window that records various events depending on the level of reporting depth selected. This is a useful window for reporting problems with the program to the developers.
- Check for updates... fldigi silently opens a download web site, checks and reports on whether a new version is available.
- About - Version number and a little about the programmers

6.9.1.7 Spot, Tx RSID, Rx RSID, and Tune Buttons

Figure 6.55: Menu

- Spot button - The "Spot" light button is visible if callsign spotting is enabled. Use this button to toggle the callsign spotting reporter on and off. It is automatically turned off when playback is selected in the Files menu. The main window text is not searched if the viewer is active, i.e., if it is displayed and the current modem is PSK. See `PskReporter` and `Notifier`.
- RxID button - toggles the detection of Reed Solomon Identification codes.
- TxID button - toggles the transmission of the RSID signal.
- Tune button - toggles the "Tune" mode which causes fldigi to insert a tone at the current waterfall frequency. The peak-to-peak amplitude of this tone is the standard by which you should set your transmitter drive or adjust your antenna matching network.
- The right most button is normally not visible. This is the count-down timer button that is enabled when a macro button has been configured to repeat after a specified number of seconds. This button shows the count-down to the next transmission. Pressing the button disables the count and restores fldigi to normal keyboard operation.

Return to Top of Page

Return to Main Page

6.10 Mouse and Keyboard Shortcuts

Fldigi has a bewildering number of keyboard and mouse shortcuts, some of which may help make your particular style of operation more efficient. You do not need to know them all to make effective use of the program!

6.10.1 Main window

6.10.1.1 Text input fields

Most text fields use a combination of CUA (PC) and Unix-style keybindings. Text can be marked, copied, pasted, saved to a file as well as transfer to other main panel controls. A right click on any text control will open a context sensitive menu for that particular control. A full list can be found on the `FLTK web site`

The received/transmitted text widgets use CUA key bindings with some modifications:

6.10.1.1.1 RX text

This widget is read-only and ignores shortcuts that would modify its contents. See logbook for details on the Rx right click popup menu system.

6.10.1.1.2 TX text

The text that has already been sent is protected, but can be deleted one character at a time with the Backspace key. Right clicking on the Tx text panel opens the following popup menu:

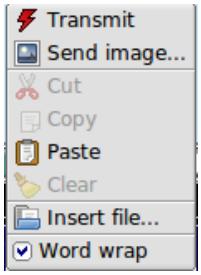


Figure 6.56: Fldigi Receiving Editing Tx text

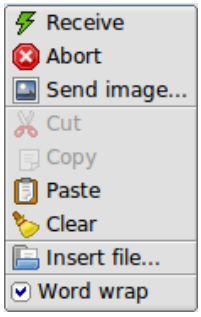


Figure 6.57: Fldigi Transmitting

- Transmit

  put the program into the transmit operation
- Receive

  during a transmit or tune, end the transmit and restore receive operation
- Abort

  during a transmit, receive without waiting for the modem to finish sending
- Send image

  for MFSK only, send an image using MFSKpic mode
- Clear

  clear all of the text
- Cut

  delete the marked text (by left click drag over text)
- Copy

  copy the marked text to the clipboard
- Paste

  the clipboard text to the current text insertion point

- Insert file

  select a file from file browser to insert in text at insertion point

- Word wrap

  turn word wrap on/off

The Tx panel is fully drag and drop aware. That means you can add a file to the transmit text by simply opening up a file manager (different for different OS and choice of desktop). Select the file from the manager and then drag and drop it onto the Tx panel. The mouse pointer will move the cursor insert point for the drop.

A number of additional shortcuts can be found in the Keyboard Operation section.

6.10.1.2 Waterfall display

Most of fldigi's unusual shortcuts are specific to this widget.

6.10.1.2.1 Waterfall display - Keyboard

- Shift Left/Right - move the b/w marker by 1 Hz
- Ctrl Left/Right - move the b/w marker by 10 Hz

6.10.1.2.2 Waterfall display - Mouse

- Left click/drag - move the b/w marker to, and start decoding at the cursor frequency
- Right click/drag - as above, but return to previous position on release
- Middle click - toggle AFC
- Ctrl-Left click - replay audio history at b/w marker position
- Ctrl-Right click - replay at cursor frequency and return on button release
- Ctrl-Middle click copy the frequency under the cursor to the currently selected (or first) channel in the PSK viewer, and select the next channel
- Shift-Left click/drag same as unmodified left click; no signal search
- Shift-Right click/drag likewise, with a return to the previous frequency when the button is released, no signal search
- Shift-mouse wheel - move the squelch slider
- Scroll wheel usage is dependent upon the configuration (see ConfigWaterfall)
  - None - no mouse wheel activity in waterfall panel
  - AFC range or BW - adjust the AFC range/BW up/down
  - Squelch level - adjust the squelch level up/down
  - Modem carrier - adjust the audio tracking point +/- Hz increments
  - Modem - select modem type from a full rotary of available modems
  - Scroll - move the waterfall left/right in 100 Hz increments (for 2x, 4x expanded waterfall view)
  - Ctrl-mouse wheel - change the AFC search width in PSK modes, or the bandwidth in CW and FeldHell
- Ctrl-Alt-Left click on the mouse when the mouse pointer is in the waterfall - if operating with flrig and a supported transceiver such as the FT-950, will set the transceiver manual notch frequency to the frequency under the mouse cursor. Repeat the Ctrl-Alt-Left click to disengage the notch. Disengagement is not dependent on where the cursor is on the waterfall, just that it be in the waterfall.

6.10.1.2.3 Waterfall "Store" button

- Left click - Add a new item for the current frequency and modem
- Shift-Left click - Delete all items
- Middle click - Select last item in menu
- Right click - Pop up menu
  - Left/right click - Select item (and switch to that frequency/modem)
  - Shift-Left/right click Delete item
  - Middle click - Update (replace) item

6.10.1.2.4 Digiscope display

- Mouse wheel - Change AFC/BW, same as Ctrl-mouse wheel on the waterfall

6.10.1.2.5 Rig control window

There are some shortcuts in addition to those described in the Rig Control

6.10.1.2.6 Frequency display

- Left/Right arrow key - change the frequency by one 1 Hz
- Up/Down arrow key - change the frequency by 10 Hz

6.10.1.2.7 Frequency list

- Shift-Left click - delete the line under the cursor
- Middle click - replace the line under the cursor with the current frequency/mode/modem

6.10.1.2.8 PSK viewer window

- Besides the bindings mentioned in the PSK Viewer section, there are mouse shortcuts to change the nominal frequency of a viewer channel:
  - Middle click copy the current waterfall b/w marker frequency to the channel under the cursor, overwriting that channel's nominal frequency
  - Right click - restore a channel's nominal frequency
  - Right click on Clear - as above, for all channels

Return to Top of Page

Return to Main Page

6.11 Notifier

This dialog available is used to specify search patterns and alerts that are triggered when the decoded Rx text matches those patterns. This only happens when the Spot button on the main window is activated, as with the PSK Reporter client.

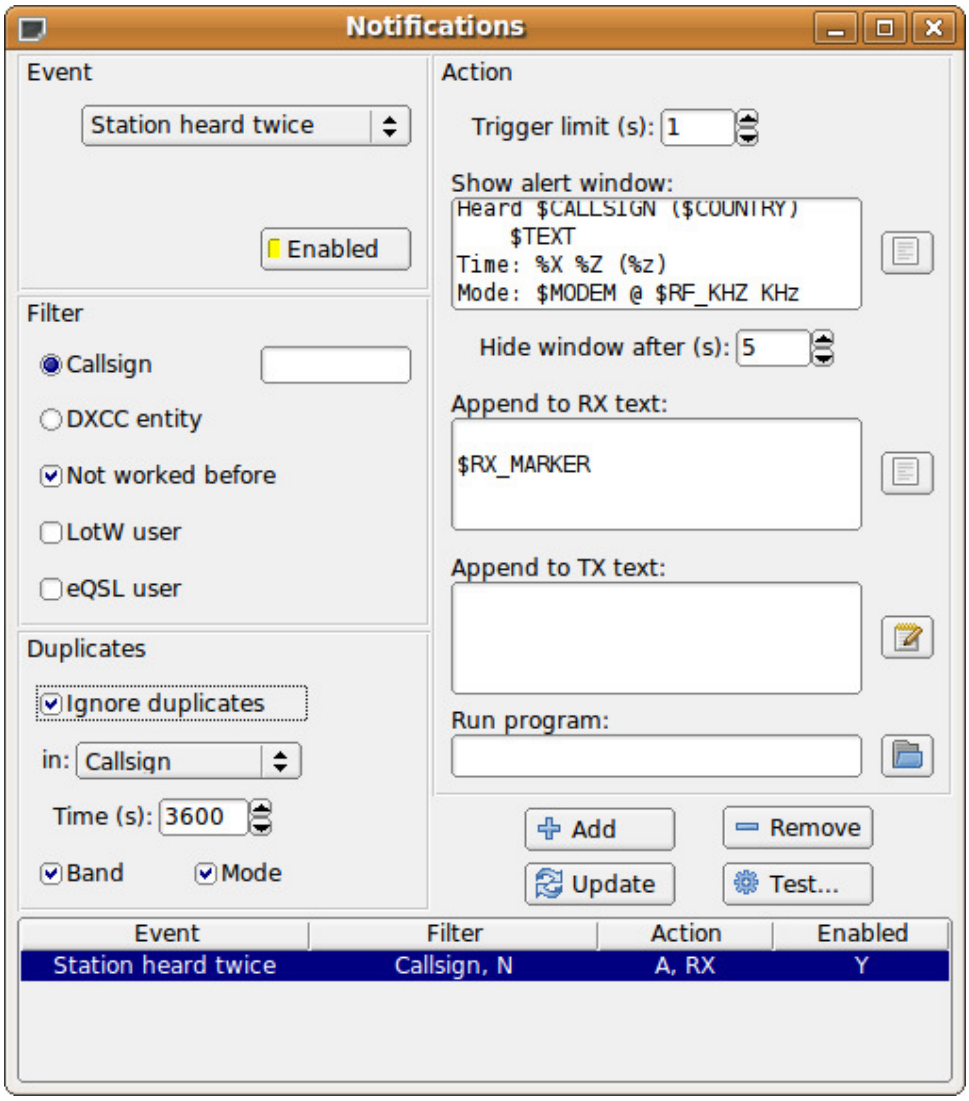


Figure 6.58: Notifications

First, here's how it works in general. You specify a regular expression (RE) that contains one or more parenthesised capturing groups. Fldigi's spotter matches it against the incoming text (main window or Signal Browser, so it works with both PSK and RTTY) and, if the RE matches, it performs one or more of the following:

Displays an alert window with some text and a "go to that frequency" button.

Enters some arbitrary text into the Transmit pane. The text may contain <MACRO>s and these will be expanded as usual.

Runs a program (Unix/Linux only for now).

The text described by the capturing group(s) can be used in all of the above. There is an example of this at the end of this page.

Not everyone is at ease in writing regular expressions for the notifier to act upon. So a few "canned" searches are coded into the notifier and are selected from the event chooser at the upper left of the dialog.

1. My Callsign de CALL. Can be used to alert you when CALL calls you.
2. Station heard twice. Pretty much the same search that the PSK reporter client does.
3. Custom text search. This reveals an input field where you type your own RE.

Both (1) and (2) are special cases of (3), but with some extra processing available because in each case fldigi knows what it has just found.

The Filter pane is available for the first two event types only, i.e. not the custom text search. In this pane you can specify some properties that the spotted callsign must have for the actions to take place:

a) The Callsign radio button reveals a text field when selected. If you enter something in that field, the event will be accepted only if the text matches the spotted callsign (I may change this to a RE match).

b) The "DXCC entity" radio button reveals a button that brings up a list of DXCC entities. Select entities by clicking or dragging. If you select any at all, the spotted callsign's country will have to be one of those or the event will be ignored. Having no entities selected is the same as selecting all of them, i.e. any country, but is a more efficient.

The entity list can be sorted by clicking on the row headers, and there is a right click context menu that can (de)select by continent and CQ zone. The buttons and search fields at the bottom behave as you'd expect.

The list is also available with the menu item "View / Countries" in the main window.

You need cty.dat for all this to work

c) The "Not worked before" check button asserts that, if you have selected (a) above, the callsign must not be in your logbook. Same with (b), but now you must not have had any QSOs with stations from that country in the log.

d) The "LotW user" and "eQSL user" buttons specify that the callsign must be on one of these two lists (the documentation explains where to get the user lists from and where to put the files).

The Action pane is where you choose how fldigi will alert you when an event matches the filter bits.

a) The text in the "Show alert text" box, if not empty, is shown in a pop-up window. The alert window has a timer and dismisses itself after a configurable time interval (the "Hide after" control). The user can click anywhere inside the window to stop the timer.

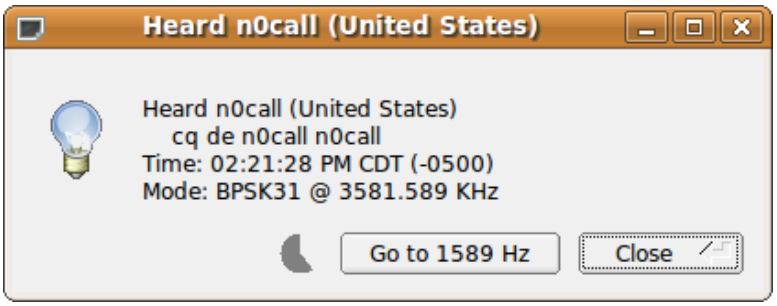


Figure 6.59: Notifier Popup

The button next to the text box enters the default alert text for the event you have selected. There are a few variables that are substituted when the window is displayed:

For all three event types: $MODEM (modem name), $DF_HZ (dial frequency), $RF_HZ (actual receive frequency), $RF_KHZ, $AF_HZ (modem audio frequency)

For the 1st event type (my call): $CALLSIGN, $TEXT (all matched text).

For the 2nd event type (station): $CALLSIGN, $TEXT, $COUNTRY.

For the 3rd event type (custom): you're on your own here, but fldigi will helpfully list all the possible substrings found in your RE.

The whole text is passed through strftime(3) so you can customize the date. Here's a reference for the % characters:

http://www.opengroup.org/onlinepubs/007908799/xsh/strftime.html

b) The "Append to TX text" box – self explanatory. The same variable substitutions apply, as well as macro expansion. The nearby button shows the macro editor. The appended rx text is clickable. Clicking it will move the waterfall frequency (and transceiver if under CAT) to the detected signal and change to the indicated mode.

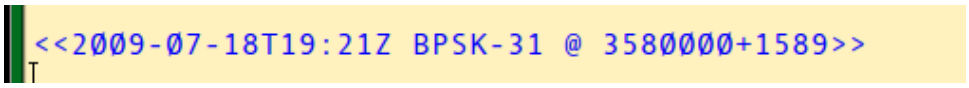


Figure 6.60: Notifier RX Text

c) The "Run program" field and browse button are only available on Unix systems. Field contents are passed to the shell ("/bin/sh -c"), as with system(3). No variable or \backref substitution is done for this field, but all substrings are exported as environment variables, such as FLDIGI_NOTIFY_STR_1. The usual <EXEC> macro variables are also there and your ~/.fldigi/scripts directory will be in the shell's path. Try it out with a test script for the full list of variables.

d) The trigger limit box specifies how much time must pass between subsequent invocations of whatever actions you have specified.

The Duplicates pane has a check button that displays the rest of that group when checked. If you enable this, fldigi will remember what it has seen and ignore the event if it is a duplicate. The other controls in that pane determine what constitutes a duplicate:

a) The menu tells fldigi what to look at. For the first two event types, the menu will display "Callsign", and for the custom search it will contain a list of \X references for the RE.

b) The time box is also essential; it determines how close the events must be in time to be considered duplicates.

c) The Band and Mode check boxes further restrict the comparison.

An example:

You are looking at callsigns, with a dup time of 600s, and both Band and Mode checked. A callsign is found once and fldigi alerts you. Now if this callsign is spotted again, less than 600s later and in the same band and mode, it is a duplicate and will be ignored. With (say) Band and Mode unchecked, it is a duplicate regardless of frequency band or mode as long as it's heard before the 600s elapse.

Three of the four buttons at the bottom left are pretty much self-explanatory. Add to the list an event you have just specified, or select an event from the list and Remove it, or change some of its parameters and Update it.

The Test... button allows you to test an event with some text of your choice. This is particularly useful with the custom text search, as it's too easy to enter a RE that will never match. The dialog will show you the default test string for the two fixed event types. Careful: the "Station heard twice" event type expects a non-alphanumeric character at the end of its input. The default test string has a space at the end.

If nothing happens, it may be because you have not specified any actions, or because the event's filter does not match, or because the trigger limit or dup handling are preventing the actions from happening. In the latter case, updating an event will reset its dup data. But it's better to add the dup and trigger limits at the end, after you've tested the event.

The list at the bottom of the window shows the events you have added. All contents are saved in the file ~/.fldigi/notify.prefs.

The list has a context menu for quick access to Update, Remove, and Toggle. The first two have the same effect as clicking on the button of the same name.

The Toggle item lets you flip the "Enabled" status of an event: this is like selecting an event, clicking on the "Enabled" button in the Event pane to (de)activate it, and then clicking "Update". Disabled events are kept on the list but are not registered with the spotter and so they are never triggered.

If you disable all the events and there is nothing else using the spotter (e.g. PSK Reporter), the Spot button will disappear from the main window.

A 2nd example:

Here's how to do the "my call" event using the custom text search:

a) Select "Custom text search" in the event pane

b) In the RE box, enter (without the quotes or leading white space):

"<YOUR_CALL>.+de[[:space:]]+([[:alnum:]]?[[:alpha:]/]+[[:digit:]]+[[:alnum:]/]+)"

and remember to replace <YOUR_CALL> with your callsign.

c) In the actions pane you can now use \0 for the whole text matched by the above RE, and \1 for the first capturing group (the callsign).

d) Select "\1" in the duplicates menu if you want dup filtering.

e) Test with "<YOUR_CALL> de <SOME_OTHER_CALL>" and you should see the alert window with the text you specified.

Addional examples:

Add a "My callsign de CALL" event with a script that will do something to get your attention when someone calls you.

Add a "Station heard twice" with the DXCC filter and the "Not worked before" option. Also set the LotW or eQSL options if desired.

Add a "Station heard twice" with no callsign/dxcc/etc. filter but with duplicate filtering. Write a script that sends the data to a DX cluster or similar.

Here is a simple Perl script that uses notify-send (in the package libnotify-bin on Debian) to display desktop notification "bubbles". A better version would use the libnotify bindings for Perl or Python directly.

```
-----------------------------snip
#!/usr/bin/perl
exec("notify-send", "-t", "5000", "-i", "/usr/share/pixmaps/fldigi.xpm",
     "Heard " . $ENV{"FLDIGI_NOTIFY_CALLSIGN"} . " ($ENV{FLDIGI_NOTIFY_COUNTRY})",
     $ENV{"FLDIGI_NOTIFY_STR_0"});
snip-----------------------------
```

Return to Top of Page

Return to Main Page

6.12 Operating Controls and Displays

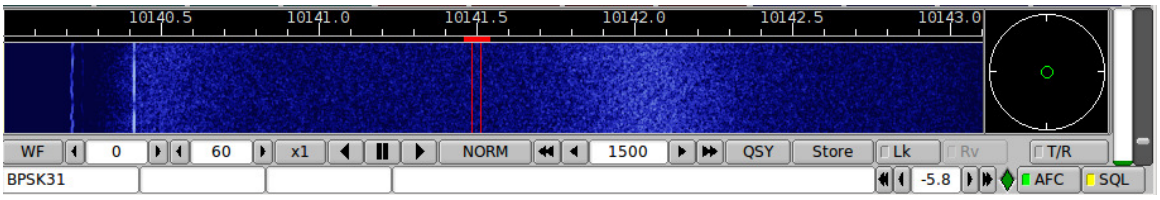


Figure 6.61: FLDIGI Controls

The main display for fldigi is the waterfall display shown above in color and in scale x1.

The button **WF** toggles the display between a waterfall, a spectrum display and an oscilloscope type view of the Rx and Tx signals. This button acts as a rotary. Left clicking moves the display selection in one direction and right clicking in the other direction. The three display modes are **WF** - waterfall, **FFT** - spectrum (Fast Fourier Transform) and **Sig** - oscilloscope time domain. Let the mouse cursor hover over any one of the controls and a small hint box will open to help you navigate the various controls.

The **Norm** button controls the speed of the waterfall drop. This is also a rotary type of button control. The speeds available are SLOW, NORM, FAST and PAUSE. The load on the cpu will be directly proportional to this selection. If your cpu is slow you might want to select the SLOW or PAUSE option for the waterfall.

The scale control (X1, X2, X4) expands or contracts the view into the fast fourier transform that is displayed on the waterfall or the FFT display. fldigi always computes the FFT to a 1 Hz resolution, and displays the results according to the scale control.

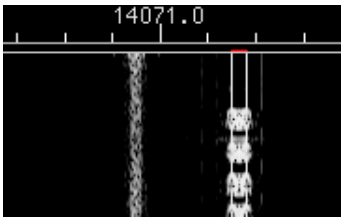


Figure 6.62: PSK Waterfall X1 scale

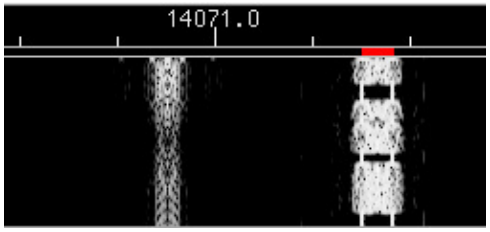


Figure 6.63: PSK Waterfall X2 scale

Figure 6.64: PSK Waterfall X4 scale

The next three controls are positional conrols for the waterfall. The waterfall can display 4096 data points, where each one can be thought of as a spectral line at the equivalent Hertz. The ratio is actually 8000/8192 and is related to the ratio of sound card sampling rate to Fast Fourier Transform length. This ratio changes for some modems that require a sampling rate other than 8000 Hz. The left arrow key will shift the display to the right (displays a lower section of the spectrum). The right arrow key moves the display higher in frequency. These two buttons are repeating buttons. Hold them down and the display slews at about 20 shifts / sec. The center button with the two vertical block lines is a "center the signal" button. The current cursor (red signal cursor in the waterfall) will be centered in the display area.

NOTE: these controls are only functional if the current waterfall or spectrum view is smaller than the full view available. This is usually the case when the X2 or X4 expansion is selected. But it also might be the case when the width of the main dialog is reduced so that the waterfall display does not extend over the entire available width.

Try moving the cursor around in the waterfall area. You will see a set of yellow cursor blocks that show the center point and bandwidth of the current operating mode (psk31 = 31.25 Hz for example). To capture a received signal just click near the signal and the AFC will perform a multi-step acquisition. This will be very fast and should not require additional operator intervention. *Casual tuning* You can take a look at any received signal on the waterfall by right-clicking and holding the mouse button on or near the signal. The modem will begin to decode that signal if it is in the currently selected mode. The text will be a unique color on the Rx text widget so that you can discern the difference between casual and normal tracking. Release the mouse button and the tracking returns to the previously selected normal tracking point.

Audio History Fldigi maintains a history buffer of the received audio. This buffer is approximately 2 minutes in duration. After tracking commences on a signal you can decode the audio history for that signal. The audio history is invoked by a Ctrl-Left click anywhere on the waterfall. You can also invoke the audio history for the casual tuning mode by pressing Ctrl-Right click on the waterfall.

The next control is your transceive audio frequency. In the display above you can see that the audio signal is 1500 Hz. The red cursor is centered beneath 14071.500 Mhz. The transceiver was set to 14070 Mhz. The arrow key pairs move up/down in cycles and tens of cycles. You can fine tune the receive point using this control.

The next two controls to the right of the audio frequency control are for the receive signal processing. The one that reads -10 is the max signal level for the waterfall/spectrum display. The one that reads 51 is for the range over which that control will display signals. Both of these are in dB. The default of -10 / 40 is a good starting point, but you need to adjust these for band conditions. You can see the impact of these controls most easily by putting the main display area in the spectrum mode. Changes in these controls will effect the waterfall instantly and for all past history displayed on the waterfall. You do not have to wait for new signal data to observe the affect.

The **QSY** button is very specific to rigs interfaced with either hamlib or the memory mapped i/o. Each rig has a sweet spot associated with its bandwidth controller. For the Argonaut V this is 1100 Hz. For the the Kachina it is 1000 Hz. As the transceivers bandwidth is changed the changes occur centered at this frequency. So .... let's say that I just started copying a rare dx at 1758 Hz and I wanted to put the signal at the sweet spot so I could easily narrow the receiver bandwidth. Click on the signal on the waterfall. Let the AFC capture and then press the QSY button. The tranceiver frequency will be shifted and the fldigi audio tracking point shifted in unison such that the signal is now at the receivers sweet spot. Very fast and very convenient! If you do not have hamlib enabled for your transceiver this button will be dimmed and not activated.

The **M**> button allows you to store, recall and manage mode/frequency pairs. If you want to save the current mode and frequency simply left click the button. A right click will enable a popup menu from which you can select a previously stored set. You can quickly move between modes and audio sub carrier using this technique. A shift-left click will clear the memory. When the popup menu is visible you left click on an entry to select it. You can shift-left click on an entry to delete that single entry.

The **T/R** button should be self-explanatory. It's your transmit/receive button. Action is immediate, so if you were transmiting some text and hit the button the PTT is disabled, the transmit text area cleared and the program returned to receive mode. The T/R button is a "lighted button" that shows when transmitting. All other lighted buttons show YELLOW when they are in the active state.

The **Lck** button locks the transmit audio frequency to its present value. You can then continue to QSY around your transmit position. I have used this to reply to a DX station that wanted a +500 Hz response. The DX was at 690 Hz audio, and wanted a response at +500. I moved the display cursor (or the audio frequency control) to 1190 Hz. Hit the Lck button and then went back to 690 with the waterfall cursor. Now the program is receiving on 690 Hz and transmitting on 1190 Hz. Caught him on the first try. Use this button also as a *Master Station* control. Not all rigs are equal in their VFO performance. Some exhibit a shift between receive and transmit. If this occurs then the stations find themselves chasing each other with every t/r exchange. Locking your transmit frequency with this control will inhibit that from happening. Be sure to disable the control when that qso is over or **you may forget and transmit over top of another qso!**

If the "Lck" is enabled the TX frequency does not follow the AFC action applied to the RX frequency.

For transceivers which are either hamlib or memmap enabled, if the "Qsy" button is pressed BOTH the RX and TX frequencies are changed to synchronize to where the RX was positioned.

Perhaps some numbers will help to make that a little clearer.

"Lck"	Before "Qsy"		After "Qsy"	
	RX	TX	RX	TX
OFF	1002 / 7071.002	1002 / 7071.002	1500 / 7071.002	1500 / 7071.002
ON	1002 / 7071.002	1000 / 7071.000	1500 / 7071.002	1500 / 7071.002
ON	1000 / 7071.000	1800 / 7071.800	1500 / 7071.000	1500 / 7071.000

With "Lk" off the TX audio frequency is always synchronized with the RX frequency.

With "Lk" on the TX audio frequency is fixed with respect to the RX frequency UNLESS the "Qsy" button is pressed in which case it shifts to the RX frequency, the Transceiver VFO is shifted and both the RX and TX audio frequencies are shifted to put both into the middle of the transceiver passband. The TX continues to be locked, but at the new audio frequency.

If the "Lk" is ON moving the cursor around will ONLY AFFECT the RX frequency and NOT the TX frequency.

The **AFC** and **SQL** buttons enable or disable the respective function in the software. The slider just above the AFC & SQL controls is the squelch level control. The bar indicator just above it is the equivalent of received signal level and relates on a 1:1 basis with the squelch level slider. The SQL button illuminates YELLOW when the SQL is selected, but the signal is below the squelch level. It illuminates GREEN when the the SQL is selected and the signal is above the squelch level.

The indicator just to the left of the AFC button is the overload indicator. It will be GREEN if your audio drive to sound card is satisfactory, YELLOW if the audio signal is marginally high and turn red when it is in overload. Back down the mixer control or the audio pad from the rig to computer. Fldigi will not perform well if the sound card is over driven. You will see ghost signals on the waterfall and the modem decoders will not work correctly.

Receive audio level should be adjusted so that the overload indicator does not illuminate red. When observing the received signals on the oscilloscope view you should expect that they do not exceed a peak-to-peak amplitude of 3/4 of the full display height.

6.12.1 Mode Status Indicators

The lower left corner of the main display (MFSK-16) in the view above is actually a button disquised as a status panel. This button responds to the mouse in several ways:

- Left Click - opens a quick pick list of associated modem types; you can switch to a new modem type from this popup menu
- Right Click - opens the configuration dialog at the tab associated with the current modem type
- Scroll Wheel - rotates forward and backwards through the various modem types in accordance with the modem menu heirarchy. Stop at the one you want and you are now in that mode

The next status indicator to the right provides information relative to the current modem, for PSK it indicates the received signal strength in dB.

The third status indicator from the left provides additional information relative to the current modem, IMD for PSK measured in dB.

Note that for PSK these values are only measured during periods when the PSK idle signal is being received.

6.12.2 Transmit level attenuator

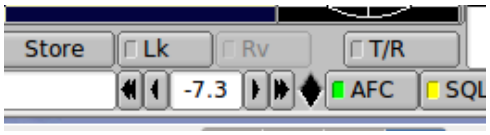


Figure 6.65: TX level

It is often difficult to adjust the audio drive for the point where ALC is just barely active. Mixer controls are OK, but not usually designed for very small changes. They are after all designed for adjusting listening levels. fldigi provides the ability to control the audio drive in increments of 0.1 dB over a 30 dB range. This control is located in the bottom right corner of the main dialog:

Set this control for -6 dB and then adjust the sound card mixer control for the best ALC level you can achieve. Then adjust the Tx-level control for best "just visible" ALC on the transceiver. That should give you a very clean PSK signal.

6.12.3 Operating split with fldigi / flrig

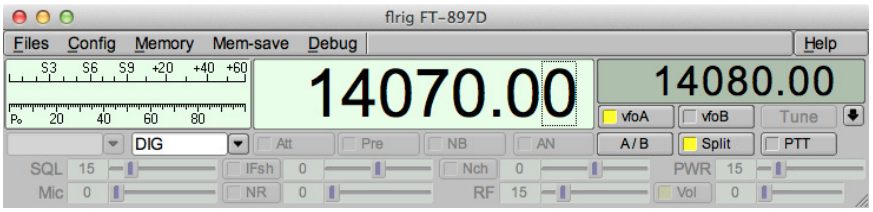


Figure 6.66: TTY Split Ops

Note: In the above example most features are disable as they are not available for the indicated radio.

On flrig:

Unless specifically supported by the transceiver and implemented in flrig.

- VFO A is always the RX frequency.
- VFO B is always the TX frequency.
- Left Click on the A/B swaps A & B VFO frequencies.
- Right Click on the A/B assignes A VFO frequency to B VFO.
- Using the mouse wheel adjust VFO B to where you want to transmit.
- Click the SPLIT button, it will change color when it is active.
- Press the VFO B button to listen on the Tx frequency, be sure to press Split again before capturing that rare DX station.

On fldigi:

- Tune the waterfall cursor to the Rx signal
- Transmit as usual, be sure you are not doubling.

See `FLRIG Help` for specific operating procedures.

Some additional information can also be found here Rig Control Page

Return to Top of Page

Return to Main Page

6.13 Operating Multiple Copies

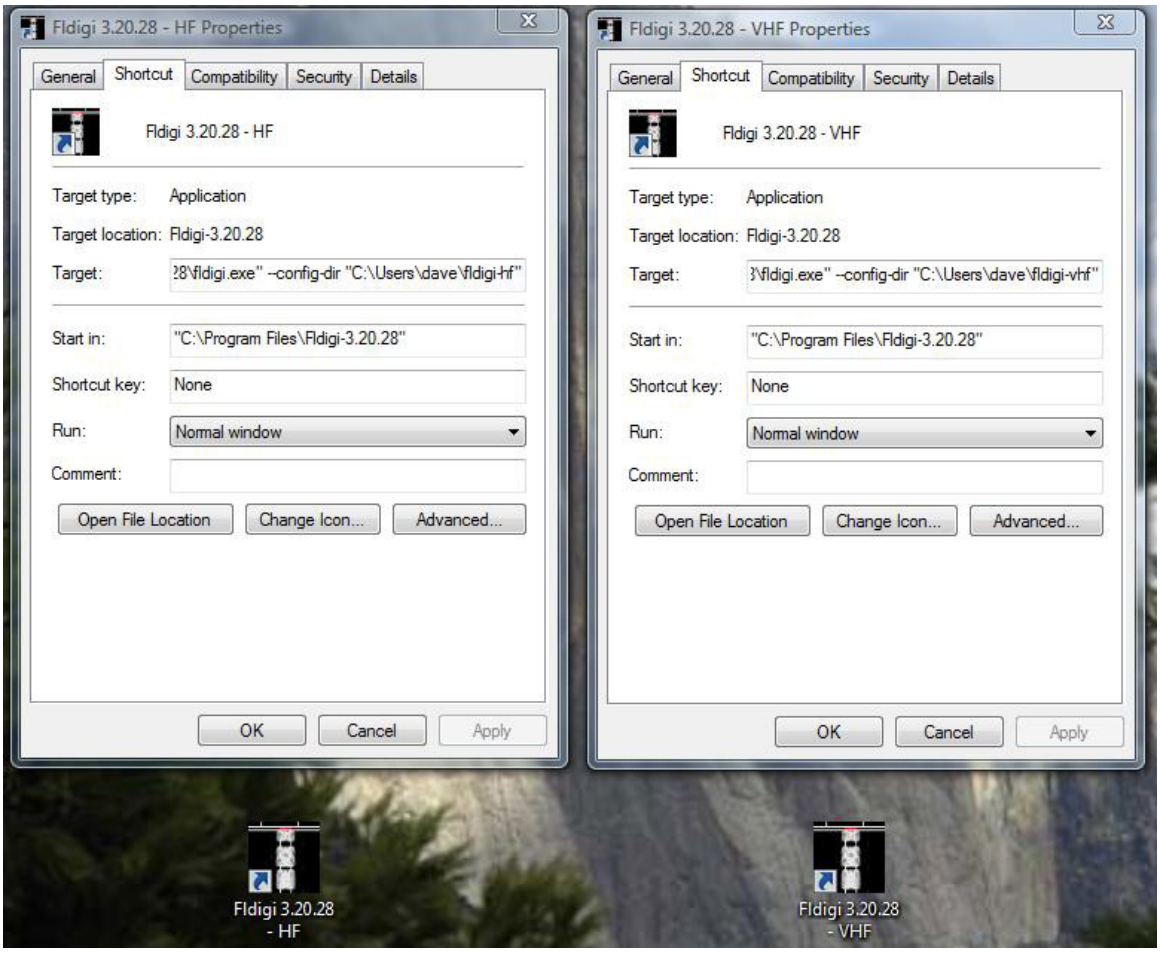


Figure 6.67: Multiple Copies of FLDIGI

There are times that you may need to simultaneously operate two or more instances of fldigi. Or you might simply want to maintain two different configurations based on hardware usage.

The screen shot shows how this is done on Vista, but the process is nearly the same on XP, Win7 and Linux. When you install fldigi it creates a desktop icon launcher. Most of the Linux window managers allow you to create a desktop launch icon. Make as many copies of the launcher as needed for your applications and rename them accordingly. Then change the executable target entry to include the command line switch '–config-dir' followed by the full pathname of the folder that will hold that particular configuration. You do not need to create that folder as fldigi will do so the first time it is launched from that desktop icon.

If the various configurations all use independent hardware, i.e. sound cards and rig control ports, then you can have them operating simultaneously. Each instance will have it's own configuration files, status file, macros, and logbook. It is possible to have each instance use the same logbook, but then simultaneous operation is not possible as the logbook file is not currently designed to allow that type of sharing.

If each instance will be paired with a separate flarq (similarly set up for multiple operation) then you will also need to add the command line switch for arq-server address and port. The same is true for use with applications that talk to fldigi via it's xml-rpc socket port. You change the address/port pairs on both the fldigi launcher and the paired application such as flarq or flrig.

Return to Top of Page

Return to Main Page

6.14 PSK Reporter

The PSK reporter can generate reception reports from three different sources:

- The decoded text, obtained from the Rx Panel text or the multi-channel browser. Note: **the psk browser must be visible** or decoding and subsequent psk reporting will not occur. If you do not want the psk browser panel visible in the main dialog then use the separate browser dialog. It can be minimized and signal decoding will continue.
- The log data
- Data entered manually

The configuration for the PSK reporter in in Misc/Spotting. PSKR needs the following fields from the Oper tab to be non-empty:

1. Callsign (freeform because it's impossible to verify, and because we need to support SWLs without callsigns)
2. Locator (standard 6 character format)
3. Antenna info (freeform, should be kept reasonably short)

Sources (1) and (2) are configurable from Misc/Spotting configuration tab, while (3) is always enabled. To keep the code sane, changing the PSKR options (or the above station info) does not take immediate effect. Instead, the Initialize button changes color to indicate that the changes have not been applied. Clicking on the button will do so (or display an error) for the current and future sessions. This is similar to the Initialize buttons in the rig control configuration.

Here are the options in some more detail:

6.14.1 Automatically spot callsigns in decoded text

The text that is sent to the main window or the PSK browser is continuously searched for callsigns. If this option is enabled, the main window gets a "Spot" light button that can toggle the auto-spotter on and off. It is automatically turned off when playback is selected in the Files menu. The main window text is not searched if the viewer is active, i.e., if it is displayed and the current modem is PSK.

6.14.2 Send reception report when logging a QSO

A reception report is queued for every QSO as soon as it's logged

6.14.3 Report QRG (etc.)

This makes the reception reports include the current rig frequency, adjusted for modem audio frequency and rig sideband. It does not need a click on "Initialize" to take effect. This needs to be an option because it is impossible to tell whether the user has real or "fake" rig control with 100% certainty. Besides that, users may want to run a dedicated spotter for a narrow modes sub-band, and in that case they won't have to synchronise fldigi's frequency display with the rig all that often.

6.14.4 Host and port

With the port set to 14739 the reports will not be entered in the main database, but instead will be analysed and displayed here:

http://pskreporter.info/cgi-bin/psk-analysis.pl

Probably of no interest to anyone who is not hacking on a PSKR client but may be useful for debugging. The PSKR protocol uses UDP with no acknowledgements or retransmissions, so don't be surprised if the occasional report never makes it to the server. There should be enough coverage overlap to make packet loss irrelevant (and save a lot of bandwidth and CPU cycles).

The spotter needs to see a repeated callsign within a short search window, but stations do not always repeat their callsigns. In addition, some operators like to be creative with their macros, and as a result some signals will decode 100% but the callsign will never be auto-captured. Such callsigns can be reported manually.

The manual spotting is done by right-clicking the QRZ "globe" icon. This will generate a report for whatever is in the Call & Loc fields, so make sure that those are correct! You should also verify the frequency (e.g. by placing the waterfall marker on the signal being spotted).

There is a confirmation popup that will open when you right click the "globe" button. The aim of course is to avoid accidentally sending rubbish reports to the PSK reporter database.

Reception reports are filtered for duplicates among all data sources: a report is queued only once every hour for each callsign and frequency band. The queue is flushed every five minutes. You can see what the spotter is doing in the Event Log window or on the terminal if you set the log level to "Info". "Debug" will show all the gory details.

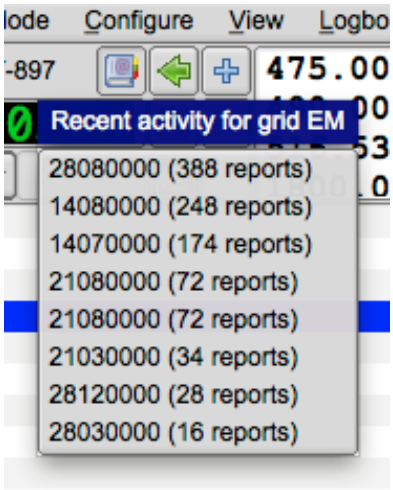


Figure 6.68: Activity Reports

A button and popup text field on the rig control frame give access to the most recent receptions reports in your geographic area.The area is determined by the contents of the field to the right of the button, or by the locator text on the operator tab if the mini field is empty. The first two characters of the locator are used. If the locator is not set, the pskreporter.info uses the current IP geolocation to approximate it.

A popup is displayed when the user clicks the button or presses the Enter key from within the field. The popup shows the frequencies by measure of activity that gives more weight to transmissions. If rig control is active, the user can click on one of the lines to go to that band. Clicking on the 18100000 (1 report) line would immediately QSY the transceiver to 18.1 MHz.

The data is retrieved from `http://pskreporter.info/cgi-bin/psk-freq.pl`

or with a filled text field, `http://pskreporter.info/cgi-bin/psk-freq.pl?grid=TEXT`

There is a link to the pskreporter.info map page in the Help menu.

Return to Top of Page

Return to Main Page

6.15 Rig Control

Figure 6.69: Initial Rig Setup

CAT not enabled

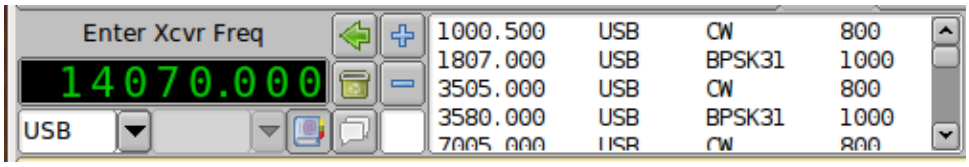


Figure 6.70: Docked Rig Control

CAT not enabled, Manual Entry of transceiver frequency

Note: The same control is also used for both manual entry of the transceiver frequency or with full CAT control. When no CAT is available the control is simply a convenient way of keeping track of the transceiver USB/LSB suppressed carrier frequency, the mode and the audio tracking point. If fldigi is being used with an FM transceiver you probably should enter the simplex frequency or the input frequency of a repeater being used. This frequency value is used with the waterfall audio frequency to compute the logged frequency. The logged frequency value will only be correct for LSB and USB operation.

The frequency/mode pick list is displayed when the book button is pressed. Pressing the book button a second time will restore the original logging panel.

The pick list buttons control selecting, adding and deleting entries in the frequency/mode list.

- add the current frequency / mode / audio track point to the list
- select the current list entry
- delete the highlighted entry from the list
- delete all entries from the list (a warning prompt will appear)
- show active frequencies based on either the entry field to the right or the stations locator, see pskreporter/spotter.
- entry field for active frequencies search, for example "EM."

The browser list contains frequency, sideband, modem type and audio frequency. The list is saved when fldigi is shut down.

The combo box on the left will allow the selection and control of the operating mode of the transceiver.

The combo box on the right will allow the selection and control of the transceiver bandwidth.

The frequency display is in fact a set of special buttons. Each digit may be left-clicked to increment in frequency by that digit value, or right clicked to decrement by that digit value. The leading digits will follow suit if a decade rollover occurs. You can also place the mouse cursor on a digit and then use the mouse wheel to roll the frequency up and down.

Manual entry of frequency can be accomplished by clicking on any digit and then entering the numeric value in KHz. Don't forget the decimal point if you are entering a fractional KHz value.

The mode combobox, the bandwidth combobox and the frequency display also annunciate the current transceiver status. If you change operating mode on the transceiver, that will be annunciated in the respective combobox and fldigi will adjust any internal parameters accordingly. Fldigi queries the transceiver 10 times per second to maintain a lock step with the transceiver.

6.15.1 Rig Configuration

Hardware PTT control

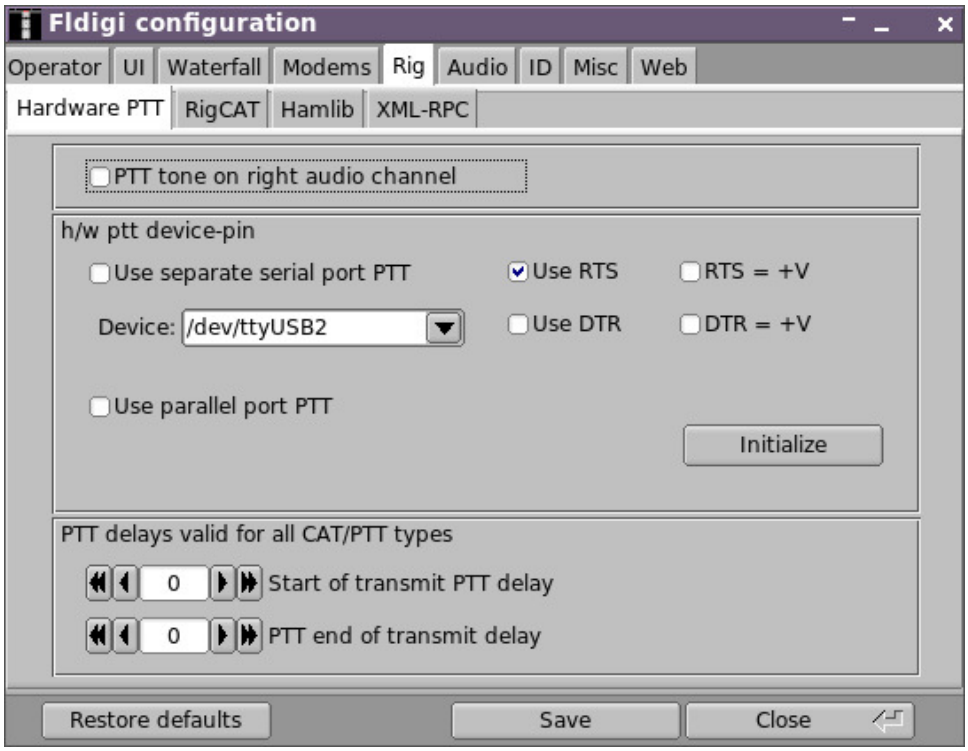


Figure 6.71: Hardware PTT control

Right Channel VOX Signal

Fldigi can generate a 1000 Hz tone for the duration of the PTT keydown period. A simple tone detector/filter and transistor switch can be used to generate a PTT signal from this sound card output. Jim, W5ZIT, has provided details on building an interface for this type of PTT control.

Serial Port using DTR or RTS

The simplest rig control is just being able to control the push to talk via an external transistor switch. You set this type of control on the first configuration tab for rig control.

You select this operation by checking the "Use serial port PTT". Select the serial port from the list (fldigi will have searched for available ports). Then specify whether the h/w uses RTS or DTR and whether a + or - voltage is required to toggle PTT on.

You can use a serial port for control with the RTS and DTR pins configured for you particular interface. The program allows you to use RTS, DTR or BOTH for the PTT signal. Press the Initialize button to start the serial port.

Parallel Port (Linux and Free BSD only)

Fldigi sets and clears the parallel port pin, PARPORT_CONTROL_INIT, pin 16 on the 25 pin parallel port connector. Keydown sets Pin 16 to +5 volts and keyup sets the voltage to zero.

μH Router (MacOS X)

Simular functionality can be achieved on the Macintosh operating system using 'μH Router' by Kok Chen, W7-

AY. See µH Router Website for specific details and requirements. A selectable (check box) option will be available on the Rig->Hardware PTT Configuration panel.

PTT delays

You can accommodate delays in transceiver switching between receive and transmit by adjusting the PTT delays. The control values are in milliseconds. These controls have no effect on external PTT circuits such as those implemented in the SignaLink interfaces. They rely on detecting the audio data stream. You can use a combination of macro tags in a macro key definition to achieve a resolution. For example try a macro definition similar to this to insure that the RSID is sent via a slow FM xcvr (or via a VHF repeater)

```
<TX><MODEM:NULL><IDLE:2.5>
<!MODEM:MT63-500>
<TXRSID:on>
```

Change the idle time value (in fractional seconds) to suit your needs.

6.15.2 RigCAT control

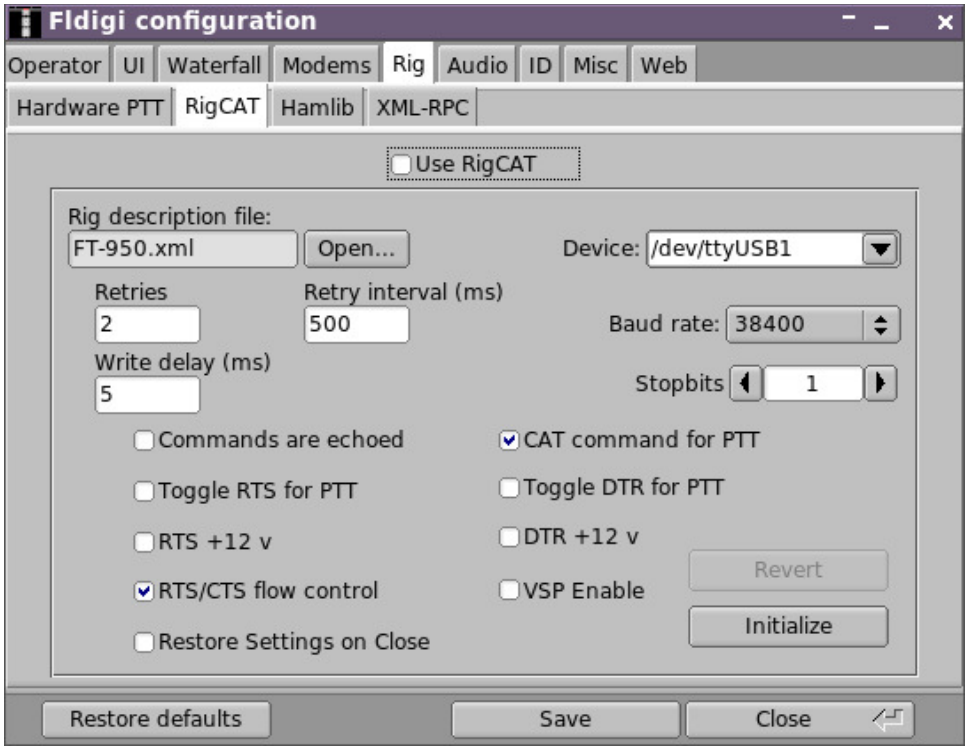


Figure 6.72: RigCAT control

RigCAT is a rig control system similar to hamlib that was developed specifically for fldigi. It uses command / response definitions that are found in various rig.xml files. You can use a rig.xml file specific for your transceiver or write and test one yourself. The easiest way is to adapt an existing rig xml file for a rig that is similar to your own. ICOM almost identical command/response strings for all of its transceiver line. Yaesu rigs have nearly all used unique command/response structures until just recently. The TS-450, TS-950 and others share a similar set of commands and responses.

RigCAT commands and responses are defined in a rig specific xml file which contains all of the required queries and responses in extended markup language format. Please read the specification document rigxml to learn more about this new way of building generic rig interface definitions and how they are used with fldigi. fldigi will look for a file in the $HOME/.fldigi/rigs directory for all files with extension ".xml". These contain definitions for the transceiver indicated by the file name, ie: FT-450.xml, IC-756PRO.xml, etc. You can download the appropriate xml files from

the resource directory tree http://www.w1hkj.com/xmls or from the archives web page. Place the file in your rigs directory and fldigi will find it.

You will need to specify how your PTT will be triggered. This can be using a CAT command, the RTS or DTR pins or none. None would be appropriate if you are using the rig's VOX or an outboard sound card interface such as the SignalLink SL-1+ which produces its own VOX type of PTT. In that case simply leave all of the PTT options unselected.

If you are using a transceiver or a rig interface such as CI-V that echos all serial data you check off the "Commands are echoed" box. That will suppress fldigi trying to respond to a command it just sent to the transceiver.

You may need to try various values of retries, retry interval, and command interval to achieve consistent rigcat control.

Press the Initialize button after setting all of the parameters. If the settings are all correct fldigi should start receiving frequency information from the rig and annunciating them on the rig control frequency display.

6.15.3 Hamlib CAT control

Hamlib is a set of standard libraries for interfacing to a large number of transceivers. The hamlib library system consists of a front end which acts on behalf of all rigs and backends which are specific to each rig.

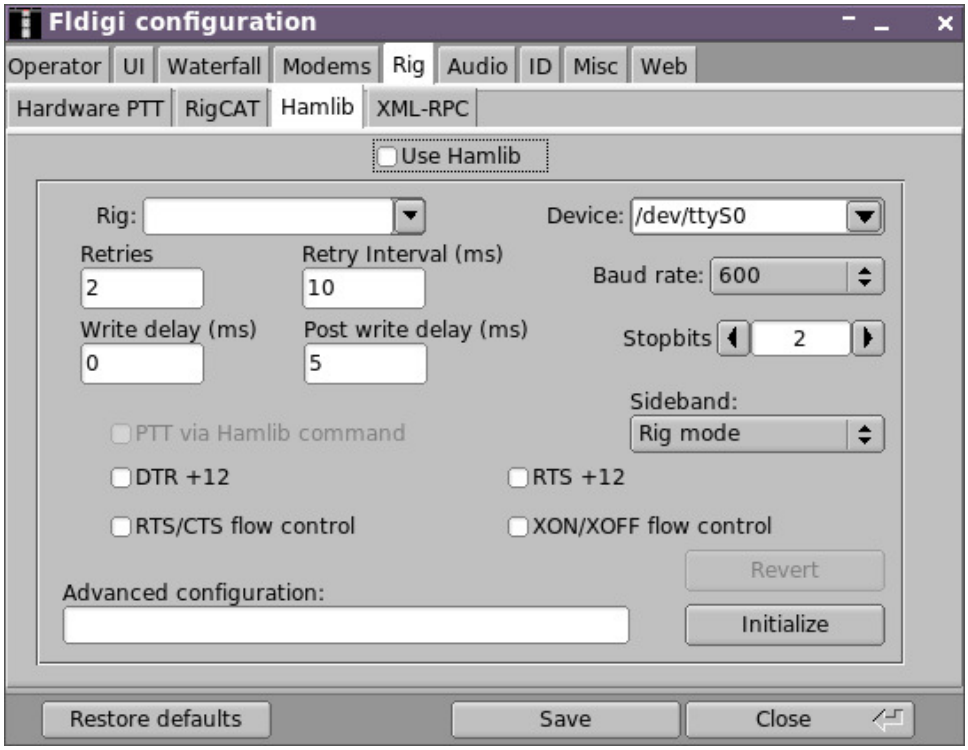


Figure 6.73: Hamlib CAT Control

Select your transceiver from the list of supported units. Then select the serial port and baud rate. If you are familiar with the hamlib library you can send various startup sequences to the rig using the advanced configuration. PTT control can be achieved using CAT commands or via DTR / RTS on the same port as the control comms. You might also need to specifiy whether RTS/CTS flow control is uses (Kenwood rigs use this quite often) or if Xon/Xoff flow control is used.

You may need to try various values of retries, retry interval, and command interval to achieve consistent hamlib control.

Press the Initialize button after setting all of the parameters. If the settings are all correct fldigi should start receiving frequency information from the rig and annunciating them on the rig control frequency display.

6.15.4 Xml-Rpc CAT

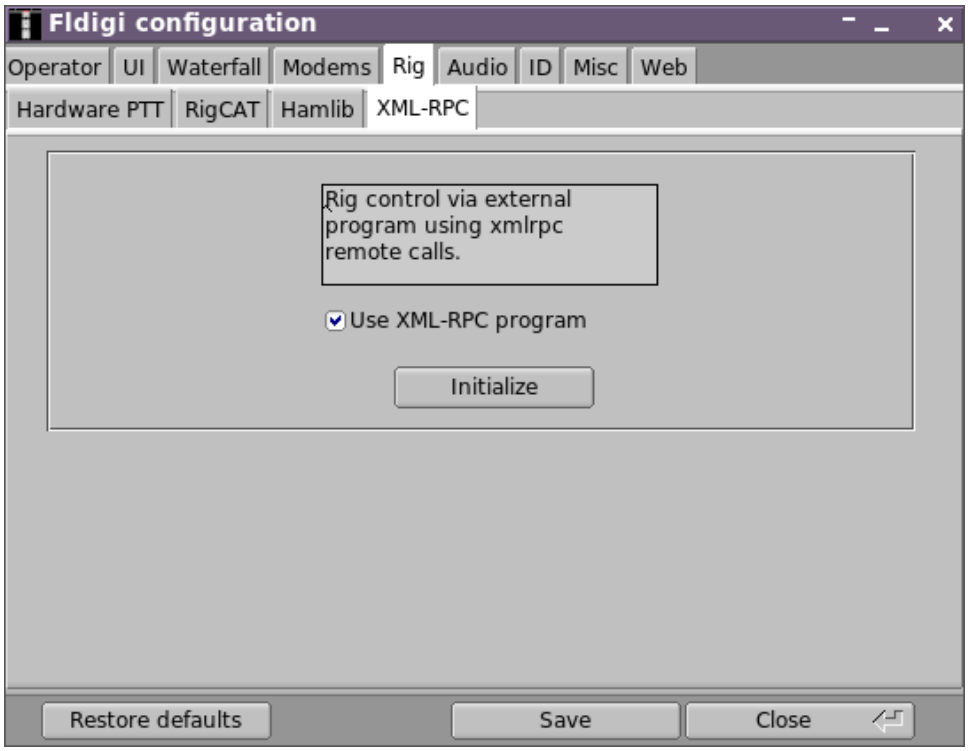


Figure 6.74: Xml-Rpc CAT Control

Xml-Rpc allows third party software to control various aspects of fldigi operation including but not limited to rig control. This is the data interface that is also used by the program **flrig**, a fldigi companion transceiver control program.

If you are using a third party interface such as DxKeeper Bridge you might be instructed to select this method of CAT.

Return to Top of Page

Return to Main Page

6.16 RST and RSQ Reporting

6.16.1 RST

Is the traditional Readability, Strength, Tone reporting system used for CW operations for nearly as long as amateurs have enjoyed the airwaves.

6.16.1.1 READABILITY

1. Unreadable
2. Barely readable, occasional words distinguishable
3. Readable with considerable difficulty
4. Readable with practically no difficulty
5. Perfectly readable (that is 100% print in todays jargon)

6.16.1.2 SIGNAL STRENGTH

1. Faint signals, barely perceptible
2. Very weak signals
3. Weak signals
4. Fair signals
5. Fairly good signals
6. Good signals
7. Moderately strong signals
8. Strong signals
9. Extremely strong signals

6.16.1.3 TONE

1. Sixty cycle AC or less, very rough and broad
2. Very rough AC, very harsh and broad
3. Rough AC tone, rectified but not filtered
4. Rough note, some trace of filtering
5. Filtered rectified ac, but strongly ripple modulated
6. Filtered tone, definite trace of ripple modulation
7. Near pure tone, trace of ripple modulation
8. Near perfect tone, slight trac of modulation
9. Perfect tone, no trace of ripple, or modulation of any kind

6.16.2 RSQ

Give the report as RSQ for digital modes, but especially BPSK and QPSK; see: http://www.psb-info.-net/RSQ-Reporting-Table.html

6.16.2.1 READABILITY

1. 0% undecipherable
2. 20% occasional words distinguishable
3. 40% considerable difficulty, many missed characters
4. 80% practically no difficulty, occasional missed characters
5. 95%+ perfectly readable

6.16.2.2 STRENGTH

1. Barely perciptible trace
2. n/a
3. Weak trace
4. n/a
5. Moderate trace
6. n/a
7. Strong trace
8. n/a
9. Very strong trace

6.16.2.3 QUALITY

1. Splatter over much of the visible waterfall
2. n/a
3. Multiple visible pairs
4. n/a
5. One easily visible pair
6. n/a
7. One barely visible pair
8. n/a
9. Clean signal - no visible unwanted sidebar pairs

Return to Top of Page

Return to Main Page

6.17 Signal Browser

PSK and RTTY signals can be viewed in a multi-channel context. You can open an embedded or a separate dialog to access the browser. These browsers can help to locate a signal of interest on a busy band. The browsers can be visible at any time, but are only active when fldigi is in one of the PSK or RTTY modes. Open the external by clicking on the **View**/**Signal browser** menu item.

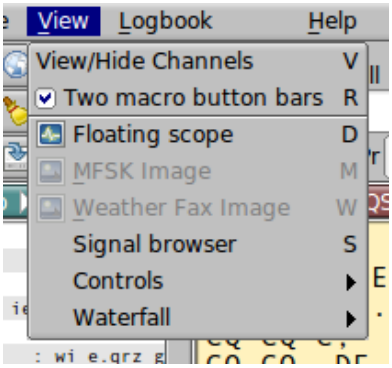


Figure 6.75: View Menu

It looks like this:

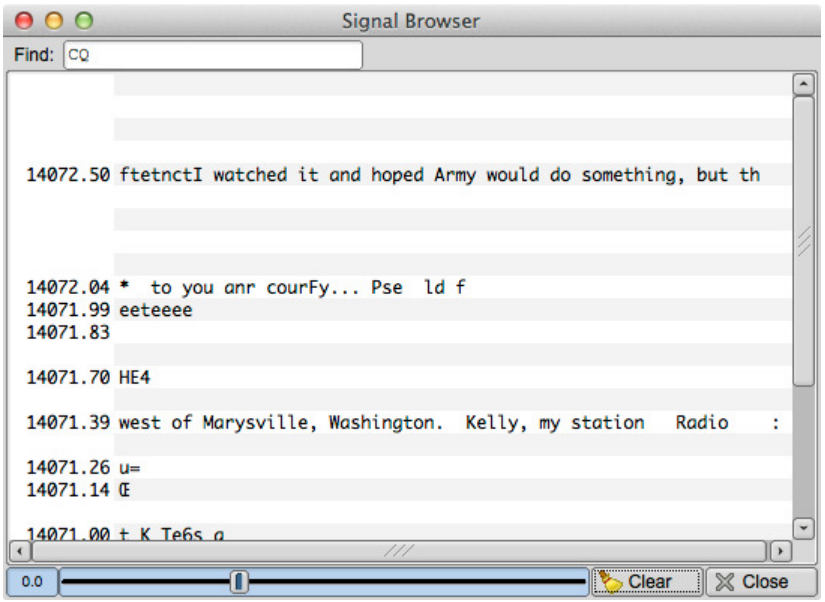


Figure 6.76: Browser Panel

Toggle the visibility of the embedded viewer by selecting the ***View/Hide Channels*** on the same menu list. Your main fldigi screen then opens a browser panel on the left side of the Tx/Rx panel(s).

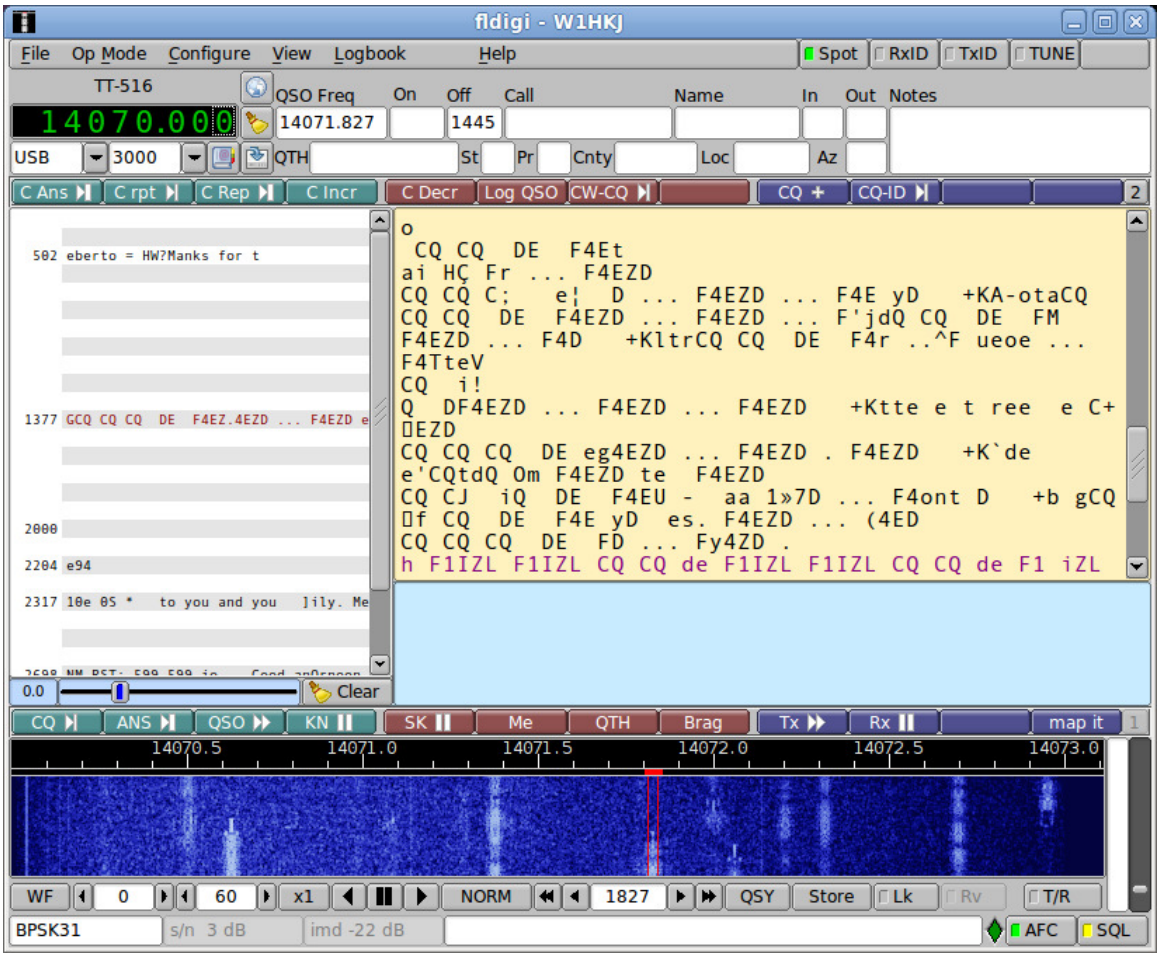


Figure 6.77: Main Dialog Browser

This panel can be resized horizontally by dragging the interface between the browser and the Rx panel to the left and right. You can drag all the way to the left to close the panel (or use the menu button). The drag to the right is limited to prevent sizing the Rx/Tx panels below their allowable limits.

The browser can decode up to 30 simultaneous signals. As each signal is acquired within a 100 Hz channel width it is printed on the associated line. The user can elect to have each line annotated with

a channel number,

the waterfall audio frequency,

the transceiver HF frequency + waterfall, or

no annotation.

Left click on a line of received text to move the waterfall frequency and Rx panel tracking to that signal. The contents of the line of text is transferred to the Rx text widget, and the main signal processing loop begins to track and decode that signal. Right click on a line of received text and that line is cleared and the channel reset for a new detection. You may have to do this occasionally if the detector for that channel has locked on to a sideband of a PSK signal. This is most likely to occur when the received signal as a marginal or bad IMD. Use the >**Clear** button to clear and reset all of the channels. Channel signal detection and processing should restart immediately.

Configuring the browser:

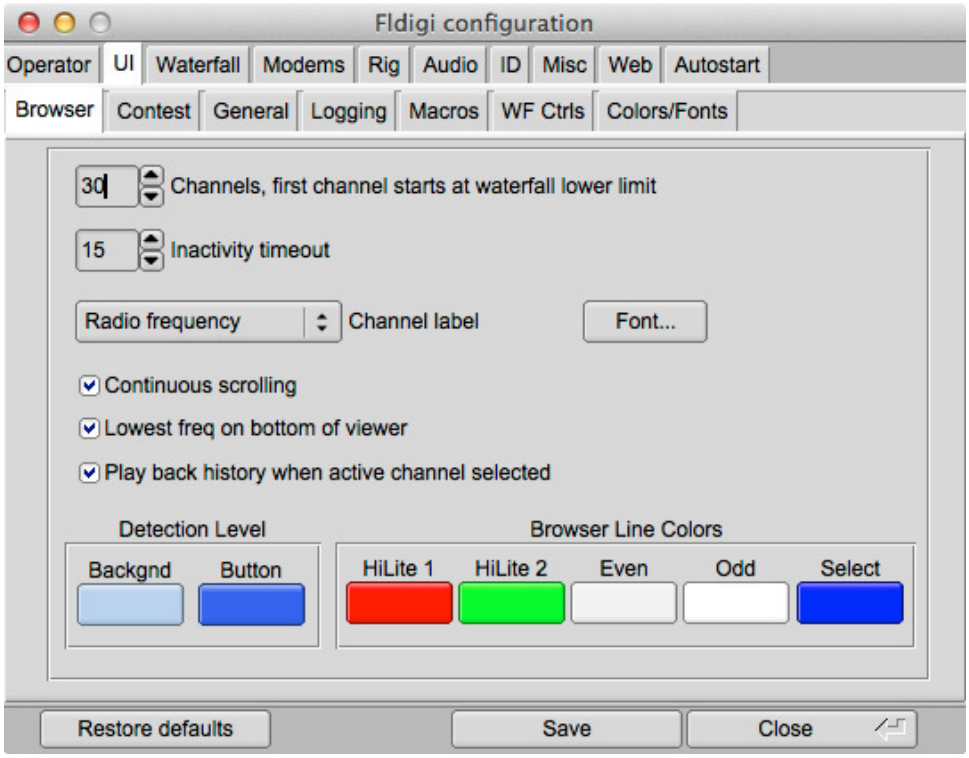


Figure 6.78: Browser Configure

1. select the number of 100 Hz channels you want visible
2. select the start frequency for the browser (your transceiver may not rx signals below this value)
3. select the inactivity timeout for the browser. After this number of seconds the channel will be cleared and prepared for the next detection cycle.
4. select whether you want a Marque type of continuous scrolling, or simply clear the line when it is filled.
5. select what kind of label annotation you want on each line
6. select the font and font size to be used in the browser
7. You can enter any text you want to search for in the **Seek Regular Expression** widget.

   This text can be a simple text snippet such as "CQ" or any regular expression. With a regex you can specify a more generic pattern, which means that you can match more things and your search is somewhat noise tolerant. Here is an example for a CQ from a US station (should match most callsigns):

   cq.+[aknw][a-z]?[0-9][a-pr-z][a-z]{1,2}

   This says "cq followed by at least one character, followed by one A, K, N, or W, followed by an optional letter, followed by a digit, followed by a letter that is not q, followed by one or two letters". The search is case-insensitive.

   All plain text is a valid regular expression, unless you really had been looking for these metacharacters:

   .[{()*+?|^$

   These will have to be escaped with a backslash.

Return to Top of Page

Return to Main Page

6.18 SYNOP decoding

6.18.1 What is Synop

6.18.1.1 Weather data

SYNOP, which stands for **surface synoptic observations** is a numerical code used for reporting weather observations made by manned and automated fixed land weather stations. It is also called FM-12 by the

World Meteorological Organization. It is closely associated to FM 13-XIV SHIP, for report of surface observation from a sea station (Created in 1982) and FM 14-XIV SYNOP MOBIL, for reports of surface observation from a mobile land station.

Here is an example of undecoded SYNOP weather report:

```
13586 31530 80000 10036 20029 39821 40218 53002 71022 886//
333 20029 88715=
```

Many SYNOP reports are available on web sites such as Ogimet .

6.18.1.2 Synop reports broadcasts

The most important broadcaster on SYNOP data is DWD, in RTTY mode.

6.18.1.2.1 Deutsche Wetterdienst schedule

SYNOP reports are typically sent every six hours by Deutscher Wetterdienst on shortwave and low frequency using RTTY.

The baud rate must be 50 bauds.

Freq	Station	Time	
147,300 kHz	DDH47	05.00 - 22.00	+ / - 42,5
4583,000 kHz	DDK2	00.00 - 24.00	+ / - 225,0
7646,000 kHz	DDH7	00.00 - 24.00	+ / - 225,0
10100,800 kHz	DDK9	00.00 - 24.00	+ / - 225,0
11039,000 kHz	DDH9	05.00 - 22.00	+ / - 225,0
14467,300 kHz	DDH8	05.00 - 22.00	+ / - 225,0

6.18.1.2.2 Other broadcasters

Until recently, many shore stations used to transmit Synop reports by Morse Code ("*Ocean Yacht Navigator*", Kenneth Wilkes, 1976). Still, according to WMO information for shipping, a couple of maritime stations might broadcast these informations in different mode that RTTY, but this has not been tested yet. There are just anecdotically cited here.

Maritime Radio Station of Mumbai

Met codes: FM 12 - SYNOP, FM 13 - SHIP for Sea areas I.

Band	Freq
WT (MF)	500 kHz
WT (MF)	521 kHz
WT (HF)	8630 kHz
WT (HF)	12710 kHz

SYNOP decoding is not accessible fromMorse CW modem yet.

Navtex station of Mumbai. COMCEN

FM 12 SYNOPSIS, FM 13 SHIP for sea area I (Time schedule 0900, 2100)

	Freq
Navtex (Also called MSI as Maritime Safety Information)	518 kHz

SYNOP decoding is not accessible from Navtex modem yet.

Marine Rescue Co-ordination Centre: Guayaquil, Radio IOA (Instituto Oceanografico Armada) o Radio Naval

Meteorological information in process: Text, SHIP, SYNOP.

Mode	Freq
AM	1515 kHz

6.18.2 Parameters

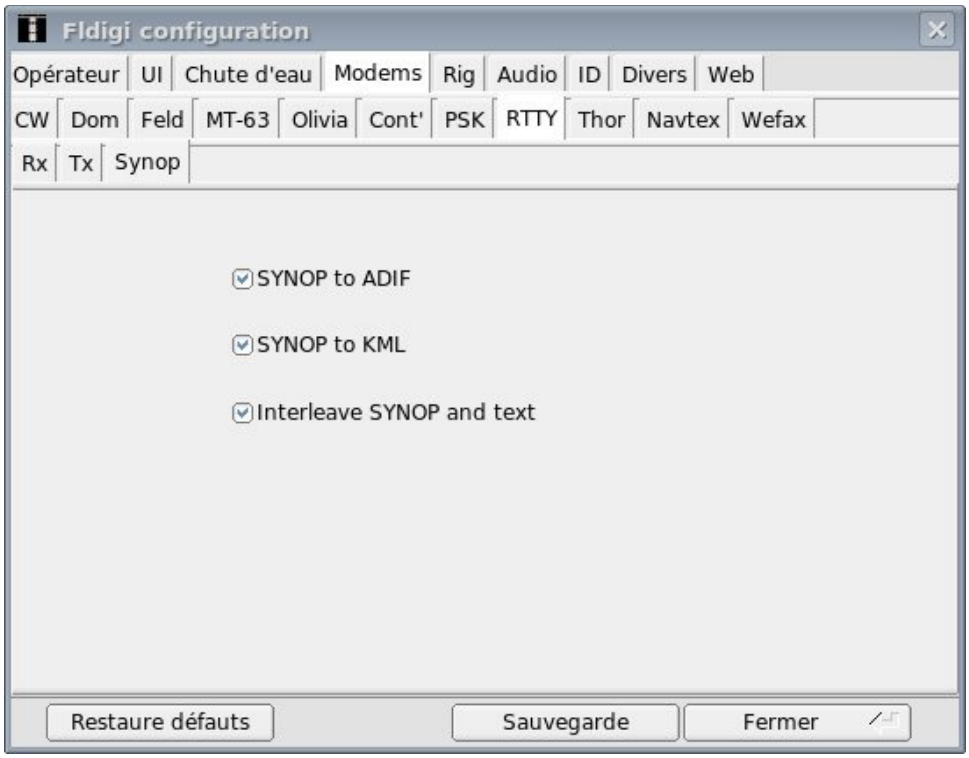


Figure 6.79: Synop configuration in RTTY tab

By default, there is not SYNOP decoding of RTTY reception, so its behavior does not change. However, it receives three new parameters related to Synop decoding.

SYNOP reports can be saved to the current ADIF log file: They are visible in the journal of QSOs with the content of the report, the timestamp, and the Maidenhead locator equivalent to the geographical coordinates of the emitter of the report.

SYNOP weather reports can also be plotted on KML files: They can then be visualized in KML viewers such as Google Earth.

If SYNOP reports are saved either to ADIF or KML files, fldigi attempts to detect and decode RTTY messages. When these messages are detected, they are displayed in the reception window. An extra parameter allows to replace the coded Synop message, giving clarity at the risk of loosing information, or mix the decoded reports with the original group of digits (Interleaving of SYNOP and text).

6.18.3 Reception window

When SYNOP messages are detected, they are displayed in the reception window in "red" characters (depicted here in "*italic*"). Here are some examples (Interleaved text and SYNOP reports):

```
13661 16123 99276 70216 46///
WMO Station=13661
WMO station=WMO_13661
UTC observation time=2013-02-16 12:00
QLLLL token=Present
Longitude=-21.6
Latitude=27.6
/// 40206 52011
222// 00203;
Sea surface temperature=20.3 -C
Temperature type=Intake measurement
64522 16123 99591 70347 46///
WMO Station=64522
WMO station=WMO_64522
UTC observation time=2013-02-16 12:00
QLLLL token=Present
Longitude=-34.7
Latitude=59.1
/./ 49920 57004.
::(1056;
, 26555 16123 998#6
WMO Station=26555
WMO station=WMO_26555
UTC observation time=2013-02-16 12:00
70269 46/// ///// 40168 52008;
WMO Station=70269
WMO station=WMO_70269
Precipitations=Omitted, no observation
Station type=Automated station.  No observation (No 7WW)
Visibility=Missing
Wind direction=No motion or no waves
Wind speed=0 knots (Estimated)
Sea level pressure=1016 hPa
Pressure tendency=Increasing steadily.  Raises
Pressure change=0.8 hPa
25617 16123 99867 1173!((6;
WMO Station=25617
```

```
WMO station=WMO_25617
UTC observation time=2013-02-16 12:00
QLLLL token=Present
149 52001;
13660 16123 992!;
WMO Station=13660
WMO station=WMO_13660
UTC observation time=2013-02-16 12:00
).'/ ///// 40222 52&17
52"// 08)02;
44551 16123 99376 70378 46//.
WMO Station=44551
WMO station=WMO_44551
UTC observation time=2013-02-16 12:00
QLLLL token=Present
Longitude=-37.8
Latitude=37.6
/// 40185 52022;
Sea level pressure=1018 hPa
Pressure tendency=Increasing steadily.  Raises
Pressure change=2.2 hPa
62680 16123
9958"
WMO Station=62680
WMO station=Atbara
UTC observation time=2013-02-16 12:00
70265 66//?  BMPROIIO TUPEQ
222!/ 00078;
64527 16123 99593 70483 46///
WMO Station=64527
WMO station=WMO_64527
UTC observation time=2013-02-16 12:00
QLLLL token=Present
Longitude=-48.3
Latitude=59.3
/// 40095 52004
```

6.18.4 Data files

Several data files containing public information are used to decrypt SYNOP data. They come from various places on the Internet, with redundancy.

SOURCE	DESC.	Web download
NOAA	WMO ident'	weather.noaa.gov/data/nsd_-bbsss.txt
NOAA	Wx buoys	tgftp.nws.noaa.-gov/data/observations/marine/stations/station-_table.txt
UK Wx Service	Wx Ships	www.metoffice.gov.-uk/media/csv/e/7/ToR-Stats-SHI-P.csv
IOC Meterology	Argos/Iridium	ftp.jcommops.org/JCOMMOPS/G-TS/wmo/wmo_list.txt

There is a specific window for downloading the latest version of these data files. It can be reached with the command "File/Folders":

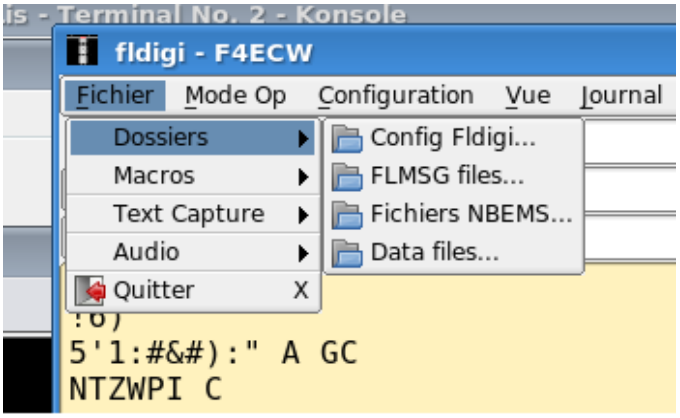


Figure 6.80: How to reach the data files sources menu

These data files can also be found:

- At installation time your fldigi binary comes with a package. All data files are silently installed on your machine.
- When building your own binary from the sources, with the command **make install**. Similarly, all files are installed.
- At first use and automatic installation, this opens the Data files sources menus, where you can download all or some data files used by fldigi.

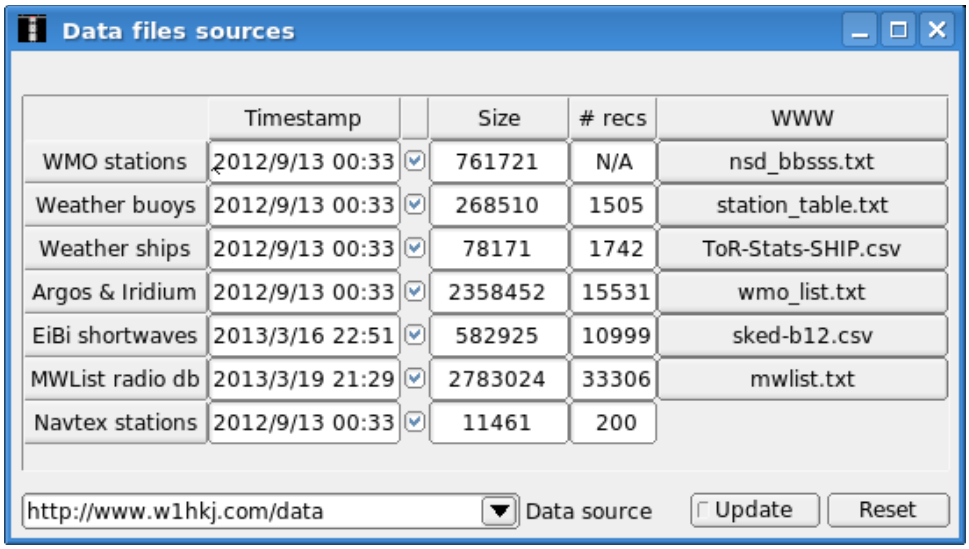


Figure 6.81: Downloading and updating data files used for various decodings

6.18.4.1 Outputs

Each message is a set of key-value pairs, associated to:

Description	Measurement
Wind direction	315 degrees
Wind speed	8 m/s (Anemometer)
Temperature	0.0 deg C
Dew point temperature	-3.8 deg C
Station pressure	1006 hPa
Sea level pressure	1030 hPa

6.18.4.2 Output formats

6.18.4.2.1 ADIF

When the option is ticked, SYNOP weather reports are saved in a new ADIF record, in the default logfile.

The report itself is saved with field <NOTES:xx>. The length can be of several hundredth of chars, and accordingly to the ADIF standard, new-lines can appear in each report.

Example of <NOTES> field in an ADIF record:

```
Header
ICAO indicator=LOWM
Identification and location
Land station observation
Land observations
Precipitations=Precipitation omitted, no precipitation
Station type=Manned station. 7WW omitted, not significant
Cloud base=600 to 1000 m
Visibility=15 km
Cloud cover=7/8
```

6.18.4.2.2 KML

SYNOP reports can generate KML files for fixed stations, mobile weather ships, buoys etc... that is, every observation which can be associated to a set of coordinates. Several reports of the same station can be aggregated in a single place mark. Mobile stations have their path drawn, linking all coordinates spotted for a given station with an unique name.

KML data are are also saved in a display-independent format in <description> tags, and thus can easily be reused by other software.

6.18.5 Command-line SYNOP decoder program : synop_tool

The decoder comes with synop_tool, a command-line tool able to decode input text files and generate the same output files as fldigi. It is intended as a development for checking internal consistency, and accuracy of SYNOP decoding without the complexity of the graphical interface.

It is accessible in the directory fldigi/src/synop-src and must be built with the command make.

Its command-line options are displayed with the option -help:

```
[fldigi/src/synop-src]$ ./synop_tool --help

Valid options are:
    data_dir  # Where the data files are stored. For example /usr/local/share/fldigi/
    kml_dir   # Out put directory of created KML files.
    load_dir  # Input directory of KML files loaded at startup.
    dbg       # Verbose mode.
    usage     # Print this message.
    test      # Decoding preceded by an internal test.
    matrix    # KML output in aggregated matrices (See KML documentation).
    regex     # text output of regular expressions, not decoded reports.
    version   # Prints version number.
    help      # Prints this messages
```

Return to Top of Page

Return to Main Page

6.19 UTF-8 support

Fldigi supports the UTF-8 character set in all of it's labels and widgets (controls). The Rx and Tx text panels accept both keyboard and text stream that is in UTF-8 format. Transmission of UTF-8 characters is supported by these modes:

- PSK, PSKR, and Multi-Channel PSK/PSKR
- Olivia - must enable 8 bit extended characters
- MT63 - must enable 8 bit extended characters
- MFSK - all baudrates supported

For example, the following Russian and annotated texts can be sent and received:

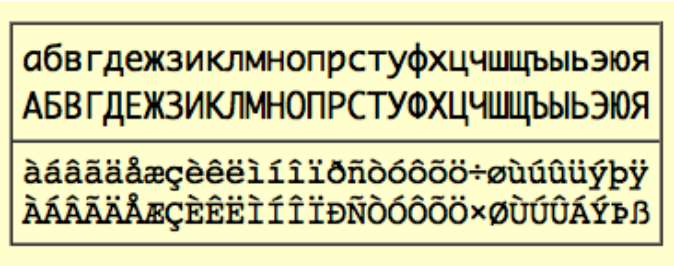


Figure 6.82: UTF-8 Encoded Characters

UTF-8 characters are represented by 2 bytes which means that a single character will take longer to transmit. Transmission speed will be further reduced on modes like PSK where the character to bit stream conversion has been optimized for the English language.

Return to Top of Page

Return to Main Page

6.20 Using the FLdigi Rt. Channel for transmitter PTT

This interface is designed to take advantage of the Right Speaker tone output in FLDigi for actuating the PTT. It is especially useful for interfacing a small notepad with minimal I/O capability to a radio. It has been my experience that complete ground isolation between a laptop and the radio (including eliminating any kind of rig control) provides the minimum QRM to the receiver from the laptop. The Headphone and Mic connections are the only computer connections necessary.

The tone output is coupled through a transformer to maintain complete ground isolation between the computer and the radio. There is no external power required for this interface and an FET is used to provide the PTT output. I use transformers in both the audio input and output paths to complete the isolation.

Note that two of the transformers are wired to the single stereo jack from the Speaker output of the computer. The normal Left Speaker Output (TIP) is used to provide the modulating audio for the radio transmit input while the Right Speaker Output (RING) provides the tone output that drives the PTT circuit. The SHELL of the connector provides the return path for both signals.

Audio from the radio is supplied to the MIC input on the computer for demodulation by the FLDigi program.

Somewhat higher output from the computer is required to reliably operate the PTT circuit and the audio balance control in the computer is used to reduce the output to the radio to the level required for normal modulation. Since the right speaker tone output is not the same audio used to modulate the radio, no signal distortion is caused by the clipping action of the diode rectifiers used in the PTT circuit.

To configure FLDigi to use this interface go to (Configure/Rig Control/Hardware PTT) and check only the (PTT tone on right audio channel) box.

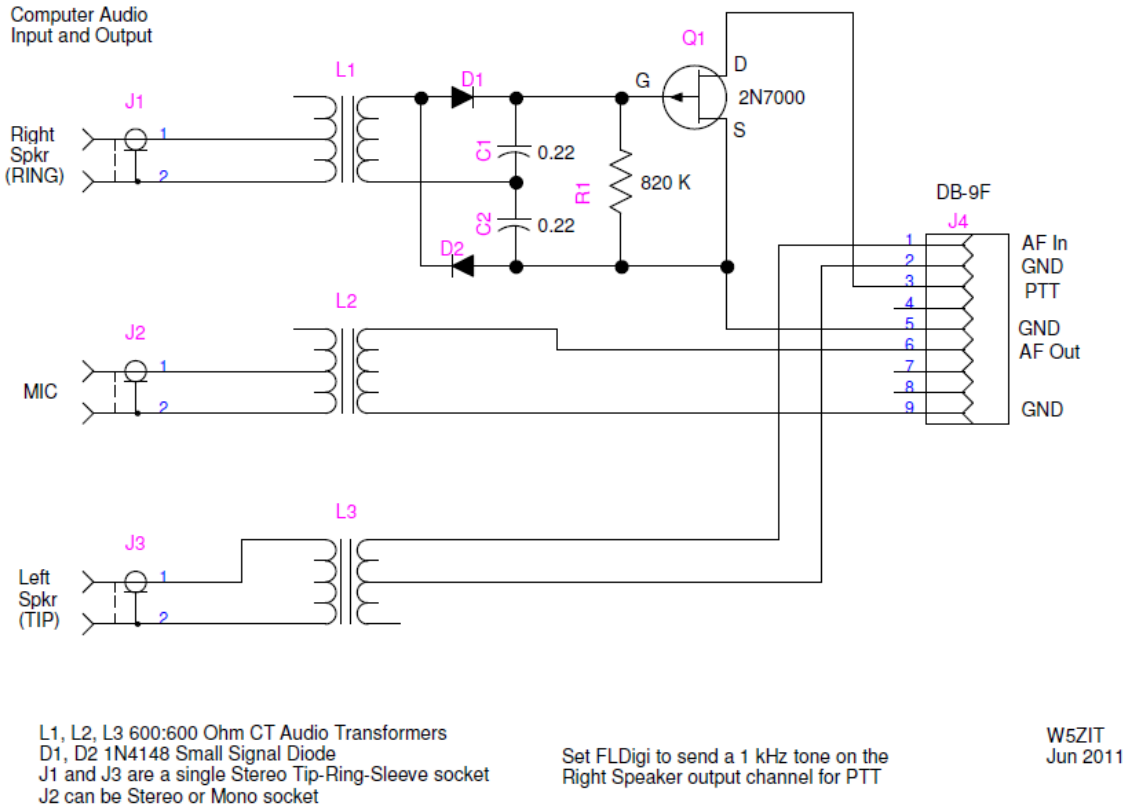


Figure 6.83: W5ZIT Interface

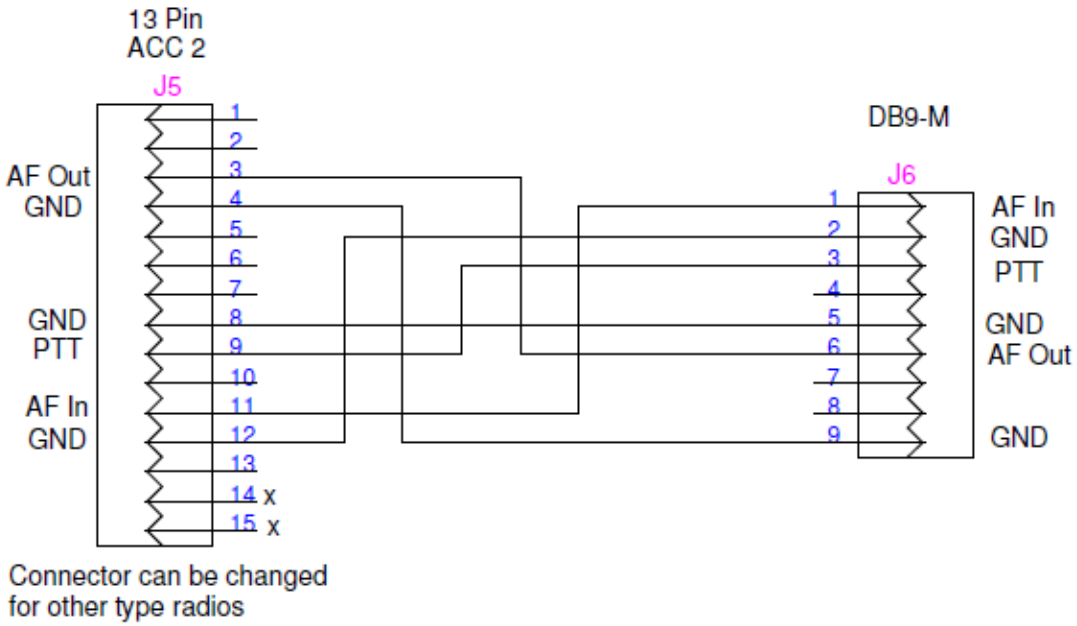


Figure 6.84: TS-2000 ACC-2 to DB9 Adapter

Return to Top of Page

Return to Main Page

6.21 WWV transmit mode

The WWV mode is normally used to measure the offset of the sound card oscillator. (see WWV ppm measurement).

It can also be used to transmit time tick signals similar to the format that is used by WWV. The WWV modem sends a time tick which is a 200 msec wide pulse at a once per second rate when the T/R button is enabled. This pulse is shaped with a 4 msec raised cosine shape on the leading and trailing edges to reduce key clicks. The accuracy of the transmitted time tick is solely dependent on the accuracy with which the WWV ppm measurement was performed.

The purpose of the WWV time tick transmission is to allow other stations to calibrate their sound cards against your calibrated system. This can be used to align all systems in a VHF/UHF net for example. It is only necessary for one of the net members to be able to calibrate his or her sound card against WWV. The other's would then be calibrated by proxy use of the WWV time tick transmit mode. This can even be used in the case where no member has access to a HF transceiver. The "master" station would set the Rx and Tx ppm settings to zero. It would then transmit the time tick signal for the other stations to calibrate their sound cards against the master sound card. Having all of the stations calibrated in this way will insure that the modem decoders will give maximum performance. Here is an example of an advanced macro that will send a CW announcement, 2 minutes of time ticks and end with another CW announcement.

```
<MODEM:CW>
<!GOFREQ:1000>
<!WPM:24>
QRZ QRZ de <MYCALL> <MYCALL>
2 minute time tick cal run follows
<IDLE:2>
<!MODEM:WWV><!IDLE:120>
<!MODEM:CW><!IDLE:2>
end of time tick run
de <MYCALL> k
<TX><RX>
```

Return to Top of Page

Return to Main Page

6.22 Working Logs

Working Logs

Fldigi maintains a number of working log files that are found in its default folder. The default folder is easy to find, simply select the menu item "File/Show config" and your OS default files explorer will be opened to that location.

Rx/Tx Capture File

Everytime you start or stop fldigi that event is recorded in a daily log file. The daily log is named as:

flidigYYYYMMDD.log

where YYYYMMDD is the current GMT date. This log will also contain your entire session of Rx and Tx data annotated as to activity and time stamped. Here is a small example of the daily log:

```
--- Logging started at Tue Dec 30 11:37:21 2008 UTC ---

RX (2008-12-30 11:37Z): o ur property. No pwr even for a day is rough.
TX (2008-12-30 11:39Z):
TX (2008-12-30 11:39Z): CQ CQ CQ de W1HKJ W1HKJ W1HKJ
TX (2008-12-30 11:40Z): CQ CQ CQ de W1HKJ W1HKJ W1HKJ pse k

RX (2008-12-30 11:40Z): mG sk

--- Logging stopped at Tue Dec 30 11:48:11 2008 UTC ---
```

This log is appended to with each start and stop. That means that no data is ever overwritten.

Status log

A log of errors, warnings and status reports is written for each session. This file is overwritten each time the program is opened and subsequently closed. Its format is also ASCII text and will contain data such as:

```
Q: main: fldigi 3.04BV log started on Tue Dec 30 05:47:10 2008
W: dxcc_open: Could not read contest country file "/home/dave/.fldigi/cty.dat"
```

This data is identical to that which can be viewed with the event log dialog which is opened using the menu item "Help/Event log"

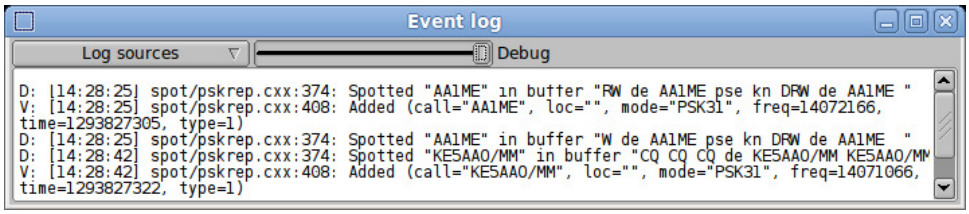


Figure 6.85: Event Log

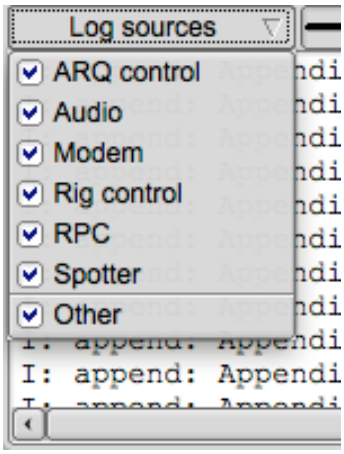


Figure 6.86: Event Log Menu

There are six levels of event logging with increasing depth of reports:

(a) Quiet

(b) Error

(c) Warning

(d) Info

(e) Verbose

(f) Debug

Figure 6.87: Event Logging Levels

The default level for logging events is "warning."

The event log show above was captured during a period of psk-reporting. Fldigi was set up to monitor and report all detected signals that satisfied the requirements of the psk reporter web site. The "spotted" signals were then

automatically sent to the web site. A complete report of the recorded events was obtained by a right click in the text pane. Select-all and Save as was chosen.

At the Debug level you will probably see more events than you need. You can select which events to suppress using the "Log sources" menu button. It defaults to all enabled.

Return to Top of Page

Return to Main Page

6.23 Supporting Data Files Acquisition

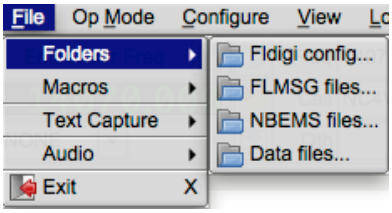


Figure 6.88: Data File Menu

To update support data file(s). Mouse click on the 'Data files..." menu item located in the above drop down menu.

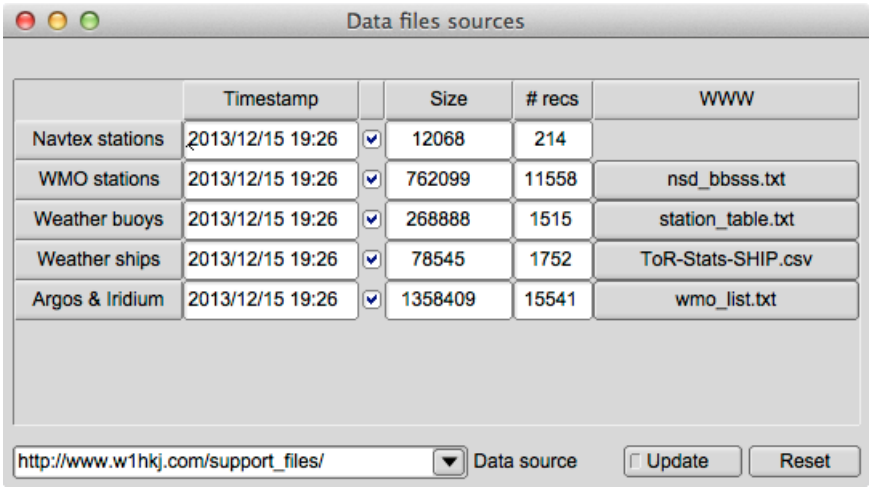


Figure 6.89: Data File Source

The dialog box contains a list of the available file(s) for downloading. Click on the 'Update' button to initiate the data transfer.

The buttons under WWW column are direct links to the data source. Selecting them either displays or downloads the data from your web browser.

Note: Data source selection contain a single entry. User selection not required.

Return to Top of Page

Return to Main Page

Chapter 7

Developers

7.1 Build Information

7.1.1 Build Info

The following is obtained by executing "fldigi --version"

```
Build information:
built :

Fri Dec 25 15:57:13 CST 2009 by dave@linux-dev on i686-pc-linux-gnu for
i686-pc-linux-gnu

 configure flags: '--without-asciidoc' '--enable-optimizations=native'

 compiler : gcc version 4.3.3 (Ubuntu 4.3.3-5ubuntu4)

 compiler flags : -I$(srcdir) -I$(srcdir)/include
-I$(srcdir)/irrxml -I$(srcdir)/fileselector -pthread
-I/usr/local/include -I/usr/local/include -I/usr/include/freetype2
-D_THREAD_SAFE -D_REENTRANT -D_REENTRANT -I/usr/local/include
-I/usr/include/libpng12 -I/usr/local/include -pipe -Wall -fexceptions
-O2 -ffast-math -finline-functions -fomit-frame-pointer -march=native
-mfpmath=sse -DNDEBUG

 linker flags : -L/usr/local/lib -lportaudio -lm
-lpthread -L/usr/local/lib -lfltk_images -lpng -lz -ljpeg -lfltk -lXft
-lpthread -ldl -lm -lXext -lX11 -lX11 -lsndfile -lsamplerate
-lpulse-simple -lpulse -L/usr/local/lib -lhamlib -lm -lpng12
-L/usr/local/lib -lxmlrpc_server_abyss++ -lxmlrpc_server++
-lxmlrpc_server_abyss -lxmlrpc_server -lxmlrpc_abyss -lpthread
```

```
-lxmlrpc++ -lxmlrpc -lxmlrpc_util -lxmlrpc_xmlparse -lxmlrpc_xmltok
-ldl -lrt -lpthread

 libraries : FLTK 1.3.2
 libsamplerate 0.1.4
 libsndfile 1.0.17
 PortAudio 19
 PulseAudio 0.9.14
 Hamlib 1.2.10
 XMLRPC-C 1.06.31

Runtime information:
system : Linux
linux-dev 2.6.28-17-generic #58-Ubuntu SMP Tue Dec 1 18:57:07 UTC 2009
i686

 libraries : libsamplerate-0.1.4 (c) 2002-2008 Erik de Castro Lopo
 libsndfile-1.0.17

PortAudio V19-devel (built May 25 2009 06:36:24) 1899
Pulseaudio 0.9.14

Hamlib version 1.2.10
```

Return to Top of Page

Return to Main Page

7.2 Installing Fldigi

7.2.1 Precompiled Binary

The precompiled binary is available with and without a dependency on PulseAudio. Unless you know that your system uses PulseAudio for its sound card service you should not download that version.

You will need three shared libraries on your system, hamlib-1.2.10; libsamplerate; and libportaudio2. Use the libraries available for your linux distribution. Most current distributions use either deb or rpm files and can be accessed from a global repository.

Building the libraries from source should be a last resort unless you are a knowledgable Linux user and have performed a library build from source in the past.

7.2.1.1 Hamlib

You will need to have hamlib-1.2.7 installed on your system before fldigi can be executed. Most current distributions either have a deb or rpm distribution file for hamlib-1.2.10. If you must compile from source you can find it at:

http://www.hamlib.org

Follow the instructions in the source code top directory to compile, link and install the library.

7.2.1.2 libsamplerate

You will need to have libsample installed on your system before fldigi can be executed. Most current distributions either have a deb or rpm distribution file for libsamplerate. If you must compile from source you can find it at:

libsamplerate source

Follow the instructions in the source code top directory to compile, link and install the library.

7.2.1.3 PortAudio

You will need to have libportaudio2 installed on your system before fldigi can be executed. Most current distributions either have a deb or rpm distribution file for libportaudio2. If you must compile from source you can find it at:

http://www.portaudio.com

Follow the instructions in the source code top directory to compile, link and install the library.

7.2.2 Installing fldigi

The static executables are tested on as many distributions as possible to insure that they work "out-of-the-box", but there are always a few Linux distributions that may have a missing link or library. The precompiled binaries have been tested and work correctly on all of the Debian and Ubuntu/Kubuntu distributions. They have also been tested and confirmed to work on Suse 10.1, and Mandriva 2007.

Download the tarball for the binary version and unpack to a directory on your HD such as $HOME/bin or some other convenient directory of your choosing. The least common denominator for unpacking a tarball is to download the file and save it to a convenient directory such as $HOME/downloads. Then open up a terminal window. Assuming you will be installing the executable to $HOME/bin do the following and that you have downloaded the tarball to $HOME/downloads

```
cd
cd bin
tar xzf ../downloads/fldigi-D.dd.npa.bin.tgz
```

where D.dd is the current version number as in 3.10

You can create a link to the fldigi executable on your desktop using the fldigi.png icon located at

http://www.w1hkj.com/fldigi-distro/fldigi-psk.png

Creating a desktop link to an application is different for each desktop manager, so please refer to the documentation for your specific manager.

The first time that you execute fldigi either from the command line or by clicking on the executable in a file manager or the desktop icon it will create a new directory and file:

- $HOME/.fldigi
- $HOME/.fldigi/macros.mdf

If this is a new installation you will be guided through some necessary configuration by a new install wizard.

Return to Top of Page

Return to Main Page

7.3 Installing from Source

The developers recommend that you build either fldigi from source or install from the repository associated with your distribution. The repository may not be immediately available for the most current version number. In that case you can try installing the binary distribution. Keep in mind that the version numbers of the dependent shared libraries must match those on the machine used to create the binary.

The source code for fldigi is very large and has a number of dependencies that must be satisfied before a successful compile. If you are not familiar with compiling and linking source code you should probably practice on a simpler package before treading these waters. Please refer to the following web site for information on building for Linux, Windows and OS X.

Fldigi WIKI - build instructions

Return to Top of Page

Return to Main Page

7.4 Parse UALR

A simple parser to create a formated console output for fldigi's <EXEC> macro:

```
snip-------------------------------------------
#include <ctime>
#include <cstdio>
#include <cstdlib>
#include <unistd.h>
#include <string>
#include <iostream>
#include <fstream>

using namespace std;
using std::cout;
using std::cin;

int main(int argc, char *argv[])
{
    size_t pos = 0, pos2 = 0, pos3 = 0, pos4 = 0, pos5 = 0;
    string commandline = "";
    string name = "";
    string qth = "";
    string answer = "";
    char c = cin.get();

    while (!cin.eof()) {
        commandline += c;
        c = cin.get();
    }

    if (commandline.find("No match found") != string::npos)
        goto noresponse;

    pos = commandline.find(", ");

    if (pos == string::npos)
        goto noresponse;

    pos += 2;
    pos2 = commandline.find("\n", pos);

    if (pos2 == string::npos)
        goto noresponse;

    name = commandline.substr(pos, pos2 - pos);
    pos3 = name.find(32);

    if (pos3 != string::npos)
        name = name.substr(0, pos3);

    for (size_t i = 1; i \< name.length(); i++)
        name[i] = tolower(name[i]);

    answer = "$NAME";
    answer.append(name);

    pos4 = commandline.find(", ", pos2);
    pos4 = commandline.rfind( "\n", pos4);
    pos4 += 1;
    pos5 = commandline.find("\n", pos4);

    qth = commandline.substr(pos4, pos5 - pos4);

    answer.append("$QTH");
    answer.append(qth);
```

```
    cout \<\< answer.c_str();

    return 0;

noresponse:;

    cout \<\< "$NAME?$QTH?";

    return 0;
}
snip-----------------------------------------------
```

Save the above as "parseUALR.cxx" and then compile and link as follows:

```
g++ parseUALR.cxx -o parseUALR
```

Copy the "parseUALR" executable to a directory on your shell exec PATH.

Return to Top of Page

Return to Main Page

7.5 Pseudo FSK

Using the FLdigi Pseudo FSK (Rt. Channel) function to key a transmitter

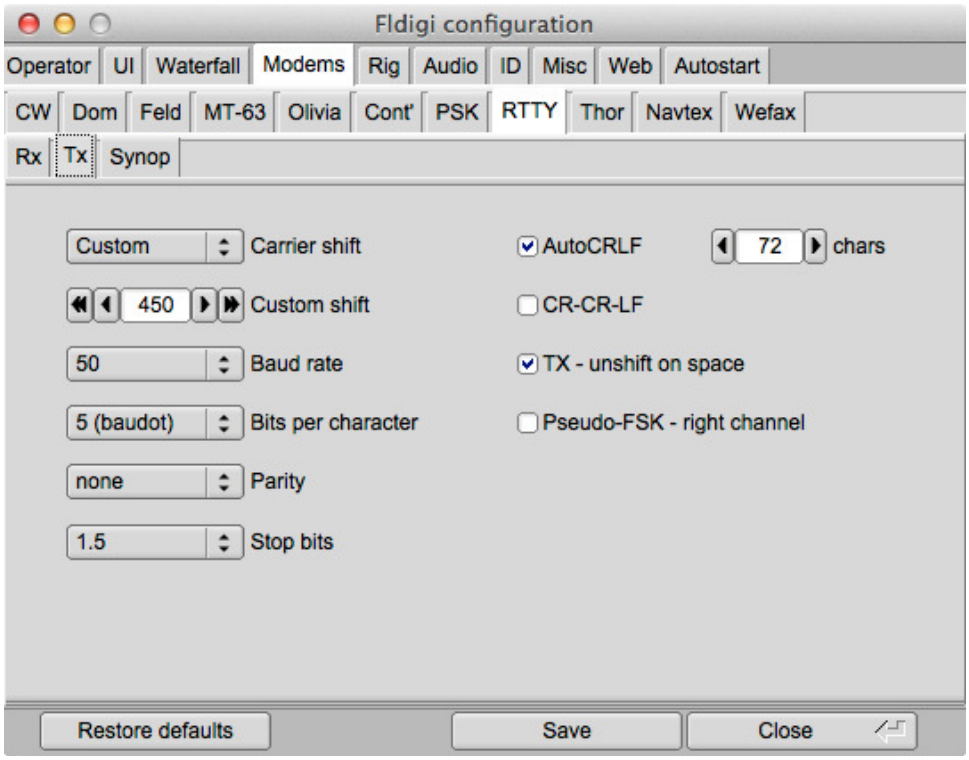


Figure 7.1: RTTY TX Configure

Select the PseudoFSK check boxes.

FLdigi is now ready to generate a 1000 hertz tone burst signal on the right channel of the stereo audio out of your sound card.

This tone burst is on when the RTTY bit is on and off when the RTTY bit is off. The left channel will be the normal AFSK signal.

The following circuit may be used to take the FLdigi PSEUDO-FSK signal from the right channel of your SOUND CARD to key your transmitter's FSK input line. You may find it necessary to invert the sense of the keying signal.

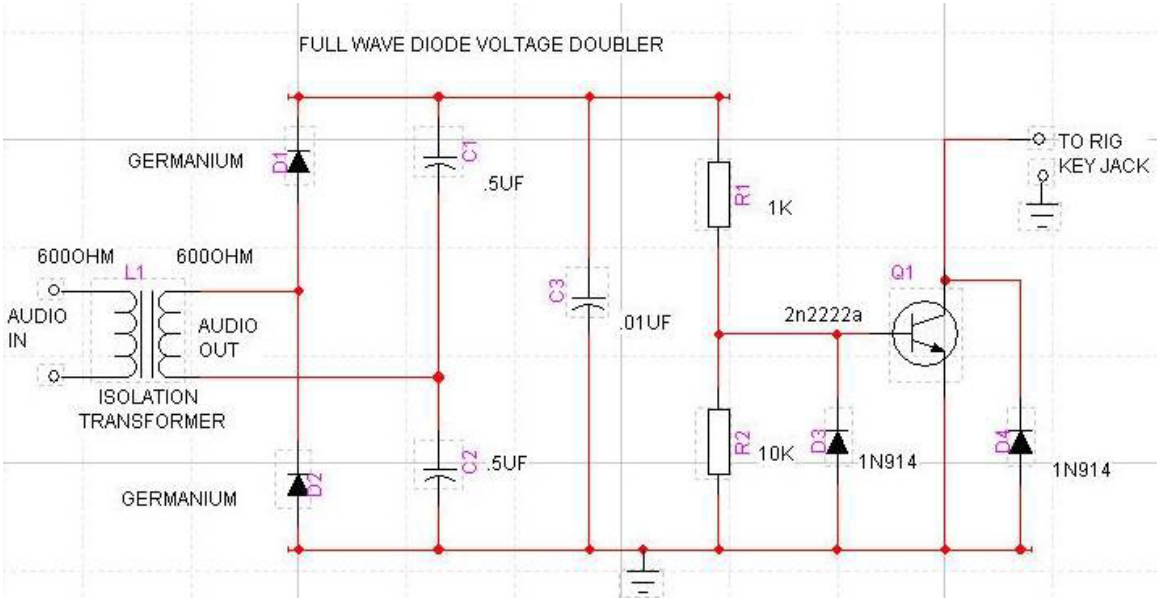


Figure 7.2: CW Keying Circuit

NOTE:

L1 - Radio Shack has two items that may be used for this isolation transformer.

`Catalog # 270-054, and Catalog # 273-1374`

Attach an audio cable from the Rt. Channel out of the your computer's SOUND CARD to the input of this FSK INTERFACE CIRCUIT (input of L1).

Attach another cable from the output of this circuit to your Rig's Keying FSK Jack.

Every PSEUDO-FSK tone that is generated by FLdigi is rectified by this FULL WAVE VOLTAGE DOUBLER circuit. The resultant voltage turns the Q1 transistor on and "grounds" the collector.

Return to Top of Page

Return to Main Page

7.6 Rig Xml How to

This document describes the contents of the rig definition file "rig.xml".

A number of transceivers have rig definition files written and tested which you may use. These are found in the xmls directory on this site: `xml archives`. You will find subdirectories by manufacturer which contain files named by rig type, ie: TS-850.xml. If you create, test and verify the proper operation for a transceiver not yet posted please share that with others by sending it as an attachment to `w1hkj@w1hkj.com` and I will post it on the web site. You are encouraged to study the various rig definition files to learn more about how they are organized.

Comments are contained within the tag pair:

<!-

–>

and may appear anywhere in the rig definition file The entire rig definition must be contained within the tag pair

<RIGDEF>

</RIGDEF>

The text within the tag pair <RIG></RIG> specifies the transceiver to which this file applies, as in:

<RIG>Icom 746 PRO</RIG>

The text within the tag pair <PROGRAMMER></PROGRAMMER> is not used by the parser, but should as a minimum say who created and who tested the definition file, as in:

<PROGRAMMER>

Dave Freese W1HKJ Tested by: W1HKJ, Dave

</PROGRAMMER>

The text within the tag pair

<STATUS>

</STATUS>

is not used by the parser, but should as a minimum state whether the definition file has been "Verified", is "Alpha&", what the Version and Date of creation or update, as in:

<STATUS> Verified Version: 1.0 Date: 2007 Jan 5 </STATUS>

The <TITLE>

</TITLE>

tag pair contains the text which will be displayed on the window decoration bar, as in:

<TITLE>Rig Control - IC-746 PRO</TITLE>

The serial port parameters may be preset in the xml file and also set or changed on the rigcat configuration tab.

<!–

default settings for initial setup

–>

xml tag	parameter
<TIMEOUT>TT</TIMEOUT>	TT in milliseconds
<RETRIES>NN</RETRIES>	NN integer
<WRITE_DELAY>TT</WRITE_DELAY>	TT in milliseconds
<POST_WRITE_DELAY>TT</POST_WRITE_DELAY>	TT in milliseconds
<BAUDRATE>BAUD</BAUDRATE>	BAUD = 1200, 2400, 4800, 9600, 19200, 38400 ...
<STOPBITS>B</STOPBITS>	B = 1 or 2
<RTSCTS>BOOL</RTSCTS>	BOOL = true, false; h/w handshake used for data flow control
<RTSPLUS>BOOL</RTSPLUS>	BOOL = true, false; set RTS signal line to +12 V
<RTSPTT>BOOL</RTSPTT>	BOOL = true, false; toggle RTS signal line for PTT
<DTRPLUS>BOOL</DTRPLUS>	BOOL = true, false; set DTR signal line to + 12 V
<DTRPTT>BOOL</DTRPTT>	BOOL = true, false; toggle DTR signal line for PTT

<ECHO>BOOL</ECHO>	BOOL = true, false; xcvr/interface echoes all chars (typical of CI-V interface)
<CMDPTT>BOOL</CMDPTT>	BOOL = true, false; use command string for PTT (not supported by all rigs)

The transceiver modes are specified within the <MODES></MODES> tag pair. Each entry or element associated with a mode has a symbol name (text) and a way to specify what the data transfer consists of. The data transfer might be a single byte, multiple bytes, or aa string

Example 1, for the Icom-746PRO

<MODES>

<ELEMENT><SYMBOL>LSB</SYMBOL><BYTE>00</BYTE></ELEMENT>

<ELEMENT><SYMBOL>USB</SYMBOL><BYTE>01</BYTE></ELEMENT>

<ELEMENT><SYMBOL>AM</SYMBOL><BYTE>02</BYTE></ELEMENT>

<ELEMENT><SYMBOL>CW</SYMBOL><BYTE>03</BYTE></ELEMENT>

<ELEMENT><SYMBOL>RTTY</SYMBOL><BYTE>04</BYTE></ELEMENT>

<ELEMENT><SYMBOL>FM</SYMBOL><BYTE>05</BYTE></ELEMENT>

<ELEMENT><SYMBOL>CW-R</SYMBOL><BYTE>07</BYTE></ELEMENT>

<ELEMENT><SYMBOL>RTTY-R</SYMBOL><BYTE>08</BYTE></ELEMENT>

</MODES>

Example 2, for the Kenwood 850

<MODES>

<ELEMENT><SYMBOL>LSB</SYMBOL><BYTE>31</BYTE></ELEMENT>

<ELEMENT><SYMBOL>USB</SYMBOL><BYTE>32</BYTE></ELEMENT>

<ELEMENT><SYMBOL>CW</SYMBOL><BYTE>33</BYTE></ELEMENT>

<ELEMENT><SYMBOL>FM</SYMBOL><BYTE>34</BYTE></ELEMENT>

<ELEMENT><SYMBOL>AM</SYMBOL><BYTE>35</BYTE></ELEMENT>

<ELEMENT><SYMBOL>FSK</SYMBOL><BYTE>36</BYTE></ELEMENT>

<ELEMENT><SYMBOL>CW-R</SYMBOL><BYTE>37</BYTE></ELEMENT>

<ELEMENT><SYMBOL>FSK-R</SYMBOL><BYTE>39</BYTE></ELEMENT>

</MODES>

Example 3, for the FT-100

<MODES>

<ELEMENT><SYMBOL>LSB</SYMBOL><BYTE>00</BYTE></ELEMENT>

<ELEMENT><SYMBOL>USB</SYMBOL><BYTE>01</BYTE></ELEMENT>

<ELEMENT><SYMBOL>CW</SYMBOL><BYTE>02</BYTE></ELEMENT>

<ELEMENT><SYMBOL>CW-R</SYMBOL><BYTE>03</BYTE></ELEMENT>

<ELEMENT><SYMBOL>AM</SYMBOL><BYTE>04</BYTE></ELEMENT>

<ELEMENT><SYMBOL>DIG</SYMBOL><BYTE>05</BYTE></ELEMENT>

<ELEMENT><SYMBOL>FM</SYMBOL><BYTE>06</BYTE></ELEMENT>

<ELEMENT><SYMBOL>W-FM</SYMBOL><BYTE>07</BYTE></ELEMENT>

</MODES>

The modes which are supported by lower sideband in the transceiver are specified in the <LSBMODES></LSBMODES> tar pair. The string data for the LSB modes must match those given in the modes id specifier For example in the Icom 746 Pro:

<LSBMODES>

<STRING>LSB</STRING>

<STRING>RTTY</STRING>

<STRING>CW-R</STRING>

</LSBMODES>

If the transceiver data stream uses identically the same format for the bandwidth data then it is specified in the <BANDWIDTHS></BANDWIDTHS> tag pair

Example for the Icom 746 Pro:

<BANDWIDTHS>

<ELEMENT><SYMBOL>50</SYMBOL><BYTE>00</BYTE></ELEMENT>

<ELEMENT><SYMBOL>100</SYMBOL><BYTE>01</BYTE></ELEMENT>

<ELEMENT><SYMBOL>150</SYMBOL><BYTE>02</BYTE></ELEMENT>

<ELEMENT><SYMBOL>200</SYMBOL><BYTE>03</BYTE></ELEMENT>

<ELEMENT><SYMBOL>250</SYMBOL><BYTE>04</BYTE></ELEMENT>

<ELEMENT><SYMBOL>300</SYMBOL><BYTE>05</BYTE></ELEMENT>

<ELEMENT><SYMBOL>350</SYMBOL><BYTE>06</BYTE></ELEMENT>

<ELEMENT><SYMBOL>400</SYMBOL><BYTE>07</BYTE></ELEMENT>

<ELEMENT><SYMBOL>450</SYMBOL><BYTE>08</BYTE></ELEMENT>

<ELEMENT><SYMBOL>500</SYMBOL><BYTE>09</BYTE></ELEMENT>

<ELEMENT><SYMBOL>600</SYMBOL><BYTE>10</BYTE></ELEMENT>

<ELEMENT><SYMBOL>700</SYMBOL><BYTE>11</BYTE></ELEMENT>

<ELEMENT><SYMBOL>800</SYMBOL><BYTE>12</BYTE></ELEMENT>

<ELEMENT><SYMBOL>900</SYMBOL><BYTE>13</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1000</SYMBOL><BYTE>14</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1100</SYMBOL><BYTE>15</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1200</SYMBOL><BYTE>16</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1300</SYMBOL><BYTE>17</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1400</SYMBOL><BYTE>18</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1500</SYMBOL><BYTE>19</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1600</SYMBOL><BYTE>20</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1700</SYMBOL><BYTE>21</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1800</SYMBOL><BYTE>22</BYTE></ELEMENT>

<ELEMENT><SYMBOL>1900</SYMBOL><BYTE>23</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2000</SYMBOL><BYTE>24</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2100</SYMBOL><BYTE>25</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2200</SYMBOL><BYTE>26</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2300</SYMBOL><BYTE>27</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2400</SYMBOL><BYTE>28</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2500</SYMBOL><BYTE>29</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2600</SYMBOL><BYTE>30</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2700</SYMBOL><BYTE>31</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2800</SYMBOL><BYTE>32</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2900</SYMBOL><BYTE>33</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3000</SYMBOL><BYTE>34</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3100</SYMBOL><BYTE>35</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3200</SYMBOL><BYTE>36</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3300</SYMBOL><BYTE>37</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3400</SYMBOL><BYTE>38</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3500</SYMBOL><BYTE>39</BYTE></ELEMENT>

<ELEMENT><SYMBOL>3600</SYMBOL><BYTE>40</BYTE></ELEMENT>

</BANDWIDTHS>

If the bandwidth data stream is unique for send and receive data streams then they are specified separately with the <BW-CMD></BW-CMD> tag pair for data sent to the transceiver, and the <BW-REPLY></BW-REPLY> tag pair for data returned to the computer.

Example: FT-100:

<BW-CMD>

<ELEMENT><SYMBOL>300</SYMBOL><BYTE>00</BYTE></ELEMENT>

<ELEMENT><SYMBOL>500</SYMBOL><BYTE>01</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2400</SYMBOL><BYTE>02</BYTE></ELEMENT>

<ELEMENT><SYMBOL>6000</SYMBOL><BYTE>03</BYTE></ELEMENT>

</BW-CMD>

<BW-REPLY>

<ELEMENT><SYMBOL>300</SYMBOL><BYTE>03</BYTE></ELEMENT>

<ELEMENT><SYMBOL>500</SYMBOL><BYTE>02</BYTE></ELEMENT>

<ELEMENT><SYMBOL>2400</SYMBOL><BYTE>01</BYTE></ELEMENT>

<ELEMENT><SYMBOL>6000</SYMBOL><BYTE>00</BYTE></ELEMENT>

</BW-REPLY>

Fldigi can parse and decode message returned from the transceiver that define 4 aspects of the transceiver operation:

OK	data accepted by the transceiver
BAD	data rejected by the transceiver
MODE	current operating mode of the transceiver
BW	current bandwidth setting of the transceiver
FREQ	frequency of the active VFO (might be either A or B for example)

These are all contained within multiple <REPLY></REPLY> tag pairs. This is an example of a fixed format message with no variable fields. It is the OK message sent back by the Icom-746 PRO:

<REPLY>

<SYMBOL>OK</SYMBOL>

<SIZE>6</SIZE>

<BYTES>FE FE E0 66</BYTES>

<BYTE>FB</BYTE>

<BYTE>FD</BYTE>

</REPLY>

The <SYMBOL></SYMBOL> pair and the command definition are mandatory. The <SIZE></SIZE> field is mandatory and specifies the number of bytes contained in this reply. The above definition could also have been coded as:

<REPLY>

<SYMBOL>OK</SYMBOL>

<SIZE>6</SIZE>

<BYTES>FE FE E0 66 FB FD</BYTES>

</REPLY>

When the reply contains variable data it is specified in a contained tag pair <DATA></DATA>. This data field contains specifiers that describe the kind and size of the data. The <DTYPE></DTYPE> tag pair may be one of:

BINARY or

DECIMAL

This is an example for the reply to a mode query that is returned by the Icom-746 PRO:

<REPLY>

<SYMBOL>MODE</SYMBOL> specifies the response name

<SIZE>8</SIZE> 8 bytes of data returned

<BYTES>FE FE E0 66</BYTES> 4 bytes of preamble

<BYTE>04</BYTE> 1 additional byte for preample

<DATA>

<DTYPE>BINARY</DTYPE> binary data field of 1 byte

<SIZE>1</SIZE>

</DATA>

<FILL>1</FILL> a variable field (data) not used

<BYTE>FD</BYTE> 1 byte postamble

</REPLY>

Fldigi rigcat will check for both the preample and postamble to insure that a valid reply has been sent by the transceiver.

Return to Top of Page

Return to Main Page

7.7 ualr telnet

```
snip ------------ copy the following to ~/.fldigi/scripts/ualr-telnet.pl

#!/usr/bin/perl

# Author: Stelios Bounanos, M0GLD
# Date: 20090103
#
# ualr-telnet is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 3 of the License, or
# (at your option) any later version.
```

```
#
# ualr-telnetl is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program.  If not, see <http://www.gnu.org/licenses/>.
# ------------------------------------------------------------------------

use strict;
use warnings;

die "Usage: $0 CALLSIGN\n" unless (@ARGV == 1);

use Net::Telnet ();

sub error { print "\$NAME?\$QTH?\n"; exit(1); }
my $t = new Net::Telnet( Host => "callsign.ualr.edu", Port => 2000, Timeout => 10,
            errmode => \&error );
$t->open();
$t->waitfor('/LOOKUP>.*$/');
$t->print($ARGV[0]);

$_ = $t->getline();      # blank line
$_ = $t->getline();      # call
error() if (m/No match found/);

$_ = $t->getline();      # name
chomp; s/.+,\s+//; s/\s.+$//;
print "\$NAME$_";
$_ = $t->getline();      # addr
$_ = $t->getline();      # qth
chomp;
$_ =~ ",";
$_ = $`;

print "\$QTH$_\n";

$t->waitfor('/LOOKUP>.*$/');
$t->print("bye");

snip------------------------------------------------
```

Return to Top of Page

Return to Main Page

7.8 Xmlrpc Control

XML-RPC data is transported via simple HTTP and client implementations exist for most programming languages. A Perl client that can be used as a control script is included in the source tar ball as scripts/fldigi-shell. This control method is currently used by several external programs including flrig, logger32 and Xlog.

The following command line arguments become available when XML-RPC support is compiled into fldigi, as described in the build instructions:

```
 --xmlrpc-server-address HOSTNAME
   Set the XML-RPC server address. The default is 127.0.0.1.

 --xmlrpc-server-port PORT
   Set the XML-RPC server port. The default is 7362.

 --xmlrpc-allow REGEX
   Allow only the methods whose names match REGEX

 --xmlrpc-deny REGEX
   Allow only the methods whose names don't match REGEX
```

```
--xmlrpc-list
  List all available methods
```

The –xmlrpc-deny and –xmlrpc-allow switches can be used as a simple access control mechanism. REGEX specifies a POSIX extended regular expression. This invocation disables the methods that may cause fldigi to transmit:

```
--xmlrpc-deny 'main\.(tx|tune|run_macro)'
```

By default all methods are allowed.

The –xmlrpc-list switch outputs the method list and exits the program. If preceded by –xmlrpc-deny or –xmlrpc-allow, it shows the list of methods as filtered by those switches.

The methods are listed below. The three columns are method name, signature (return_type:argument_types), and description. Refer to the XML-RPC specification for the meaning of the signature characters

7.8.1 XML Command Symbol Interpretation

Symbol	Interpretation
n	nil
b	boolean
i	integer
d	double
s	string
6	bytes
A	array
S	struct

7.8.2 Table of XML Commands

Method Name	Sig (ret:arg)	Description
fldigi.list	A:n	Returns the list of methods
fldigi.name	s:n	Returns the program name
fldigi.version_struct	S:n	Returns the program version as a struct
fldigi.version	s:n	Returns the program version as a string
fldigi.name_version	s:n	Returns the program name and version
fldigi.config_dir	s:n	Returns the name of the configuration directory
fldigi.terminate	n:i	Terminates fldigi. “i” is bitmask specifying data to save: 0=options; 1=log; 2=macros
modem.get_name	s:n	Returns the name of the current modem
modem.get_names	A:n	Returns all modem names
modem.get_id	i:n	Returns the ID of the current modem
modem.get_max_id	i:n	Returns the maximum modem ID number
modem.set_by_name	s:s	Sets the current modem. Returns old name
modem.set_by_id	i:i	Sets the current modem. Returns old ID
modem.set_carrier	i:i	Sets modem carrier. Returns old carrier
modem.inc_carrier	i:i	Increments the modem carrier frequency. Returns the new carrier
modem.get_carrier	i:n	Returns the modem carrier frequency
modem.get_afc_search_range	i:n	Returns the modem AFC search range
modem.set_afc_search_range	n:i	Sets the modem AFC search range. Returns the old value
modem.inc_afc_search_range	n:i	Increments the modem AFC search range. Returns the new value

modem.get_bandwidth	i:n	Returns the modem bandwidth
modem.set_bandwidth	n:i	Sets the modem bandwidth. Returns the old value
modem.inc_bandwidth	n:i	Increments the modem bandwidth. Returns the new value
modem.get_quality	d:n	Returns the modem signal quality in the range [0:100]
modem.search_up	n:n	Searches upward in frequency
modem.search_down	n:n	Searches downward in frequency
modem.olivia.set_bandwidth	n:i	Sets the Olivia bandwidth
modem.olivia.get_bandwidth	i:n	Returns the Olivia bandwidth
modem.olivia.set_tones	n:i	Sets the Olivia tones
modem.olivia.get_tones	i:n	Returns the Olivia tones
main.get_status1	s:n	Returns the contents of the first status field (typically s/n)
main.get_status2	s:n	Returns the contents of the second status field
main.get_wf_sideband	s:n	Returns the current waterfall sideband
main.set_wf_sideband	n:s	Sets the waterfall sideband to USB or LSB
main.get_frequency	d:n	Returns the RF carrier frequency
main.set_frequency	d:d	Sets the RF carrier frequency. Returns the old value
main.inc_frequency	d:d	Increments the RF carrier frequency. Returns the new value
main.get_afc	b:n	Returns the AFC state
main.set_afc	b:b	Sets the AFC state. Returns the old state
main.toggle_afc	b:n	Toggles the AFC state. Returns the new state
main.get_squelch	b:n	Returns the squelch state
main.set_squelch	b:b	Sets the squelch state. Returns the old state
main.toggle_squelch	b:n	Toggles the squelch state. Returns the new state
main.get_squelch_level	d:n	Returns the squelch level
main.set_squelch_level	d:d	Sets the squelch level. Returns the old level
main.inc_squelch_level	d:d	Increments the squelch level. Returns the new level
main.get_reverse	b:n	Returns the Reverse Sideband state
main.set_reverse	b:b	Sets the Reverse Sideband state. Returns the old state
main.toggle_reverse	b:n	Toggles the Reverse Sideband state. Returns the new state
main.get_lock	b:n	Returns the Transmit Lock state
main.set_lock	b:b	Sets the Transmit Lock state. Returns the old state
main.toggle_lock	b:n	Toggles the Transmit Lock state. Returns the new state

main.get_rsid	b:n	Returns the RSID state
main.set_rsid	b:b	Sets the RSID state. Returns the old state
main.toggle_rsid	b:n	Toggles the RSID state. Returns the new state
main.get_trx_status	s:n	Returns transmit/tune/receive status
main.rx	n:n	Receives
main.tx	n:n	Transmits
main.tune	n:n	Tunes
main.abort	n:n	Aborts a transmit or tune
main.get_trx_state	s:n	Returns T/R state
main.run_macro	n:i	Runs a macro
main.get_max_macro_id	i:n	Returns the maximum macro ID number
rig.set_name	n:s	Sets the rig name for xmlrpc rig
rig.get_name	s:n	Returns the rig name previously set via rig.set_name
rig.set_frequency	d:d	Sets the RF carrier frequency. Returns the old value
rig.set_modes	n:A	Sets the list of available rig modes
rig.set_mode	n:s	Selects a mode previously added by rig.set_modes
rig.get_modes	A:n	Returns the list of available rig modes
rig.get_mode	s:n	Returns the name of the current transceiver mode
rig.set_bandwidths	n:A	Sets the list of available rig bandwidths
rig.set_bandwidth	n:s	Selects a bandwidth previously added by rig.set_bandwidths
rig.get_bandwidth	s:n	Returns the name of the current transceiver bandwidth
rig.get_bandwidths	A:n	Returns the list of available rig bandwidths
rig.take_control	n:n	Switches rig control to XML-RPC
rig.release_control	n:n	Switches rig control to previous setting
log.get_frequency	s:n	Returns the Frequency field contents
log.get_time_on	s:n	Returns the Time-On field contents
log.get_time_off	s:n	Returns the Time-Off field contents
log.get_call	s:n	Returns the Call field contents
log.get_name	s:n	Returns the Name field contents
log.get_rst_in	s:n	Returns the RST(r) field contents
log.get_rst_out	s:n	Returns the RST(s) field contents
log.get_serial_number	s:n	Returns the serial number field contents
log.set_serial_number	n:s	Sets the serial number field contents

log.get_serial_number_sent	s:n	Returns the serial number (sent) field contents
log.get_exchange	s:n	Returns the contest exchange field contents
log.set_exchange	n:s	Sets the contest exchange field contents
log.get_state	s:n	Returns the State field contents
log.get_province	s:n	Returns the Province field contents
log.get_country	s:n	Returns the Country field contents
log.get_qth	s:n	Returns the QTH field contents
log.get_band	s:n	Returns the current band name
log.get_notes	s:n	Returns the Notes field contents
log.get_locator	s:n	Returns the Locator field contents
log.get_az	s:n	Returns the AZ field contents
log.clear	n:n	Clears the contents of the log fields
log.set_call	n:s	Sets the Call field contents
log.set_name	n:s	Sets the Name field contents
log.set_qth	n:s	Sets the QTH field contents
log.set_locator	n:s	Sets the Locator field contents
text.get_rx_length	i:n	Returns the number of characters in the RX widget
text.get_rx	6:ii	Returns a range of characters (start, length) from the RX text widget
text.clear_rx	n:n	Clears the RX text widget
text.add_tx	n:s	Adds a string to the TX text widget
text.add_tx_bytes	n:6	Adds a byte string to the TX text widget
text.clear_tx	n:n	Clears the TX text widget
spot.get_auto	b:n	Returns the autospotter state
spot.set_auto	n:b	Sets the autospotter state. Returns the old state
spot.toggle_auto	n:b	Toggles the autospotter state. Returns the new state
spot.pskrep.get_count	i:n	Returns the number of callsigns spotted in the current session

Deprecated methods:

Method Name	Sig	Resolution
main.get_sideband	s:n	use main.get_wf_sideband and/or rig.get_mode
main.set_sideband	n:s	use main.set_wf_sideband and/or rig.set_mode
main.rsid	n:n	use main.{get,set,toggle}_rsid
main.set_rig_name	n:s	use rig.set_name
main.set_rig_frequency	d:d	use rig.set_frequency
main.set_rig_modes	n:A	use rig.set_modes
main.set_rig_mode	n:s	use rig.set_mode
main.get_rig_modes	A:n	use rig.get_modes
main.get_rig_mode	s:n	use rig.get_mode
main.set_rig_bandwidths	n:A	use rig.set_bandwidths
main.set_rig_bandwidth	n:s	use rig.set_bandwidth

main.get_rig_bandwidth	s:n	use rig.get_bandwidth
main.get_rig_bandwidths	n:A	use rig.get_bandwidths
log.get_sideband	s:n	use main.get_wf_sideband

7.8.3 Minimized WF Window

If your external control program duplicates some of the fldigi controls such as the Rx and Tx pane you can run fldigi in a fully minimized mode. Fldigi then only provides the controls necessary for signal acquisition and waterfall management. Minimization is accomplished by setting the command line switch;

```
--wfall-only, or --wo.
```

The user interface then has this appearance:

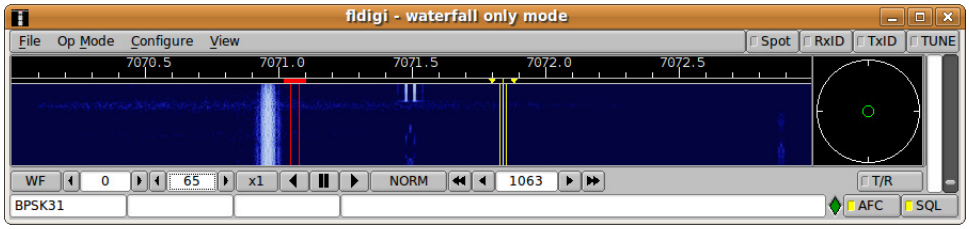


Figure 7.3: Simplified WF Window Display

The documentation for the external control program will provide additional information if this user interface is desired.

Return to Top of Page

Return to Main Page

7.9 Google Maps

```
snip ------------ copy the following to ~/.fldigi/scripts/map.pl

#!/usr/bin/perl

# Author: Stelios Bounanos, M0GLD
# Date:   20080625

use warnings;
use strict;
use Getopt::Std;

our $VERSION = "0.3141";
our %opts = ( "e" => 0, "m" => 1, "z" => 4);

cmdline();
open(STDOUT, '>', "/dev/null");

my $loc = exists($opts{'l'}) ? $opts{'l'} : $ENV{'FLDIGI_LOG_LOCATOR'};
die "Invalid locator\n" unless ((defined($loc) && length($loc) =~ /[2-6]/));

my $label = exists($opts{'t'}) ? $opts{'t'} : $ENV{'FLDIGI_LOG_CALL'};
$label = $loc if (!defined($label) || $label eq "");

my ($lon, $lat) = map { sprintf("%+.6f", $_) } mtoll($loc);
if ($opts{'m'}) {
    my $url = "http://maps.google.com/maps?q=${lat},${lon}(${label})&t=p&z=$opts{'z'}";
#    $url =~ s/([(),])/sprintf("%%%02X", ord($1))/ge; # encode some chars
    exec("xdg-open", $url);
    die "Could not exec xdg-open: $!\n";
}
```

```
exit(0) unless ($opts{'e'});
my $kml = (exists($ENV{'TMPDIR'}) ? $ENV{'TMPDIR'} : "/tmp") .
    "/" . $loc . ".kml";
open(KML, '>', $kml) or die "Could not write $kml: $!\n";
print KML <<EOF
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.2">
  <Placemark>
    <name>$label</name>
    <description>
      $label
      $loc
    </description>
    <Point>
      <coordinates>$lon,$lat,0</coordinates>
    </Point>
  </Placemark>
</kml>
EOF
    ;
close(KML);

######################################################################

sub cmdline
{
    $Getopt::Std::STANDARD_HELP_VERSION = 1;
    my $old_warn_handler = $SIG{__WARN__};
    $SIG{__WARN__} = sub { die $_[0]; };
    getopts('t:l:mz:e', \%opts);
    $SIG{__WARN__} = $old_warn_handler;
}


# Convert a 2, 4, or 6-character Maidenhead locator string
# to decimal degrees. Return a (longitude, latitude) pair.
sub mtoll
{
    my $len = length($_[0]);
    $_[0] .= join("", ("A", "A", "0", "0", "A", "A")[$len .. 5]) if ($len < 6);
    $_[0] = uc($_[0]);
    die "Invalid locator\n" unless ($_[0] =~ /[A-R]{2}\d{2}[A-X]{2}/);

    my @digits = split(//, $_[0]);
    my ($lon, $lat) = (-180, -90);

    $lon += (ord($digits[0]) - ord('A')) * 20 +
            (ord($digits[2]) - ord('0')) * 2 +
            (ord($digits[4]) - ord('A') + 0.5) / 12;
    $lat += (ord($digits[1]) - ord('A')) * 10 +
            (ord($digits[3]) - ord('0')) +
            (ord($digits[5]) - ord('A') + 0.5) / 24;

    return ($lon, $lat);
}

sub HELP_MESSAGE
{
print <<EOF

Usage: $0 [-OPTIONS [-MORE_OPTIONS]] [--] [PROGRAM_ARG1 ...]

The following single-character options are accepted:

        -t LABEL  Use LABEL as the marker label
                  The default is \$FLDIGI_LOG_CALL

        -l LOC    Place marker at IARU locator LOC
                  The default is \$FLDIGI_LOG_LOCATOR

        -m        Show in Google Maps (default)
        -z        Zoom level (Google Maps only)
```

```
        -e        Write a Google Earth kml file in
                  \$TMPDIR/LOC.kml
EOF
;
}

snip------------------------------------------------
```

Return to Top of Page

Return to Main Page

Chapter 8

Command Line Switches

```
Usage:
    fldigi [option...]

fldigi options:

  --home-dir DIRECTORY
    Set the home directory to full pathname of DIRECTORY
    fldigi will put the file stores
      .fldigi.files, and .nbems.files
    in this directory
    The default is: /Users/robert/

  --config-dir DIRECTORY
    Look for configuration files in DIRECTORY
    The default is: /Users/robert/.fldigi/

  --arq-server-address HOSTNAME
    Set the ARQ TCP server address
    The default is: 127.0.0.1

  --arq-server-port PORT
    Set the ARQ TCP server port
    The default is: 7322

  --flmsg-dir DIRECTORY
    Look for flmsg files in DIRECTORY
    The default is /Users/robert/.nbems/

  --auto-dir DIRECTORY
    Look for wrap_auto_file files in DIRECTORY
    The default is /Users/robert/.nbems/WRAP/auto/

  --xmlrpc-server-address HOSTNAME
    Set the XML-RPC server address
    The default is: 127.0.0.1

  --xmlrpc-server-port PORT
    Set the XML-RPC server port
    The default is: 7362

  --xmlrpc-allow REGEX
    Allow only the methods whose names match REGEX

  --xmlrpc-deny REGEX
    Allow only the methods whose names don't match REGEX

  --xmlrpc-list
    List all available methods

  --cpu-speed-test
    Perform the CPU speed test, show results in the event log
    and possibly change options.
```

```
  --noise
    Unhide controls for noise tests

  --wfall-only
    Hide all controls but the waterfall

  --debug-level LEVEL
    Set the event log verbosity

  --debug-pskmail
    Enable logging for pskmail / arq events

  --debug-audio
    Enable logging for sound-card events

  --version
    Print version information

  --build-info
    Print build information

  --help
    Print this option help

Standard FLTK options:

   -bg COLOR, -background COLOR
    Set the background color
   -bg2 COLOR, -background2 COLOR
    Set the secondary (text) background color

   -di DISPLAY, -display DISPLAY
    Set the X display to use DISPLAY,
    format is ``host:n.n''

   -dn, -dnd or -nodn, -nodnd
    Enable or disable drag and drop copy and paste in text fields

   -fg COLOR, -foreground COLOR
    Set the foreground color

   -g GEOMETRY, -geometry GEOMETRY
    Set the initial window size and position
    GEOMETRY format is ``WxH+X+Y''
    ** fldigi may override this setting **

   -i, -iconic
    Start fldigi in iconified state

   -k, -kbd or -nok, -nokbd
    Enable or disable visible keyboard focus in non-text widgets

   -na CLASSNAME, -name CLASSNAME
    Set the window class to CLASSNAME

   -ti WINDOWTITLE, -title WINDOWTITLE
    Set the window title

Additional UI options:

  --font FONT[:SIZE]
    Set the widget font and (optionally) size
    The default is: Arial:12
```

Return to Top of Page

Return to Main Page

Chapter 9

License(s)

9.1 Copyright

Return to Top of Page

Return to Main Page

Chapter 10

Recognitions

This software would not have been possible without the contribution of many programmers who have given their best to the open source community. The application is built upon the foundation of the Fast Light Tool Kit (http://www.fltk.org), a wonderfully fast and efficient graphical user interface design library. Many have asked what the Fast Light means. There are probably as many answers as there are programmers using the toolkit. I prefer to think of it as lightning fast and light on the code size. Take a look at the size of the executable for fldigi and then compare it with similar applications. I think you will be surprised by how small it is for what it does.

The active current development team consists of:

- Dave Freese - W1HKJ
- Stelios Bounanos - M0GLD
- Remi Chateauneu - F4ECW
- Leigh Klotz - WA5ZNU
- Stephane Fillod - F8CFE
- John Douyere - VK2ETA
- Joe Veldhuis - N8FQ
- Chris Sylvain - KB3CS
- Gary Robinson - WB8ROL
- Stefan Fendt - DO2SMF
- John Phelps - KL4YFD
- Andrej Lajovic - S57LN
- Robert Stiles - KK5VD

Localization files:

- French: Stephane Fillod - F8CFE
- German: Marc Richter - DF2MR
- Italian: Pierfrancesco Caci - IK5PVX
- Spanish: Pavel Milanes Costa - CO7WT
- Polish: Roman Bagiński - SP4JEU

Several authors have placed their digital modem code and signal processing code in the public domain and their source was either an inspiration or in some cases formed the backbone of the code used in *Fldigi*.

- AE4JY - WinPsk - a windows application
- Tomi Manninen, OH2BNS - gmfsk - a great digital modem program for Linux
- Hamish Moffatt, VK3SB - dominoEX code originally for gmfsk
- Dr. Steven W. Smith - author of "Digital Signal Processing", who has kindly placed an entire book on digital signal processing on the internet. (http://www.dspguide.com)

If you make a side-by-side comparison between gmfsk and fldigi source code you will see that they follow the same general structure. The primary difference is that gmfsk is written in the C language and uses the gnome/gtk libraries for the user interface. *Fldigi* is a C++ application that uses the Fast Light Tool Kit (Fltk) gui library. The design of *Fldigi* puts emphasis on separating the user interface from the sound card and transceiver input/output operations. Nearly all modern digital modem programs use a programming paradigm called "threads." Threads are light weight processes that share the same memory space, but each has its own stack. The use of threads makes the program look and feel responsive to the user while a lot of code is being executed in the background.

Many of the modem source code files are C to C++ rewrites from the gmfsk application. They say that copying is the best form of flattery and gmfsk simply had the best explanations and the easiest source code to read and understand. The author had also spent several months creating improvements and fixing bugs in the original gmfsk application. That exercise was the impetus to create *Fldigi*.

The Fast Fourier Transform used by *Fldigi* is a rewrite of John Green's public domain FFT code (FFT for RISC for MAC). The rewrite is in C++ and is implemented as a C++ template. Some of the signal processing algorithms used in *Fldigi* are from Dr. Smith's book. His on-line publication is sufficient to allow you to become fluent in FFT analysis and the creation of digital filters. I printed the relevant pdf files and then purchased the hard bound copy. Improvements to the original gmfsk signal processing algorithms can all be attributed to this excellent source.

And last but certainly not least, I must thank the crew who perform alpha testing and on-line support of the application. These are stalwart amateurs who risk their operating system and radio equipment in testing, testing and more testing. Their only reward is in being able to influence the design of the application and the fun of seeing it work and the bugs disappear. Thank you to:

Call Name	Call Name	Call Name	Call Name
4Z5ST Boris	K3GAU David	KU1T Zibi	VA3DB Dianne
AA0HW Chuck	K4XTT Victor	KV9U Rick	VE3IXI Dave
AC7JN Dave	K6KAR Kirk	N0NB Nate	VK2TMG Brett
CT1DRB David	K7BRK Chris	N2AMG Rick	VK4BDJ David
CX7BF Walter	K4RE Brian	N4UM Tim	W3NR Ed
DF4OR Ekki	K9AO Rick	N4ZNV Mike	W4ROS Ross
DK1JBE Tom	KB3FN Lynn	N6WFL Jason	W6JVE Jim
DL6XAZ Fred	KD0AR Mike	N8FQ Joe	WA3VPZ Marshal
DL8FCL Walter	KD4O Phil	NN8B Don	WA4SXZ Rich
G0UZP Paul	KD8DKT Mike	NT1G Skip	WB8ROL Gary
G3TDJ Andy	KE3Y Travis	OZ4KK Erik	WD4FDW Steve
G6CKR Roger	KH6TY Skip	PA0R Rein	WD4FNY Bill
G8SQH David	KL7NA Rob	PA3GWH Richard	WU9Q Bob

and many others whose names are not listed, please accept my apology.

The test team is representative of users on Windows, Linux, Free BSD and OS X operating systems. They have varying interests from very slow speed CW to high speed keyboard full break-in CW, from RTTY contesters to PSK rag chewers. They have insisted that fldigi perform well under all of those operations. I have been amazed by the global distribution of the testing team. It is easy to think that the internet will be the death of amateur radio. On the contrary it opens up so many additional ways for us to be cooperative.

Return to Top of Page

Return to Main Page