Fldigi-3.20

Contents

License

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
- Leigh Klotz - WA5ZNU
- Stephane Fillod - F8CFE
- John Douyere - VK2ETA
- Joe Veldhuis - N8FQ
- Chris Sylvain - KB3CS
- Gary Robinson - WB8ROL

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
- Takuya OOURA - a generic Fast Fourier Transform for real valued data streams - http://momonga.t.u-tokyo.ac.jp/~ooura/fft.html
- 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 Takuya Ooura's C code. The rewrite is in C++ but you will see the strong resemblence to Takuya's original if you study both. Takuya's FFT code was also used in the Winpsk program. 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.

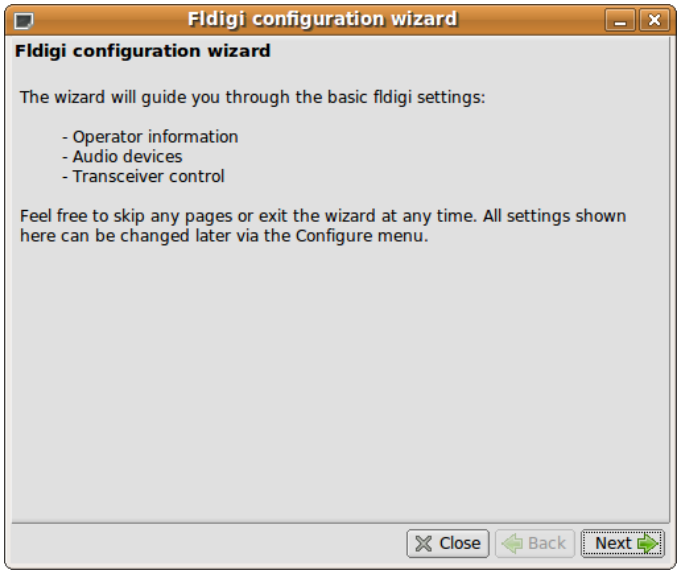
And last but certainly not least, I must thank the crew who perform alpha testing and on-line support of the application. These are stalwart amateurs who risk their operating system and radio equipment in testing, testing and more testing. Their only reward is in being able to influence the design of the application and the fun of seeing it work and the bugs disappear. Thank you to:

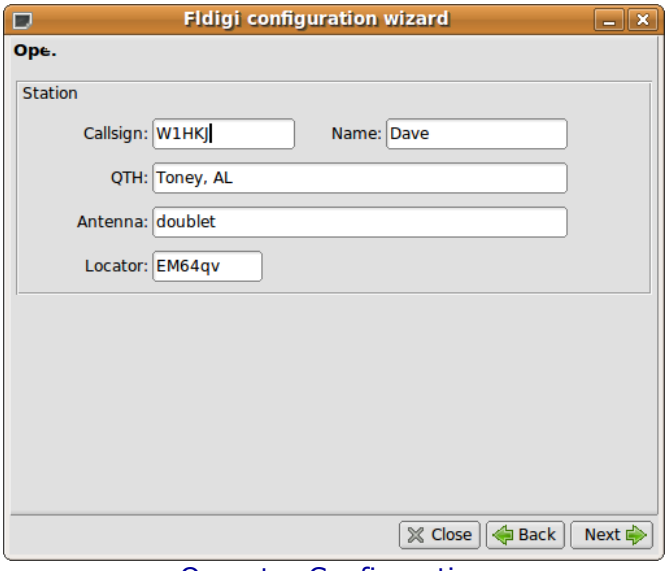
Call	**Name**	**Call**	**Name**	**Call**	**Name**	**Call**	**Name**
4Z5ST	Boris	K3GAU	David	KU1T	Zibi	VA3DB	Dianne
AA0HW	Chuck	K4XTT	Victor	KV9U	Rick	VE3IXI	Dave
AC7JN	Dave	K6KAR	Kirk	N0NB	Nate	VK2TMG	Brett
CT1DRB	David	K7BRK	Chris	N2AMG	Rick	VK4BDJ	David
CX7BF	Walter	K4RE	Brian	N4UM	Tim	W3NR	Ed
DF4OR	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.

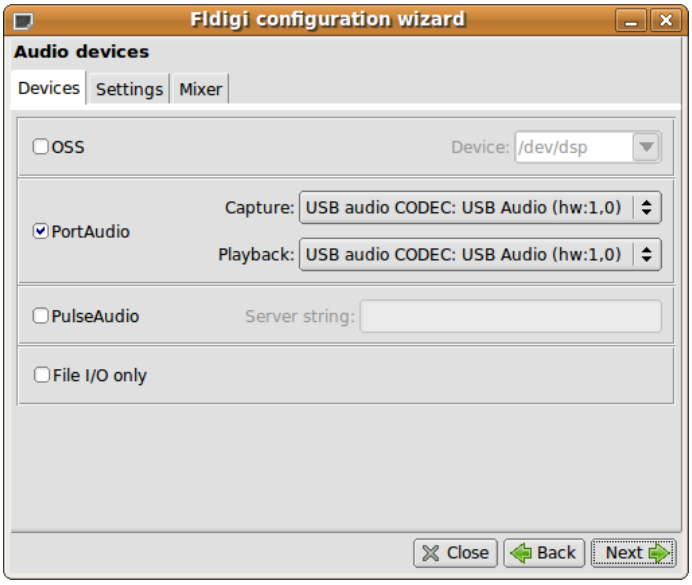
New Install Wizard

The new installation wizard borrows from the normal configuration dialogs. You will find information on setting each of the 3 wizard dialog pages on the associated configuration link.

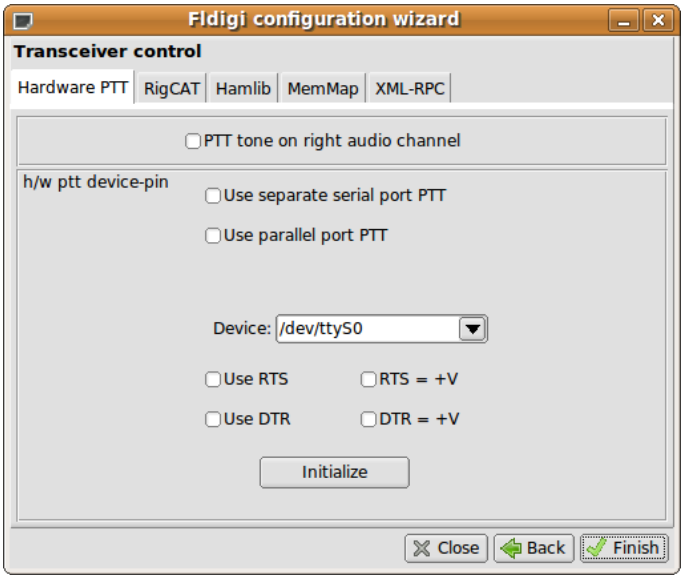




Operator Configuration



Sound Card Configuration



Transceiver Configuration

Installing Fldigi

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.

Hamlib required

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.

libsamplerate required

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.

PortAudio2 required

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.

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.

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\

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.status 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.

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. Tha 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 greys 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?

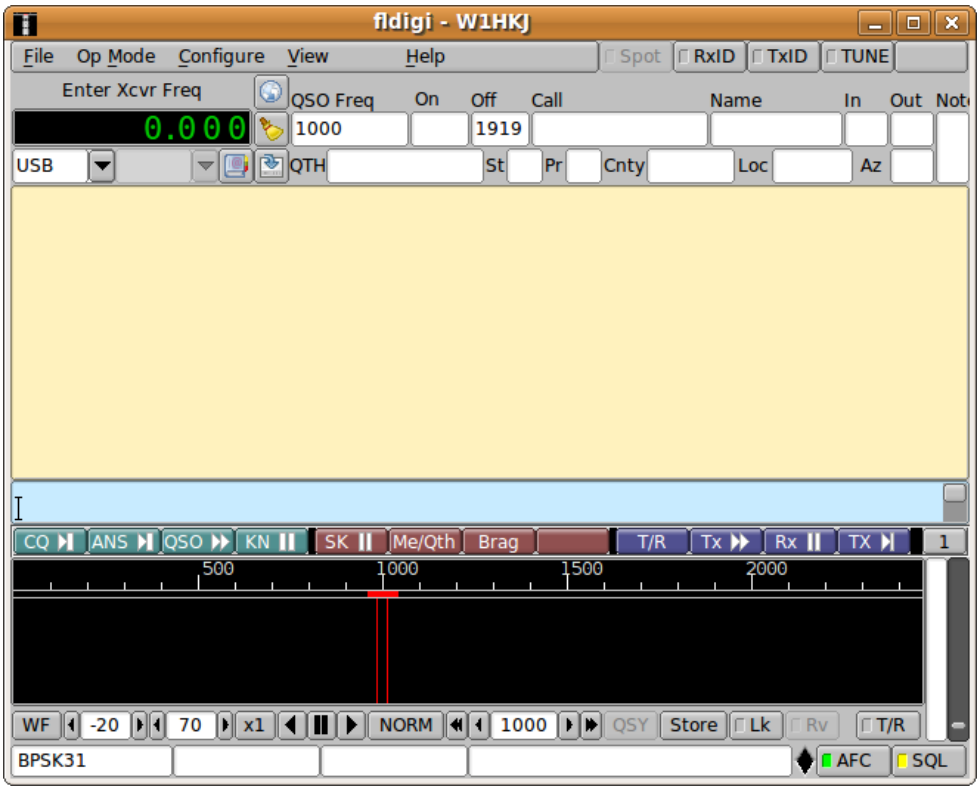
I solved the mystery! Cockpit error, kind of! I tried installing MULTIPSK and got the "no sound card error" also. Went into the Control Panel to see what was going on. When I looked at sound input it said there was NO microphone plugged in. Then the light came on and I remembered that you MUST have a 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 querk 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.

New Installation

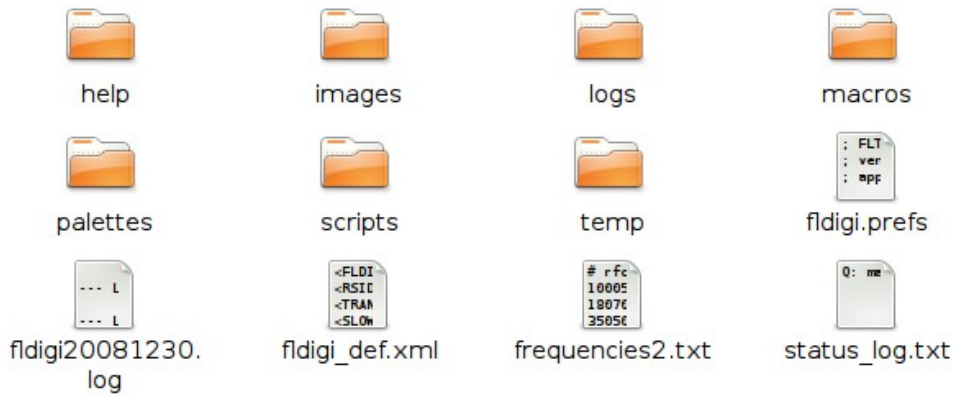
fldigi's opening screen looks like the following when starting fldigi for the first time or when setting up a second or subsequent instance using the --config-dir command line switch. The Wizard has been completed and the callsign, W1HKJ, entered.



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.

XP / W2K	C:\Documents and Settings\<username>\fldigi.files
Vista	C:\Users\<username>\fldigi.files
Linux	/home/<username>/.fldigi

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

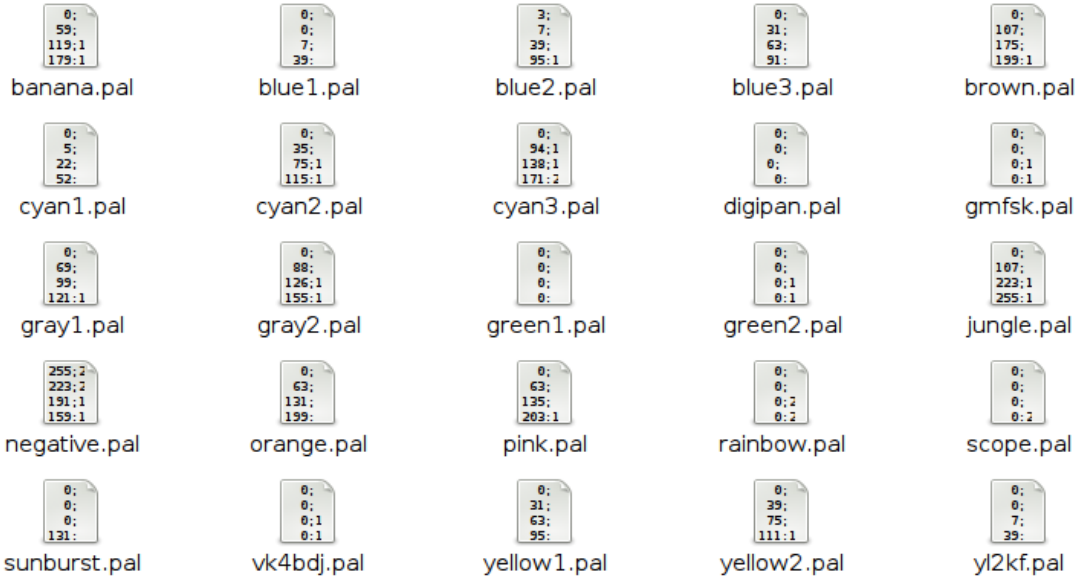


The help, images, logs, scripts and temp folders will be empty. They will contain program created files as you use the program or you may post files in those folders for use by fldigi. Images to be sent with the MFSKpic mode should be placed in images. Your logbook database will appear in logs. If you 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:



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
255; 58;  0
```

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.

Configuring

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.

You should initially configure the following:

Operator	UI	**Waterfall**	Modems	**Rig**	**Audio**	Id	Misc	Callsign DB

and

Colors & Fonts

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

Operator	**UI**	Waterfall	Modems	Rig	Audio	**Id**	**Misc**	**Callsign DB**

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

CW	**DominoEX**	**FeldHell**	**MT-63**	**Olivia**	**Contestia**	**Psk**	**Rtty**	**Thor**

To learn more about the characteristics of specific digital modes look here: Digital Modes, Sights & Sounds.

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.

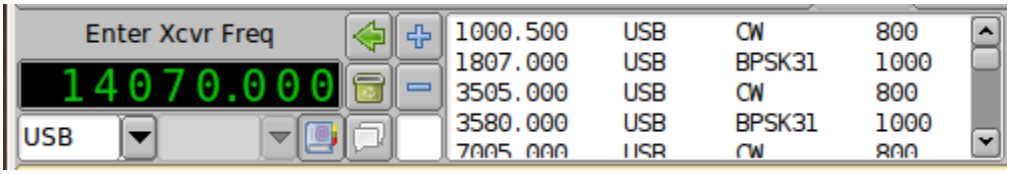
Exit the program and restart it to test that your configuration was saved and is working correctly.

Your fldigi install is now ready for you to start receiving and transmitting digital signals.

Fldigi recognizes if any configuration changes are made and not saved. You will then be prompted to save the configuration when exiting the program.

Rig Control

CAT not enabled



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.

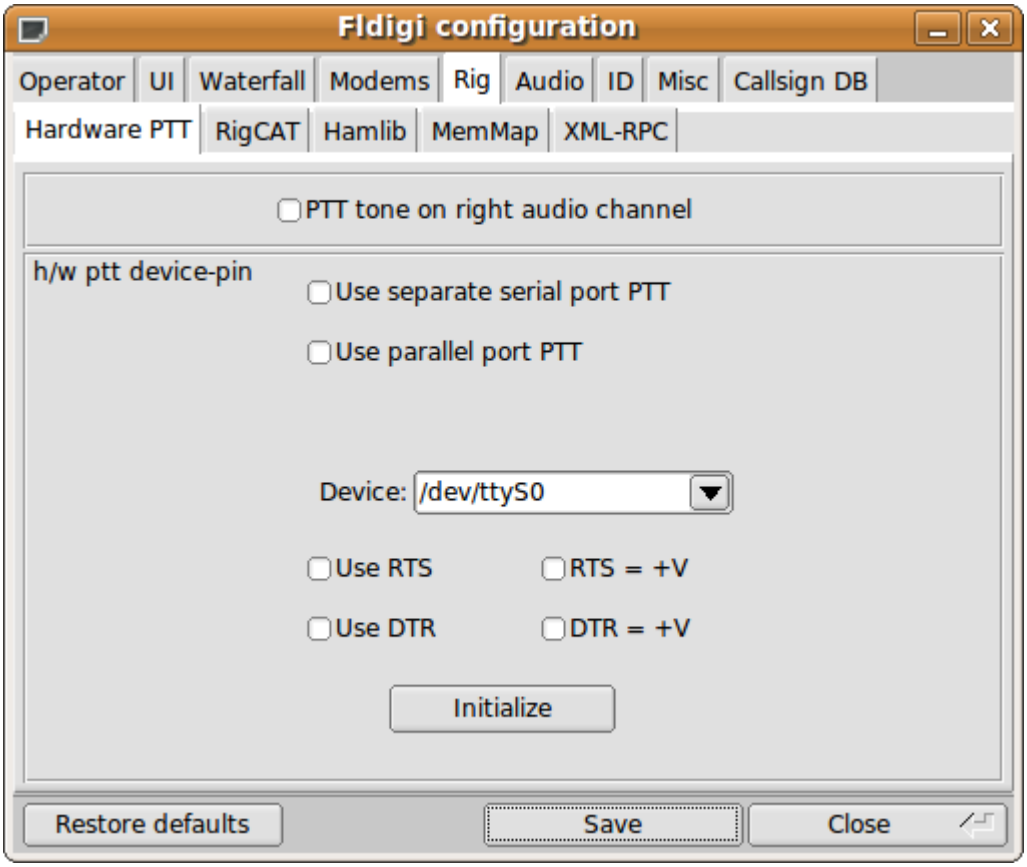
Manual entry of frequency can be accomplished by clicking on any digit and then entering the

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.

Rig Configuration

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. The circuit will be similar to that used for QSK control. This might be a convenient way to create a PTT signal for a small notebook or netbook computer that does not have a serial or a parallel port.

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.

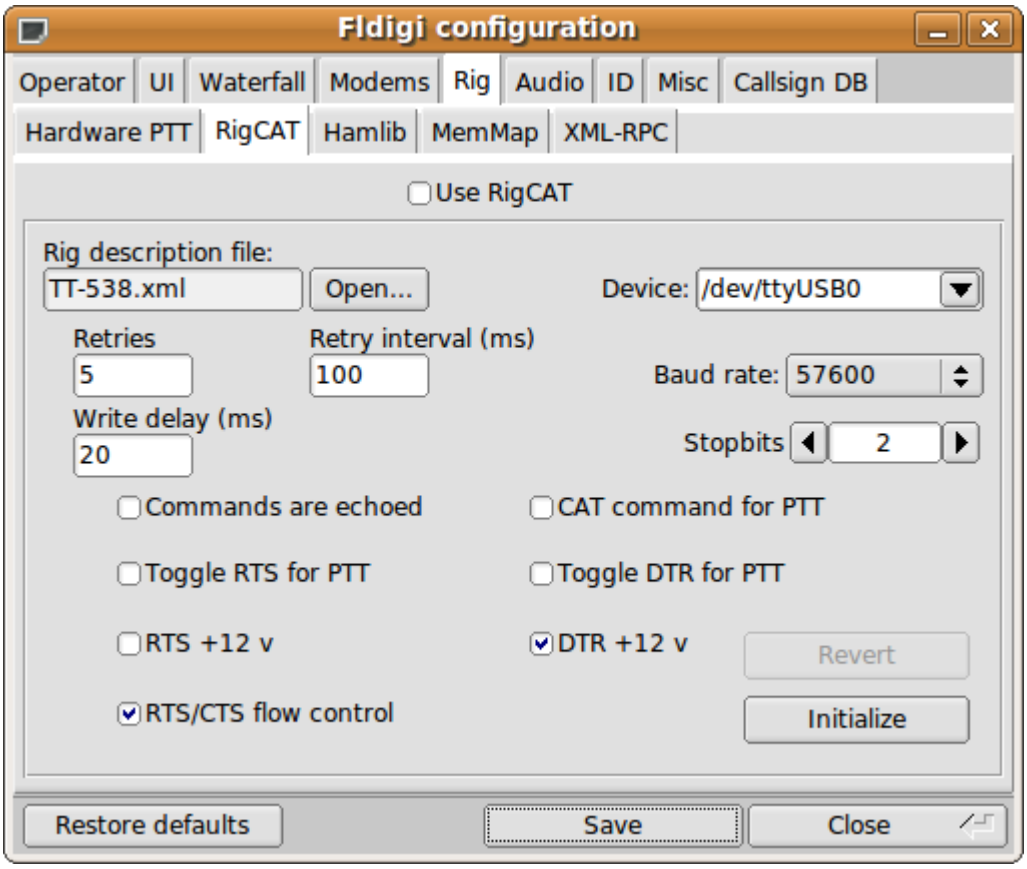
You select this operation by checking the "Use serial port PTT". Select the serial port from the list (fldigi will have searched for available ports). Then specify whether the h/w uses RTS or DTR and whether a + or - voltage is required to toggle PTT on. You can use a serial port for control with the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

Fldidi 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.

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.

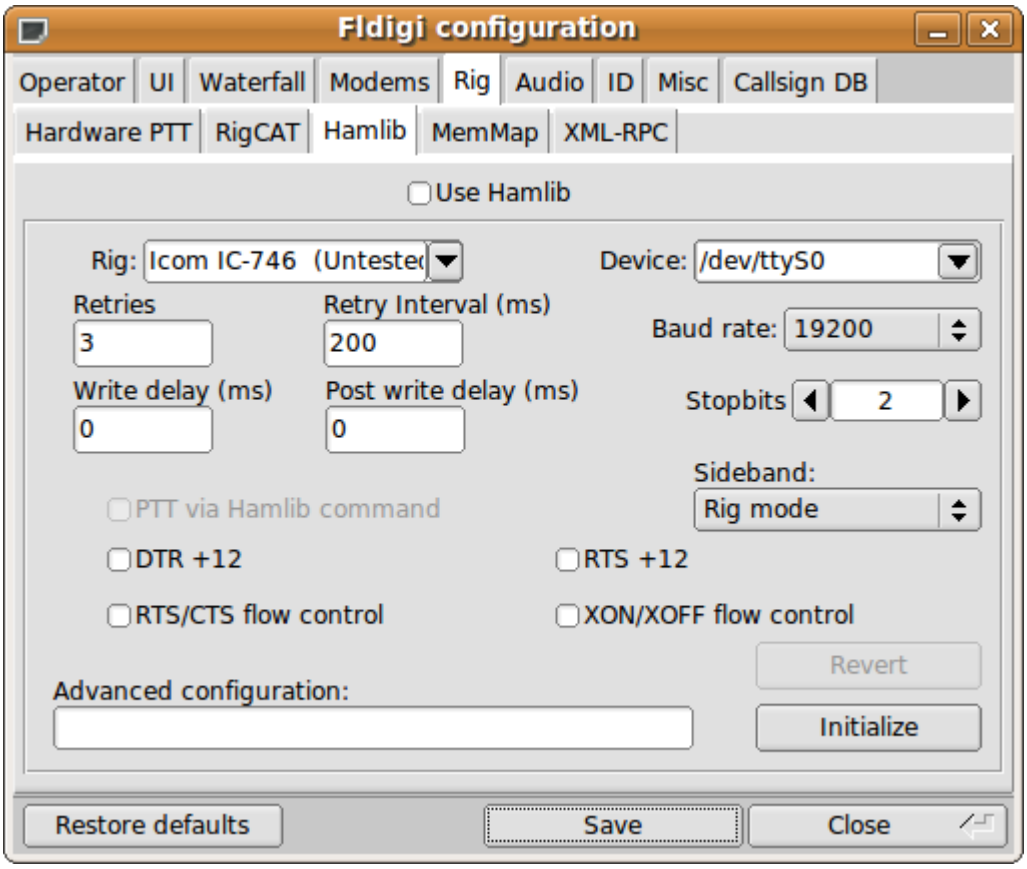
If you are using a transceiver or a rig interface such as CI-V that echos all serial data you check off the "Commands are echoed" box. That will suppress fldigi trying to respond to a command it just sent to the transceiver.

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.

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. The fldigi implementation of hamlib differs on the various OS for which it is targeted. On the Unix/Linux based systems the hamlib is a shared library which the user must have installed on his or her system. This is the standard way of handling hamlib on Unix/Linux systems. On Windows the entire hamlib library has been compiled and statically linked into the application executable. No additional dynamic link libraries are necessary. This approach simplifies the installation of fldigi on Windows platforms.

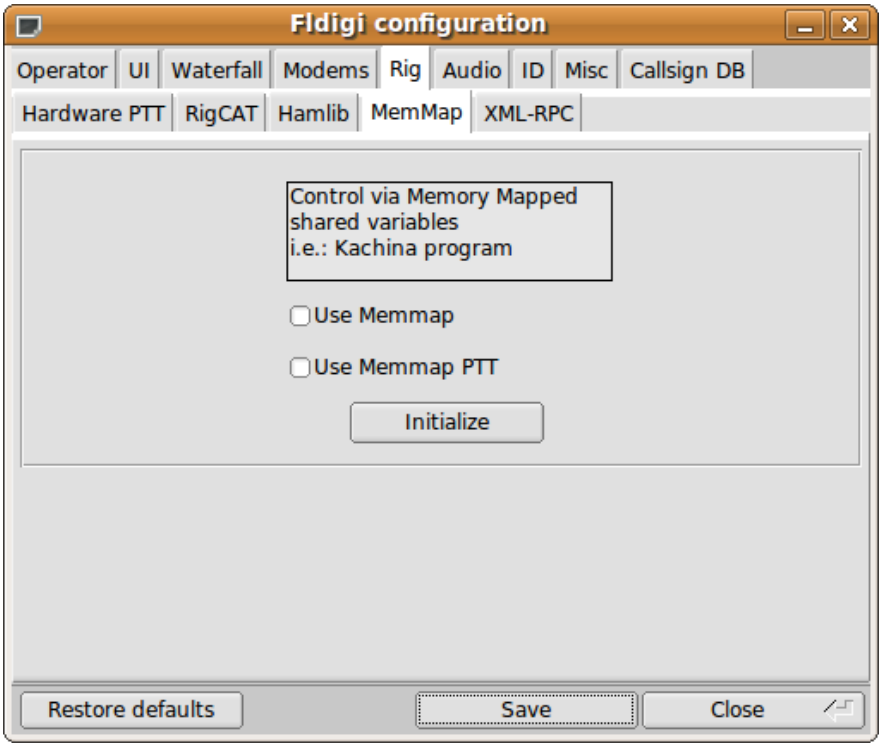


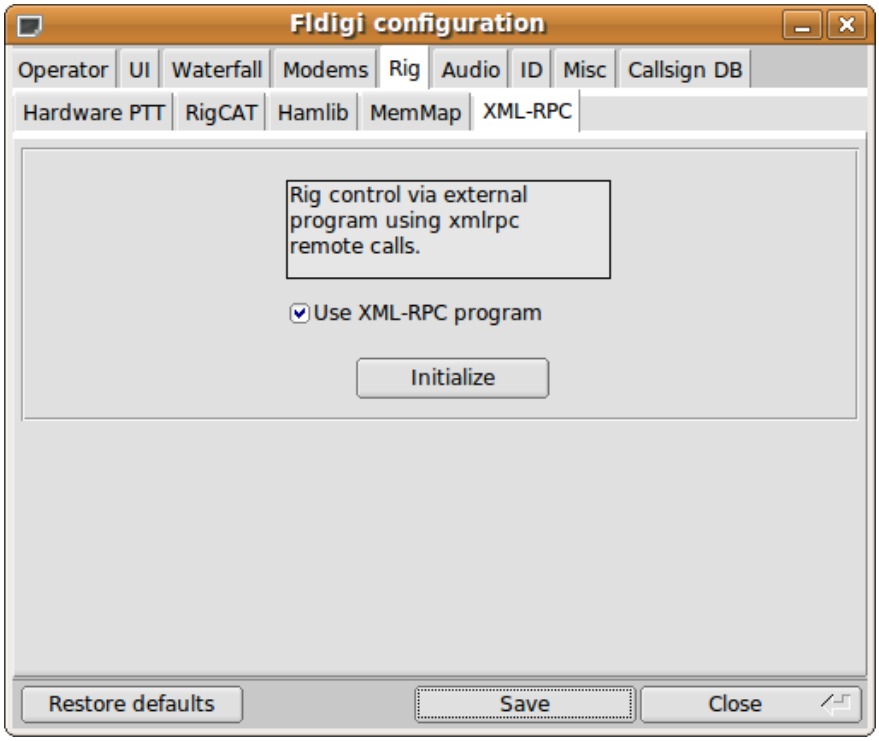
Select your transceiver from the list of supported units. Then select the serial port and baud rate. If you are familiar with the hamlib library you can send various startup sequences to the rig using the advanced configuration. PTT control can be achieved using CAT commands or via DTR / RTS on the same 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.

Memory Mapped CAT & Xml-Rpc CAT





Memory mapped control is selected if you are operating a Kachina 505DSP using the W1HKJ control software for that rig.

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

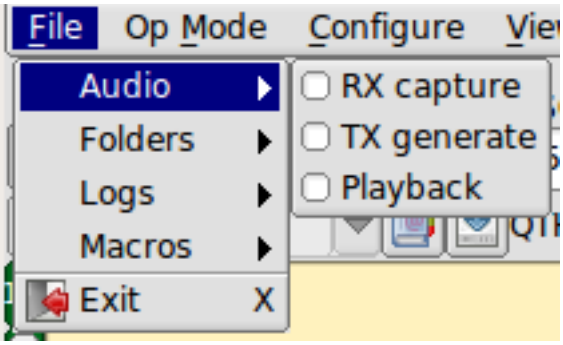
fldigi companion transceiver control program.

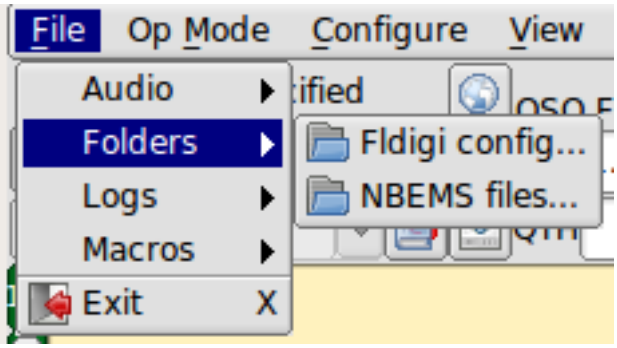
If you are using a third party interface such as DxKeeper Bridge you might be instructed to select this method of CAT.

Menus

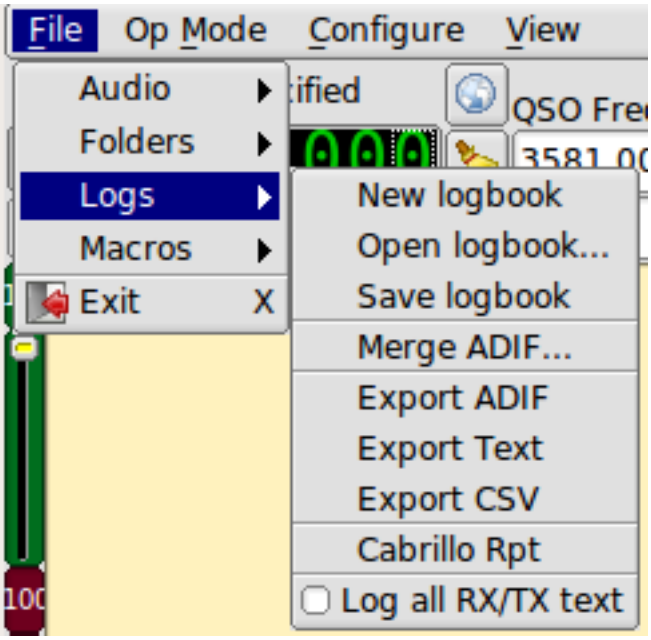
The menu heirarchy is:

- Files

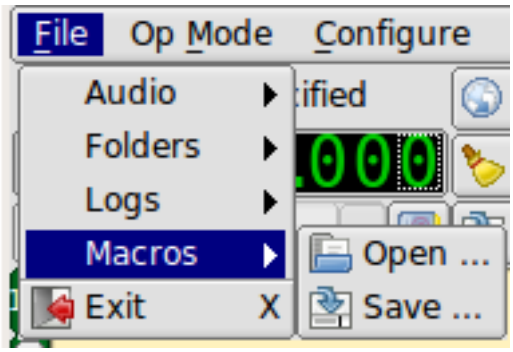




  - Rx Capture - allows capturing the incoming audio to a wav file
  - Tx Generate - allows capturing the generated tx audio to a wav file
  - Playback - playback a previously captured or generated wav file
  - Fldigi config... - open the OS native file explorer to the folder containing the fldigi operating & data files
  - NBEMS files... - open the OS native file explorer to the folder containing the NBEMS data files



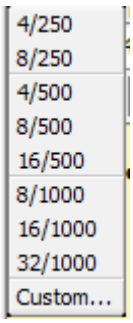
- Create a new logbook
- Open an existing logbook
- Save the current logbook
- Merge current log with an ADIF file from another source
- Export selected or all logbook records to an ADIF formated file - see Log Exports.
- Export selected or all logbook records to a text file suitable for printing
- Export selected or all logbook records to a tab delimited file
- Create a Cabrillo contest report. – see Cabrillo Reports.
- write all received and transmitted text to the file "fldigi.log" which will be in the $HOME/.fldigi directory



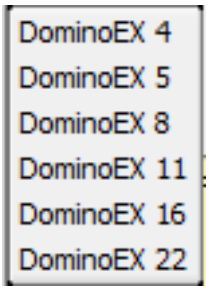
- Open Macros - open a macro definition file ... changes the MACRO keys immediately
- Save Macros - save the current macro definitions to a designated file
- Exit - exit the program closing down the various interfaces in a nice controlled manner.

- Op Mode - the current operating mode will show as a highlighted menu item.

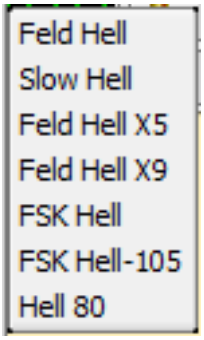
- CW - receive CW 5 to 200 WPM and transmit on any audio frequency using AFCW
- Contestia



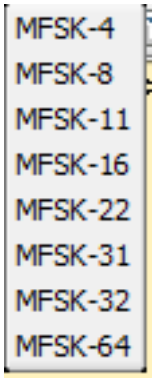
  - Contestia 4/250
  - Contestia 8/250
  - Contestia 4/500
  - Contestia 8/500
  - Contestia 16/500
  - Contestia 8/1000
  - Contestia 16/1000
  - Contestia 32/1000
- DominoEX



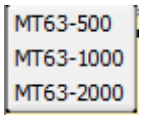
  - dominoex 4
  - dominoex 5
  - dominoex 8
  - dominoex 11 - the default calling mode for dominoEX
  - dominoex 16
  - dominoex 22
- Feld



  - Feld-Hell
  - Slow-Hell
  - Feld-Hell X5
  - Feld-Hell X9
  - FSK-Hell (also called FM-Hell by some programs)
  - FSK-Hell105
  - Hell-80
- MFSK

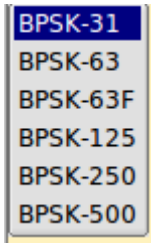


  - mfsk 4 (4 tones
  - mfsk 8
  - mfsk 11
  - mfsk 16
  - mfsk 22
  - mfsk 31
  - mfsk 32
  - mfsk 64
- MT-63



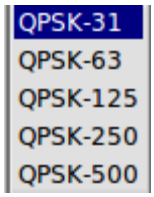
- MT63-500 - interleave & extended characters set on configuration tab
- MT63-1000 "
- MT63-2000 "

- PSK



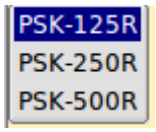
- psk 31 - phase shift keying - 31.625 baud
- psk 63 - phase shift keying - 63.25 baud
- psk 63F - phase shift keying with FEC - 63.25 baud
- psk 125 - phase shift keying - 126.5 baud
- psk 250 - phase shift keying - 253 baud
- psk 500 - phase shift keying - 506 baud

- QPSK



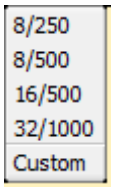
- qpsk 31 - quadrature phase shift keying - 31.25 baud
- qpsk 63 - quadrature phase shift keying - 63.25 baud
- qpsk 125 - quadrature phase shift keying - 126.5 baud
- qpsk 250 - quadrature phase shift keying - 253 baud
- qpsk 500 - quadrature phase shift keying - 506 baud

- PSKR



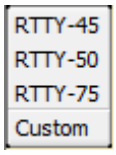
- psk 125R - phase shift keying, with FEC and interleaving - 126.5 baud
- psk 250R - phase shift keying, with FEC and interleaving - 253 baud
- psk 500R - phase shift keying, with FEC and interleaving - 506 baud

- Olivia



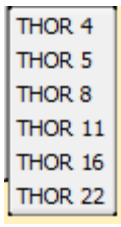
- 8/500 8 tone, 500 Hz wide signal format
- 16/500 16 tone, 500 Hz wide signal format
- 32/1000 32 tone, 1000 Hz wide signal format
- Custom - tones and bandwidth configurable on Olivia tab

- RTTY



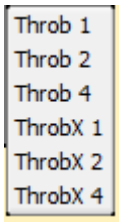
- RTTY-45 45 Baud Baudot, 170 Hz shift, used primarily in U.S.
- RTTY-50 50 Baud Baudot, 170 Hz shift, used primarily in Europe
- RTTY-75 75 Baud Baudot, 800 Hz shift
- Custom - Baud Rate, Baudot/ASCII, Shift etc configurable on RTTY tab

- Thor



- Thor-4
- Thor-5
- Thor-8
- Thor-16
- Thor-22

- Throb



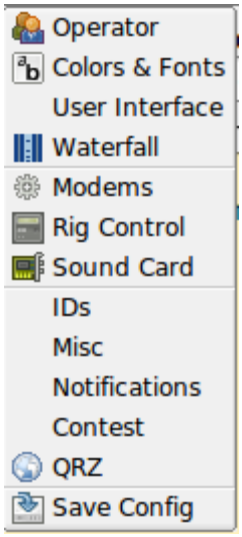
- Throb1
- Throb2
- Throb4
- ThrobX-1
- ThrobX-2
- ThrobX-4

- NBEMS modes

These are the recommended modes to use when fldigi is used with flarq to form the NBEMS, Narrow Band Emergency Message System.

- WWV - special receive only modem used for calibrating sound card
- Freq Anal - used for carrier detection and frequency measurement

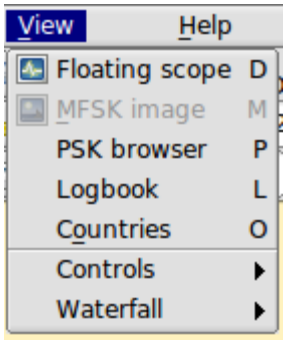
- Configure

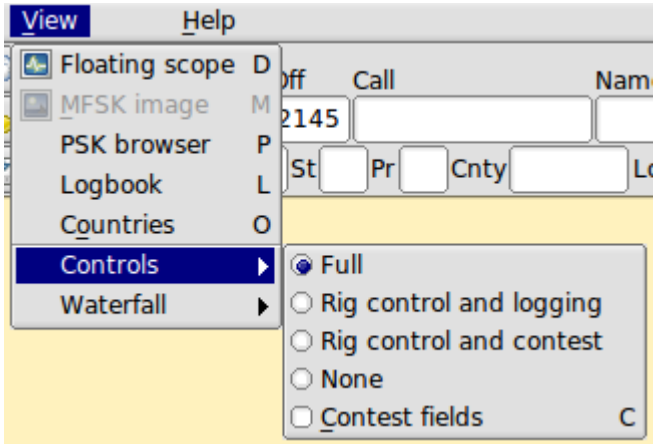


- Operator - open the operator configuration tab
- Colors - Fonts - select the colors and fonts for various main dialog controls
- User Interface - configure various aspects of the main fldigi dialog
- Waterfall - open the waterfall configuration tab
- Modems - open up modem configuration to the current modem tab
- Rig Control - open the rig control configuration tab
- Sound Card - open the sound card configuration tab
- IDs - configure various IDentification signals that can be sent and received by fldigi
- Misc - open the miscellaneous configuration tab (sweet spot definitions)
- QRZ - open the QRZ/Hamlog access configuration tab

- Save Config - write the current configuration to the file ~/.fldigi/fldigi_def.xml

- View

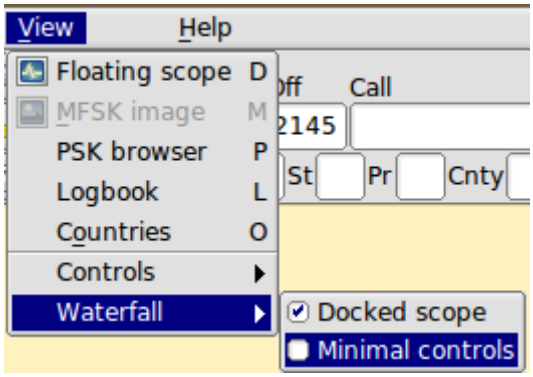




- Floating Scope - Opens up a resizeable, moveable scope display
- MFSK Image - Opens the MFSK picture image (if being received)
- PSK Browser - open the psk viewer dialog to display up to 30 simultaneously decoded psk signals
- Logbook - Opens up the logbook dialog

- Countries - Opens a dialog which displays the DXCC list
- 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

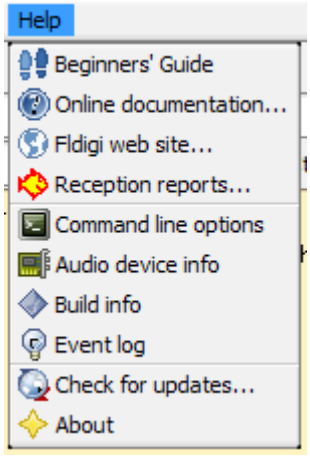


2nd line in qso logging area; provides access to contest logging fields

- 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

- 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

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.

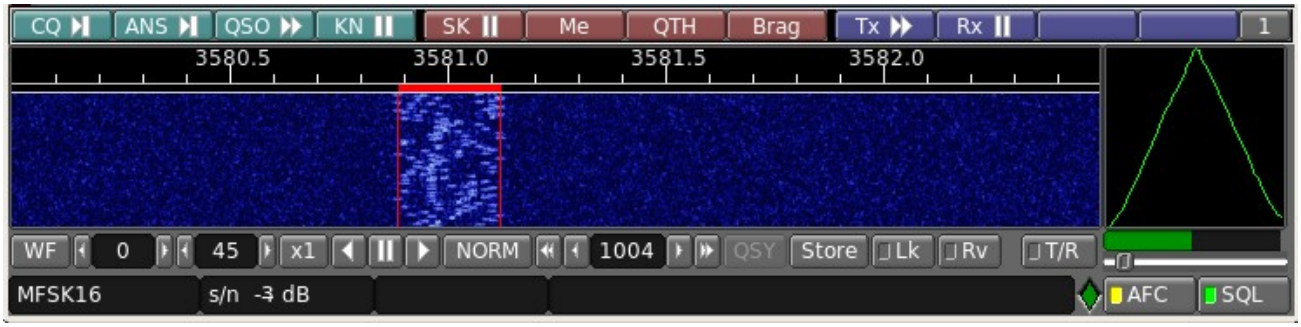
RxID button - toggles the detection of Reed Solomon Identification codes.

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.

Operating Controls & Displays



The main display for fldigi is the waterfall display shown above in color and in scale x1. The above display shows fldigi configured by invoking the following command line switches:

fldigi -bg2 black -fg white -bg grey40 --wfall-height 150 --wfall-width 3000 --font sans:12

The macro button colors are set to the default on the colors-fonts dialog.

You don't have to remember all of those switch settings every time you start fldigi. Just enter them on the Command Line, Launcher tab for the desktop icon properties (Gnome desktop).

Or from Windows XP on the Target Line, of the Shortcut tab for the properties dialog associated with the fldigi desktop icon. The fg, bg and bg2 specification on Windows is not the same as Linux. On Windows those three command line parameters need to be:

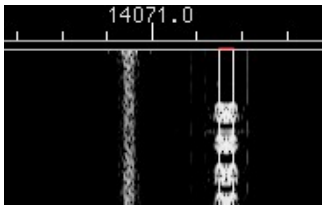
-bg2 FFFFFF -fg 000000 -bg 606060

where each color is specified by its RRGGBB component as a hexadecimal value.

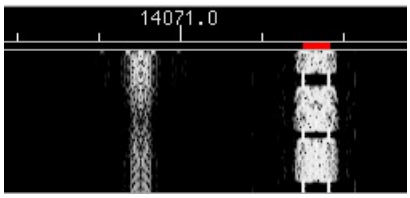
The button **Wtr** toggles the display between a waterfall and a spectrum display. This button acts as a rotary. Left clicking moves the display selection in one direction and right clicking in the other direction. The three display modes are **Wtr** - waterfall, **FFT** - spectrum (Fast Fourier Transform) 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.



X1 scale



X2 scale

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 1679 Hz. The red cursor is centered beneath 14071.679 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 effect.

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 RED 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 "Lck" off the TX audio frequency is always synchronized with the RX frequency.

With "Lck" 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 "Lck" is ON moving the cursor around will ONLY EFFECT 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.

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 accordanced 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.

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.

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.

Mouse & 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!

- **Main window**
    - **Text input fields**

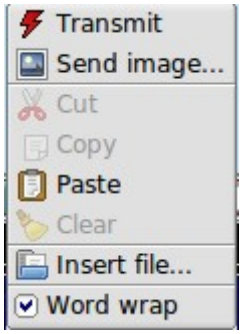
      Most text fields use a combination of CUA (PC) and Unix-style keybindings. Text can be marked, copied, pasted, saved to a file as well as transfer to other main panel controls. A right click on any text control will open a context sensitive menu for that particular control. A full list can be found on the FLTK web site.

      The received/transmitted text widgets use CUA key bindings with some modifications:

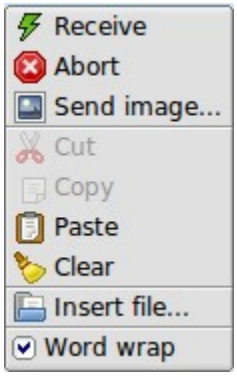
        - RX text
          This widget is read-only and ignores shortcuts that would modify its contents. See logbook for details on the Rx right click popup menu system.

        - 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:



Fldigi Receiving Editing Tx text



Fldigi Transmitting

Select:

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.

- **Waterfall display**

  Most of fldigi's unusual shortcuts are specific to this widget.

  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

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-Left click/drag likewise, with a return to the previous frequency when the button is released
- 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

- **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

- **Digiscope display**

    - Mouse wheel Change AFC/BW, same as Ctrl-mouse wheel on the waterfall

- **Rig control window**

  There are some shortcuts in addition to those described in the Rig Control

    - **Frequency display**

        - Left/Right arrow key change the frequency by one 1 Hz
        - Up/Down arrow key change the frequency by 10 Hz

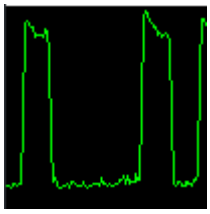
- **Frequency list**
    - Shift-Left click delete the line under the cursor
    - Middle click replace the line under the cursor with the current frequency/mode/modem

- **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

Digiscope Display

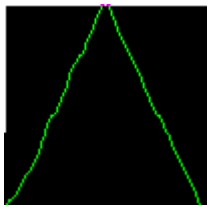
Fldigi provides several different views of the decoded signal with its waterfall, text and a scope displays. The scope display is either a separate moveable, resizeable dialog that is opened from the "View/Digiscope" menu item or a docked scope.

CW

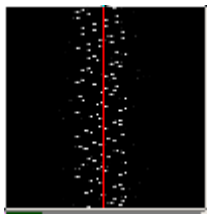


The CW signal will consist of the time domain amplitude detected signal. The horizontal timing is dependent on CW speed, so that the display will appear similar independent of CW speed.

DominoEX / Thor

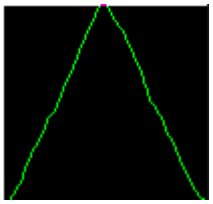


DominoEX and Thor have two alternate views available on the digiscope display. You can toggle between the views by left clicking on the digiscope display area. The triangular view shows data propogation through the interleave filter. As signal s/n degrades this display will become more wavy.



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.

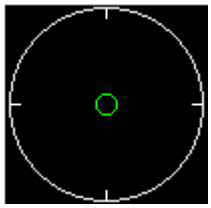
MFSK



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.

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:

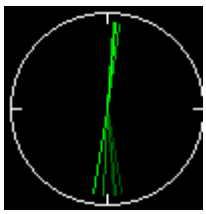


The display with no signal or below squelch level. If the SQL is off this display will be random vectors driven by noise.

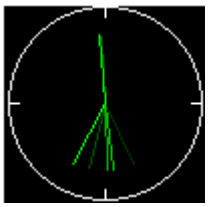
The display with a normal psk31 signal. The vector flips between 0 and 6 o'clock

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

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



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



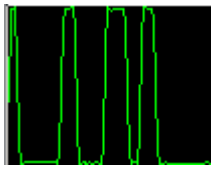
AFC enabled, Fading History / Amplitude Display Mode Selected (2nd left click on scope)

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.

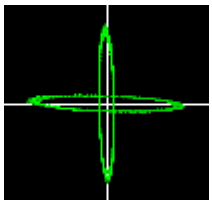
You can alter the appearance of the phase vectors by left clicking on the digiscope display. One click will give you a history of phase vectors that fade with time. A second click will give you a history of phase vectors that both fade with time and are amplitude significant. The third click returns 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.

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 tranmitting 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.



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.

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.

Tag	Description	Tag	Description
<FREQ>	my frequency	<RX>	receive, places ^r tag at end of expanded macro always the last macro tag executed
<MODE>	mode	<TX>	transmit
<MYCALL>	my call	<TX/RX>	toggle T/R
<MYLOC>	my locator	<SRCHUP>	search UP for signal
<MYNAME>	my name	<SRCHDN>	search DOWN for signal
<MYQTH>	my QTH	<GOHOME>	return to sweet spot
<MYRST>	my RST	<GOFREQ:NNNN>	move to autdo freq NNNN
<VER>	Fldigi version		
		<FILE:>	insert text file
<CALL>	other call		
<NAME>	other name	<IDLE:NN.nn>	idle signal for NN.nn sec
<QTH>	other QTH	<TIMER:NN>	repeat every NN sec
<LOC>	other locator	<TUNE:NN>	tune signal for NN sec
<RST>	other RST	<WAIT:NN>	delay xmt for NN sec
<INFO1>	S/N etc. in status		

	bar		
<INFO2>	IMD etc. in status bar	<CWID>	CW identifier
		<ID>	mode ID
<CLRRX>	clear RX pane	<TEXT>	video text
<GET>	text to NAME/QTH	<TXRSID:on|off|t>	Tx RSID on,off,toggle
<LOG>	save QSO data immediate	<RXRSID:on|off|t>	Rx RSID on,off,toggle
<LNW>	insert ^L tag into TX text stream save log entries to logbook when ^L is reached.		
<QSOTIME>	QSO time (HHMM)	<POST:+/-nn.n>	CW QSK post-timing
<ILDT>	LDT in iso-8601 format	<PRE:nn.n>	CW QSK pre-timing
<LDT>	Local datetime	<RISE:nn.n>	CW rise time
<IZDT>	ZDT in iso-8601 format	<WPM:NN>	CW WPM
<ZDT>	UTC datetime		
<LT>	current local time as HHMM	<ZT>	current Zulu time as HHMMZ
<LD>	current local date as YYYY-MM-DD	<ZD>	current Zulu date as YYYY-MM-DD Z
		<AFC:on|off|t>	AFC on,off,toggle
<CNTR>	contest counter		
<DECR>	decrement counter	<LOCK:on|off|t>	LOCK on,off,toggle
<INCR>	increment counter		
<XOUT>	insert exchange out text from contest dialog	<MACROS:>	change macro defs file
<XBEG>	mark start of "Exchange Out" text	<MAPIT>	open default browser to maps.google.com trying Postal Address / Lat-Lon / Locator in order
<XEND>	mark end of "Exchange Out" text, save to field	<MAPIT:adr/lat/loc>	Map starting with Postal Address / Lat-Lon / Locator in order. <MAPIT:adr> is the same as <MAPIT>

<SAVEXCHG>	save entire expanded macro text to "Exchange Out"		

<MODEM:CW>	<MODEM:MFSK-8>	<MODEM:BPSK31>	<MODEM:OLIVIA>	<MODEM:THOR4>
<MODEM:DomEX4>	<MODEM:MFSK16>	<MODEM:BPSK63>	<MODEM:OLIVIA:250:8>	<MODEM:THOR5>
<MODEM:DomEX5>	<MODEM:MFSK-32>	<MODEM:BPSK63F>	<MODEM:OLIVIA:500:8>	<MODEM:THOR8>
<MODEM:DomEX8>	<MODEM:MFSK-4>	<MODEM:BPSK125>	<MODEM:OLIVIA:500:16>	<MODEM:THOR11>
<MODEM:DomX11>	<MODEM:MFSK-11>	<MODEM:BPSK250>	<MODEM:OLIVIA:1000:8>	<MODEM:THOR16>
<MODEM:DomX16>	<MODEM:MFSK-22>	<MODEM:BPSK500>	<MODEM:OLIVIA:1000:32>	<MODEM:THOR22>
<MODEM:DomX22>	<MODEM:MFSK-31>	<MODEM:QPSK31>	<MODEM:RTTY>	<MODEM:THROB1>
<MODEM:FELDHELL>	<MODEM:MFSK-64>	<MODEM:QPSK63>	<MODEM:RTTY:170:45.45:5>	<MODEM:THROB2>
<MODEM:SLOWHELL>	<MODEM:MT63-500>	<MODEM:QPSK125>	<MODEM:RTTY:170:50:5>	<MODEM:THROB4>
<MODEM:HELLX5>	<MODEM:MT63-1XX>	<MODEM:QPSK250>	<MODEM:RTTY:850:75:5>	<MODEM:THRBX1>
<MODEM:HELLX9>	<MODEM:MT63-2XX>	<MODEM:QPSK500>		<MODEM:THRBX2>
<MODEM:FSK-HELL>		<MODEM:PSK125R>		<MODEM:THRBX4>
<MODEM:FSK-H105>		<MODEM:PSK250R>		<MODEM:WWV>
<MODEM:HELL80>		<MODEM:PSK500R>		<MODEM:ANALYSIS>

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

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.

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.

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.

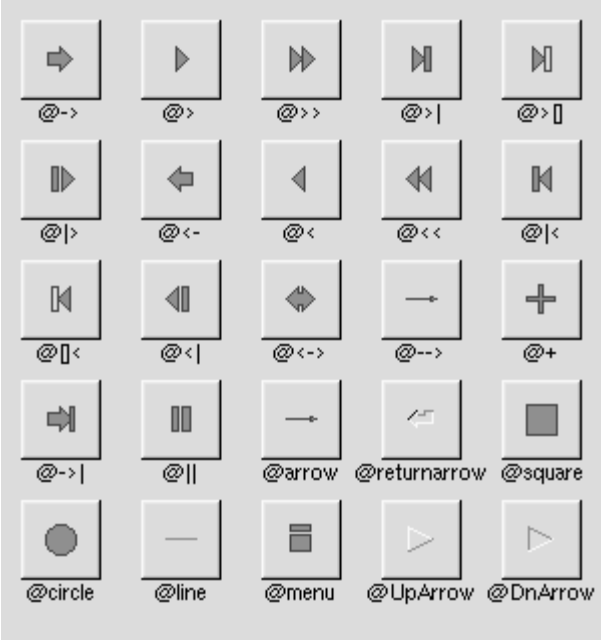
The macro reference in the pick list can be transfered to the current editing cursor location. Highlight the desired macro reference and then press the double << arrow key for each occurance of the reference to be put into the macro text. You can change the label name but any more than 8 characters may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



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 horizontaly, '%' flips it verticaly.
- [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:

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.

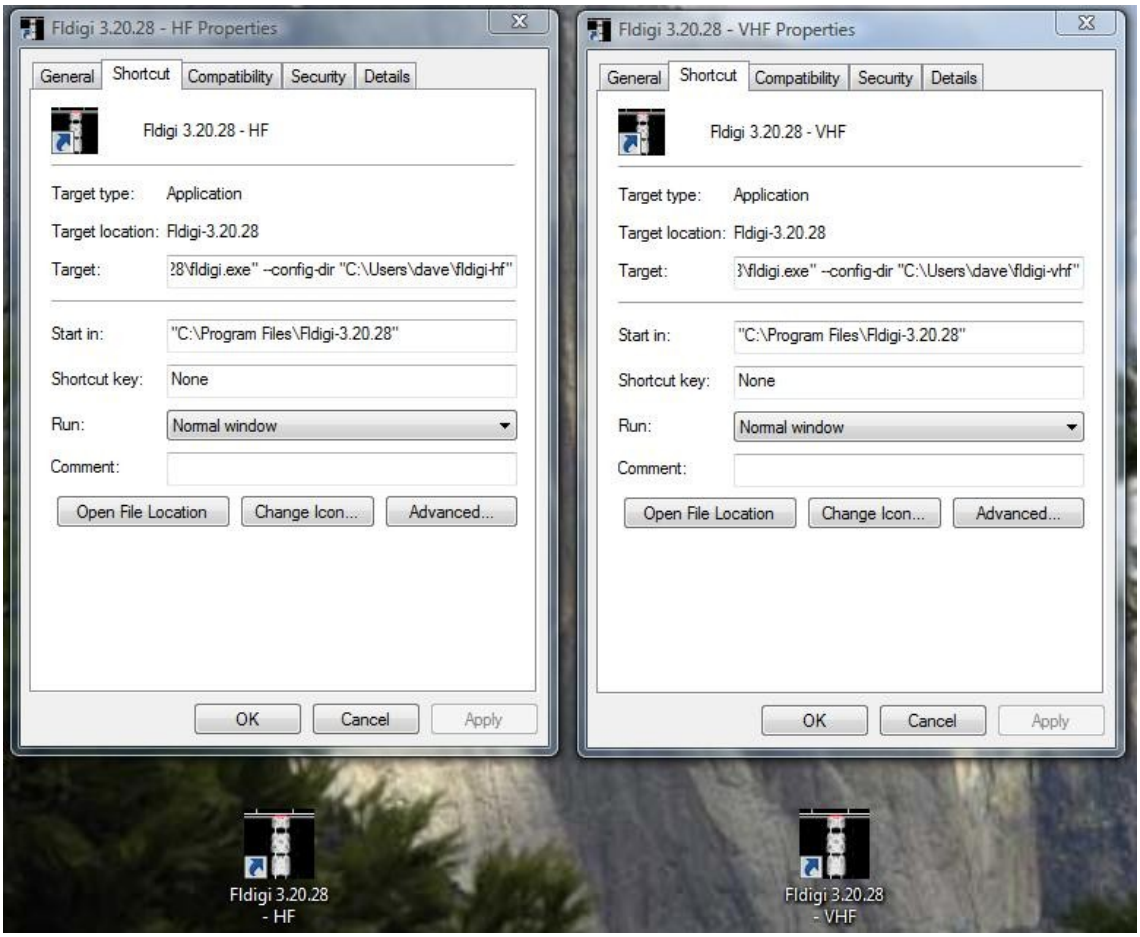
If you modify the macros and do not save them ("Files/Save Macros" on the main window) fldigi will prompt you to save the macros when you exit the program if you have the "Nag me" option selected.

The <EXEC>...</EXEC> macro tag provides a way to create external shell scripts and programs that can interact with fldigi. See Exec Macro to learn more about the <EXEC> macro tag.

Using contest macro tags

Refer to Contest-How-To

Operating Multiple Copies



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.

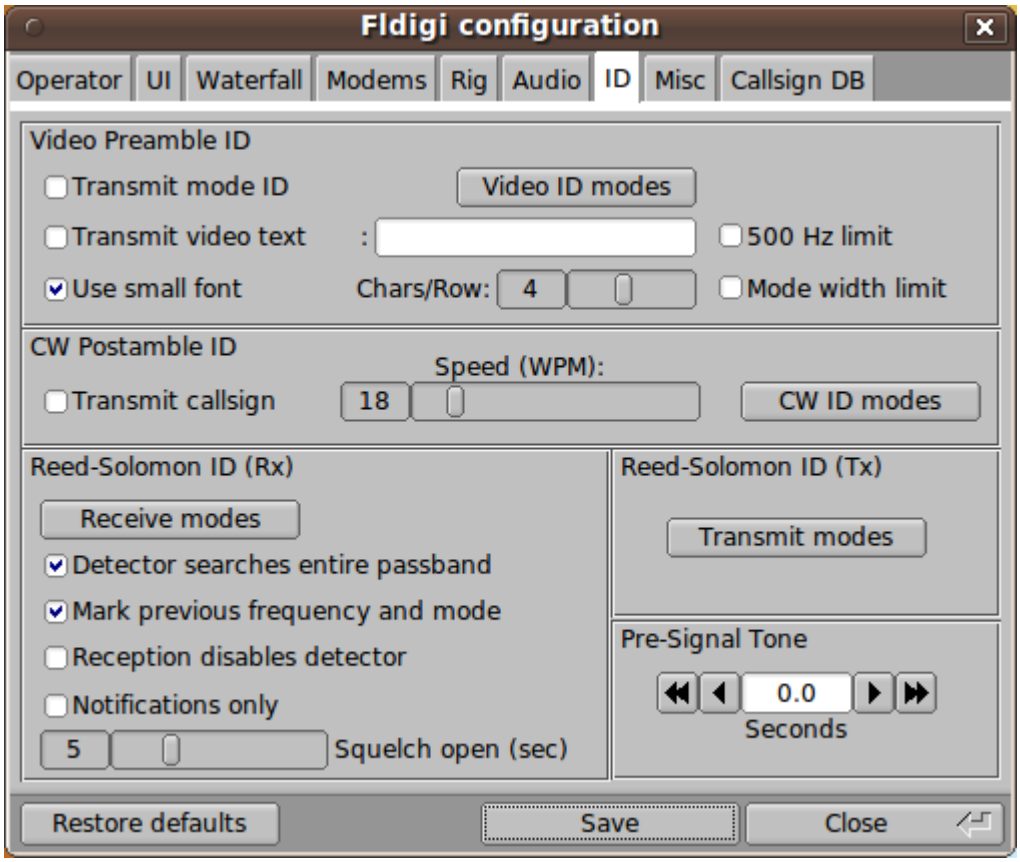
The screen shot shows how this is done on Vista, but the process is nearly the same on XP, Win7 and Linux. When you install fldigi it creates a desktop icon launcher. Most of the Linux window managers allow you to create a desktop launch icon. Make as many copies of the launcher as needed for your applications 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.

Id Configuration



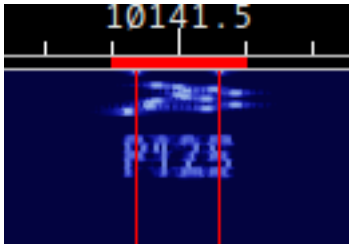
Fldigi offers several ways to identify the operator or mode that is being used. This is particularly useful when using a hard to recognize mode such as Thor, Olivia or MT63.

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.



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.

Cw postamble

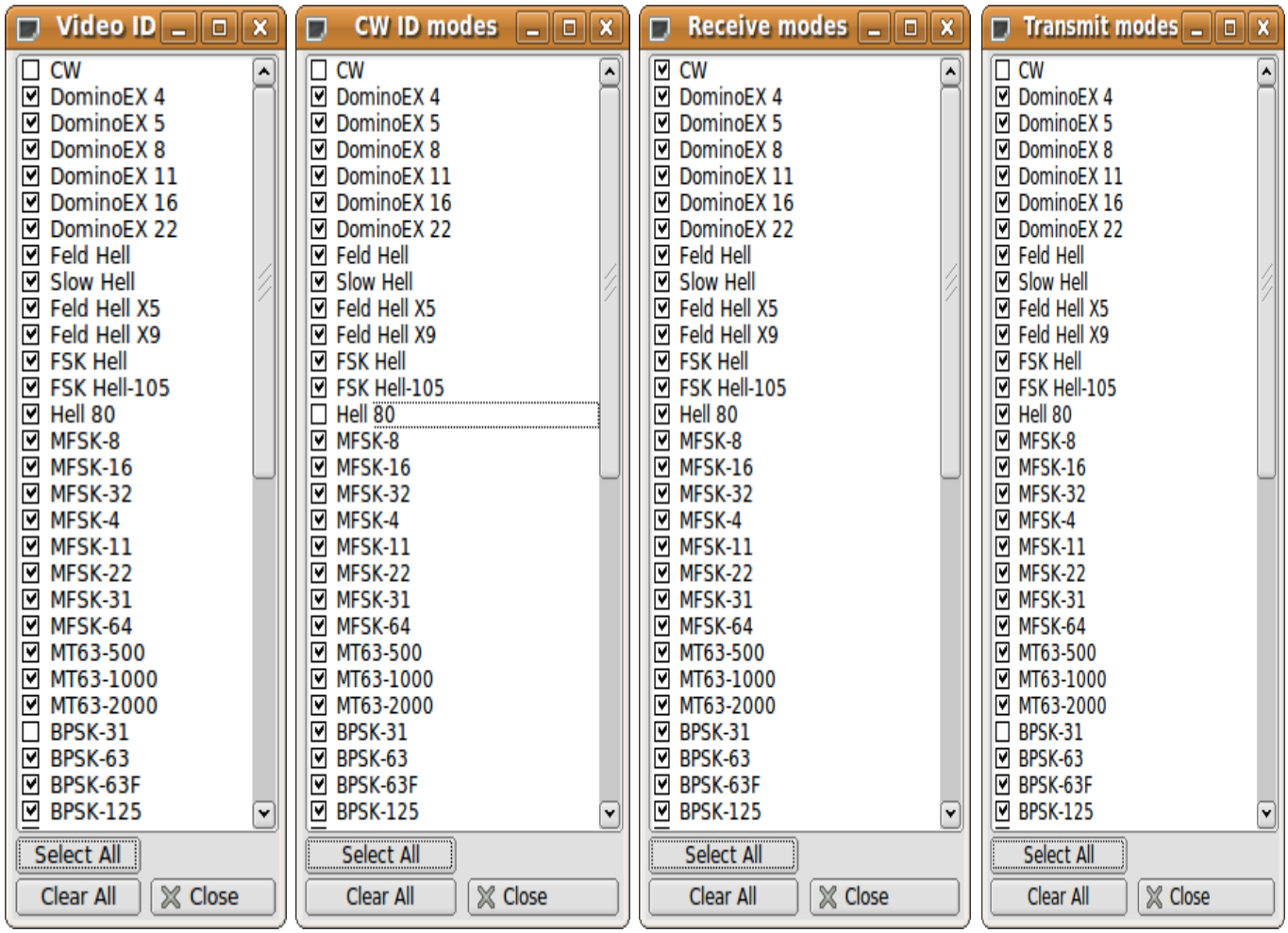
You can transmit your callsign in CW as a postamble to all modes except of CW (a bit redundant to do that). You can select which modes will include the cw postamble.

Reed Solomon Identifier

RSid, Reed Solomon Identifier, 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 RS Id 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.

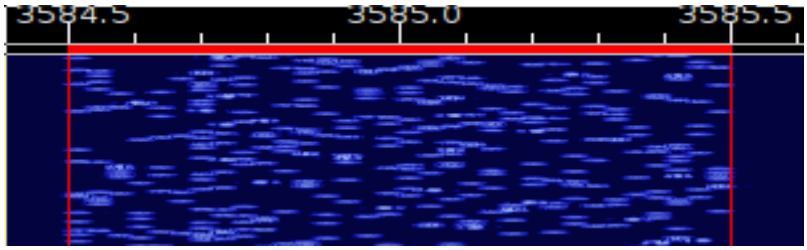


Pre-Signal Tone

This control allows you to specify the duration of a tone that will precede all transmissions. A duration of less than 0.3 seconds disables the tone. This tone can be used for a variety of purposes but the most common will be to trigger a carrier operated amplifier such as a VHF/UHF brick amplifier. The tone duration can be extended to 10 seconds in 1/10 second increments. The tone has a raised cosine leading and trailing edge that is 0.1 second in duration. This prevents generating clicks on the shared spectrum. Leave this control set to zero unless you need the tone for some utility purpose.

MODES

Olivia



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 algorythm 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 indentify 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 QRM, 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.

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

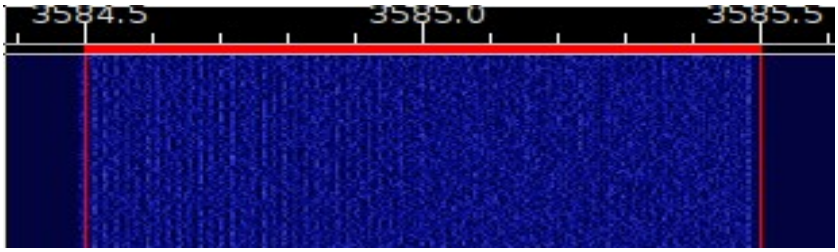
In addition there are two interleaver options (short and long) which can be set on the MT63 configuration tab. The default calling mode is MT63-1000. If the short interleaver is used then one can expect some compromise in robustness. The long interleaver results in somewhat excessive latency (delay between 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 is the only fldigi mode that does not allow random placement 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.



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	Audio Range	Symbol Rate	Character Rate	Interleave / Char.
500 Hz	500 - 1000 Hz	5 baud	5 char / sec	6.4 or 12.8 sec
1000 Hz	500 - 1500 Hz	10 baud	10 char / sec	3.2 or 6.4 sec
2000 Hz	500 - 2500 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 carefull 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 for the basic 1000 Hz mode. 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. MT63 samples at 8000 Hz thus if your card runs at 8000.5 it's probably OK but if it runs at 8005 Hz it's not good ! An extreme example can be a Soundman-16 (PAS-16 clone) which when asked to run 8000 Hz reports that it can only do 8008 Hz and it reality it runs at 7910.3 Hz which makes an error of more than 1% - far too much for the MT63 even at infinitely good S/N.