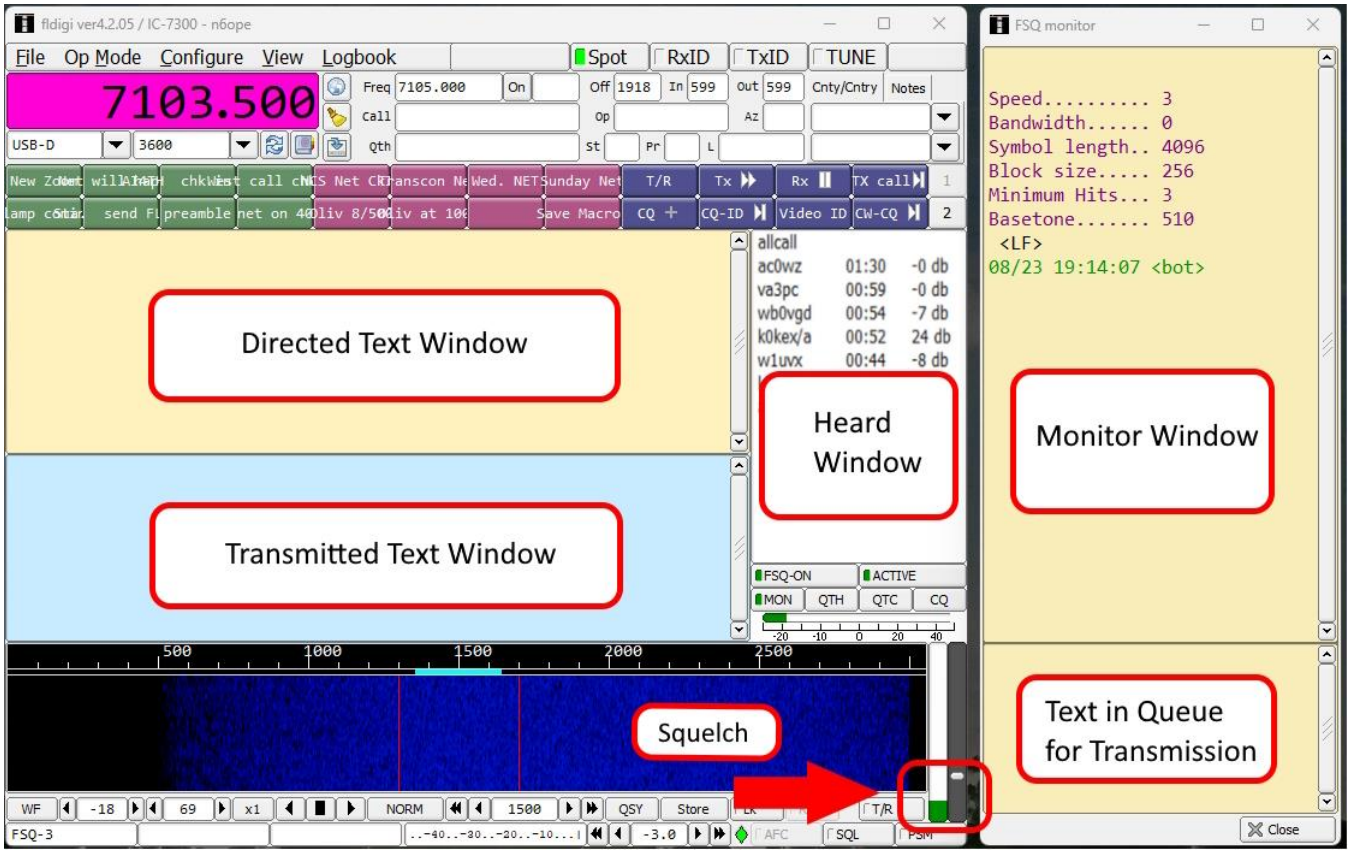
FSQ Command Structure Using FLDIGI

(parts shamelessly copied from various web sources, Updated Nov. 2024, N6OPE)

Additions to this Version (Nov. 2024)

- The puzzling function of the CQ button
- Detailed examples of the FSQ commands with simulated screen shots to show what a sender and a recipient will see in their FLDIGI window and FSQ Monitor window.
- An explanation of the “default”, aka message file, versus a named or general file
- The difference between sending a message using the SPACE command versus the HASHTAG (#) command.
- Best Practices pointers

The FLDIGI Screen



First off . . . The illustration shown above actually shows two windows, side by side. On the left side is the regular, FLDIGI operating window. On the right is the **Monitor Window** which has to be opened separately.

The FLDIGI Window -- Parts & Pieces . . .

The directed text window: (the top, yellow pane) is where

- text directed to you OR to allcall will print *as long as it is decoded*; and
- a copy of text transmitted from your station will print
- displays all decoded text in the UnDirected Mode.

The transmitted text window: (the bottom, blue pane) is the space into which you can type commands and text to be transmitted. *Note*: once text has been transmitted it disappears from this window.

The heard window: displays a list of call signs heard along with the time heard (in UTC time) and their s/n ratio. The list is presented with the most recently heard station on top. When you receive multiple transmissions from the same station the time and s/n information are updated to show the most current data.

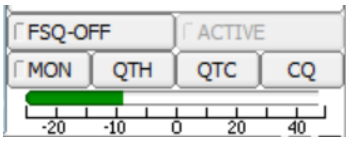
The monitor window: is available in Directed Mode only. It's hidden in Undirected Mode. The monitor window must be turned on manually – click the "MON" button to do that.

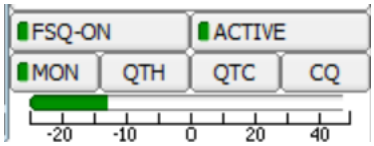
All incoming traffic prints in the top section of the Monitor window whether you are using Directed or Undirected modes. The top half of the Monitor window displays all FSQ activity on the frequency – whether specifically directed to you or not, prints here.

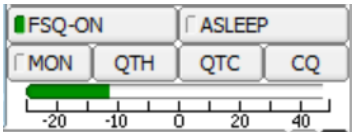
The bottom section of the Monitor window displays text originating from YOUR station that is in queue for transmission. Examples might be a response to the ? or $ commands. Automatic responses aren't sent immediately, but only after a small random delay AND when the squelch is not open.

Note: *The FLDIGI Directed Text Window and the Monitor Window will show the same information for messages that involve your station. However the Monitor Window shows additional information (gibberish if your squelch setting is too low), but also a running time stamp, and traffic that is not specifically directed to your station.*

FSQ Dialog Box







The FSQ Dialog Box is located below the heard list. It:

1. shows a green 'light' to indicate whether
   - FSQ is ON or OFF;
   - Whether FSQ is ACTIVE
   - Whether the Monitor Window (MON) is turned on or off.
2. has buttons for the QTH, QTC, and CQ functions
3. shows a signal to noise indicator

Dialog Box Functions

FSQ-ON

When **ON** transmit and receive are in FSQ directed mode; when **OFF** the TX and RX are in undirected mode.

ACTIVE / ASLEEP

when **ON** the decoder will accept all incoming directed traffic; when **OFF** the decoder will only respond to the wake-up * trigger

MON: this button lights up when on the monitor panel is visible

QTH: sends the myQTH text to the Transmitted Text Window

QTC: sends the myQTC text to the Transmitted Text Window

CQ: 'tis a puzzlement. It notifies other stations that you are on the air and are sending out a CQ but only via the FSQ monitor screen! Nothing appears in a recipient station's Directed Text Window to indicate you are sending a CQ. After running experiments here's what I learned ***from a recipient's point of view***:

- A "CQ" which is transmitted will be picked up by a station if it can hear the transmission
- If a recipient is in **DIRECTED MODE**
  - they **will not** see anything resembling "CQ" in their Directed Text window. What they will see instead is simply "**w1jhk:**" where W1JHK sent out the CQ.
  - They **will see** "**w1jhk:f8cqcqcq"** in their monitor window.

- If a recipient is in **UNDIRECTED MODE**
  - They will see "**w1jhk:f8cqcqcq"** in both their Directed Text and Monitor windows.

Squelch Setting: Turn Squelch on and set the slider slightly above the green bar when there are no transmissions ongoing.

➔ **When the green bar is above the slider, FSQ thinks the frequency is busy and will delay sending your message.**

Definitions:

CRC: A *cyclic redundancy check* is an error-detecting code commonly used in digital networks and storage devices to detect accidental changes to digital data. This will be two characters: two letters, two numbers, or one of each. Your particular station's *crc* will always be the same because it's based on your call sign. Your crc will appear directly after your call sign – separated by a colon – when your station transmits. Examples:

n6ope:3fwa1yzn?	Station n6ope sends the ? command to wa1yzn; n6ope's crc is "3f"
k2rsb:93ab1ph snr= 2db	station k2rsb responds to a ? command from ab1ph; k2rsb's crc is "93"
kd0nbh:da	station kd0nbh sounds, his crc is "da"

Print: ***"Print"*** as used in various manuals for FSQ does not mean a document is being sent to a physical printer; instead it means text will appear in a station's ***directed RX Window.***

Sounding: I think of "sounding" as sort of like sending a 'proof of life'. It's the simple transmission of a call sign and announces to anyone on the frequency that your station is alive. A "sounding" can be done two ways:

1. Tap the " T/R " button in FLDIGI *[Note: any time you tap the T/R button you will "sound" and all you will transmit is a "sound"]*
2. Positioning your cursor in the Transmit window of FLDIGI and tap the ENTER key.

Call Signs: FSQ recognizes three strings of characters as a "callsign"

1. An actual [FCC] licensed call sign, say, w1hkj
2. "allcall" - this text string is recognized as a legit callsign
3. "cqcqcq" - this text string is recognized as a legit callsign

Directed vs. Undirected Mode

Undirected Mode: In **undirected mode**, any text that you type – and send by hitting ENTER on your computer keyboard – is transmitted. It is not addressed to any particular station. Every station within range will print the message. [Meaning the text will appear in their FLDIGI RX window.]

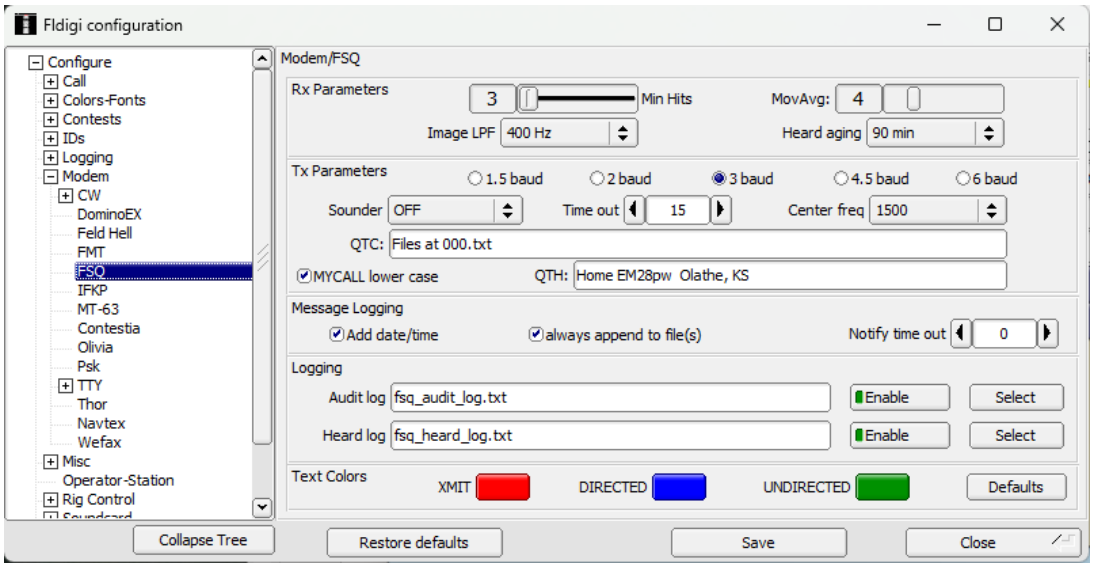
The Undirected mode has no error correction.

Directed Mode: In Directed mode text is sent to a specific call sign or call signs or to Allcall . As a recipient, only text which has been directed to YOUR call sign (or allcall) will print in your FLDIGI RX pane.

FSQ Configuration Settings In FLDIGI

There are two ways to get to the configuration settings for FSQ in the FLDIGI software:

1. From the main FLDIGI menu select Configure > Config Dialog > Modem > FSQ
2. Right click on either the 'QTC' or 'QTH' button in the dialog box at the bottom of the Heard Window



RX Parameters

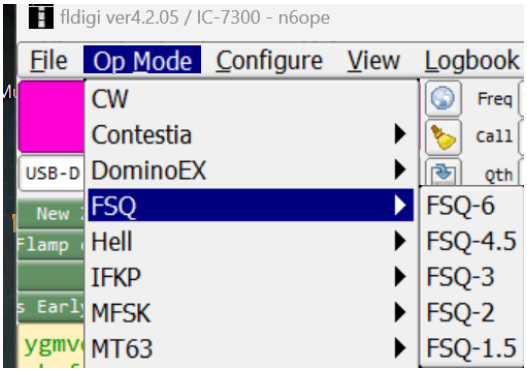
Min Hits: *(From the FLDIGI manual)*: the minimum number of consecutive times a spectral line hits maximum before its designated as a detected signal. Default value = 3.

MovAvg.: *(From the FLDIGI manual)*: the moving average filter is the optimum filter for locating a signal transition. It can be adjusted to average 1 to 4 samples. Setting the MovAvg. To 1 disables it, while the default value is 3.

Image LPF: This control sets a filter to reduce the effect of noise on received images. The FLDIGI manual suggests narrower filters for noise conditions but notes that while filtering will reduce noise in the image it will also blur it.

Heard Aging: Can be set to 'Never', 1 min, 5, 10, 20, 30, 60, 90 or 120 minutes. This parameter sets how long a *heard callsign* remains in your **heard window.** "Never" means they stay in the heard list until you close FLDIGI. What's the "best" setting? It depends on your circumstances. If a mesh net of stations is running fast and furious then 10-20 minutes may be "best". If you are running an unmanned station and just want to monitor occasional soundings and traffic then 90 may be "best". As noted in **Logging** below, FSQ saves a continuous log of all heard stations.

TX Parameters



1.5 to 6 baud -- Transmission speed can be set here as a configuration choice or by selecting an operating modem. Example: you can set up and save an FLDIGI configuration that always starts your use of FSQ at 3 baud.

Sounder: Allows you to choose an interval for your station to automatically sound. Your choices are "OFF", 1 min., 10 min., 30 min, and 60 min.

➢ **Best Practices Recommendation**: *set Sounder to Zero during nets* so sounding from your station doesn't prevent other net traffic from getting on the frequency.

Time Out: Controls how long FSQ will wait if the frequency is busy before it abandons trying to send your message. [Suggestion: 20 seconds.]

Center Freq.: the FSQ Configuration menu as of FLDIGI version 4.2.05 indicates you can set the center frequency [on the waterfall] to 1150, 1500 or 'variable'; ***this is incorrect as w1hkj has confirmed. FSQ will operate only on 1500.***

QTC: A brief message that you can send to other stations OR that other stations can retrieve by invoking a query. This message can be revised / edited / amended at any time and "on the fly". Your

QTC message can be used very effectively to direct other stations to files that are stored on your computer. *(See Embedding File Directory Info in Your QTC Message.)*

QTH: Your location. *NOTE:* will be matched with location listed under Configure > Misc > Operator Station. A change in one will change the other.

➢ **Best Practices Recommendation**: If you want to show your 'home' location, list City, County, State.

However, you can use this space to provide more than a City. For example you could indicate something like: "Red Cross Station A, Wichita, KS".

☑ **MYCALL lower case:** This box should be checked because lower case is faster to send (most characters are sent in one symbol).

☑ **Add date/time:** checking the box will automatically add the date and time to received messages. *(Why wouldn't you want to use this feature in order to keep messages organized?)*

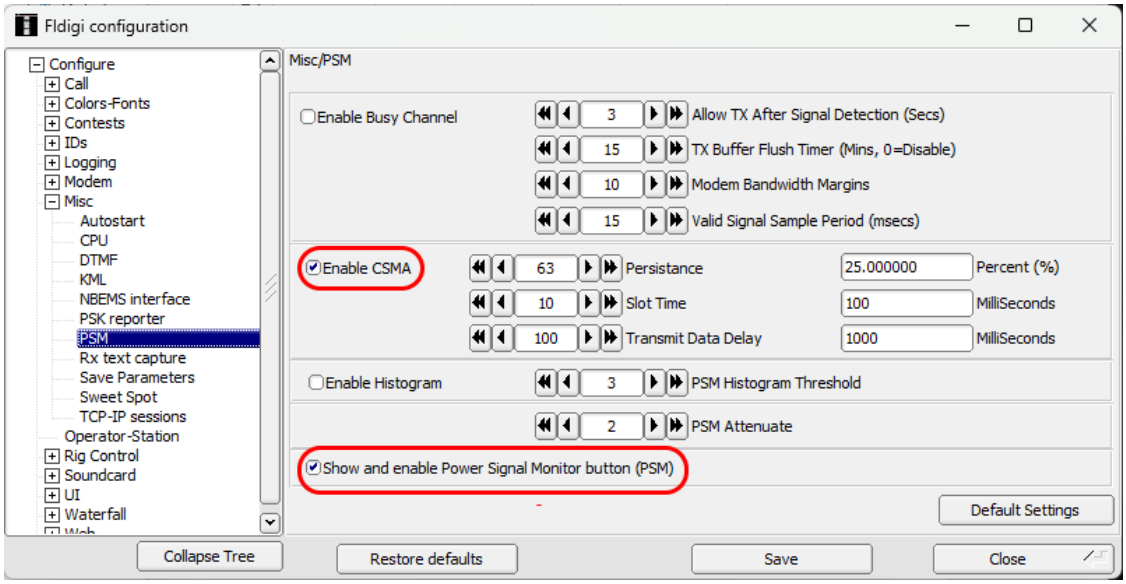
☑ **always append to file(s):** checking the box automatically appends new messages to existing messages from the same station.

(10/23/24 Edit): If you do not check this box, then any new message file you receive from a station will replace the old message file you have from that station.

➢ **Best Practices Recommendation**: Check the box to always append to files.

Notify Time out: allows you to set the time a dialog [alert] window will remain visible. Your choice is 0 to 30 seconds. Leave this control set to zero if you want the dialog (or alert) to remain visible until you dismiss it.

Recommended General FLDIGI Configuration Settings



➢ **Best Practices Recommendation**:

Enable CSMA and the Power Signal Monitor Button in the "Misc" configuration settings of FLDIGI and

turn on PSM by clicking the button at the far bottom right of your FLDIGI window.

(From Wikimedia:) Carrier-sense multiple access (**CSMA**) is a medium access control (MAC) protocol in which a node verifies the absence of other traffic before transmitting on a shared transmission medium, such as an electrical bus or a band of the electromagnetic spectrum.

Logging

Choices here for your Audit Log and Heard Log, enable or not. If you prefer to use other than the default file name you have that option by clicking on the "Select" box. FSQ saves a continuous log of all heard stations and an Audit log of all received data. *(See File Storage - Logging for examples.)*

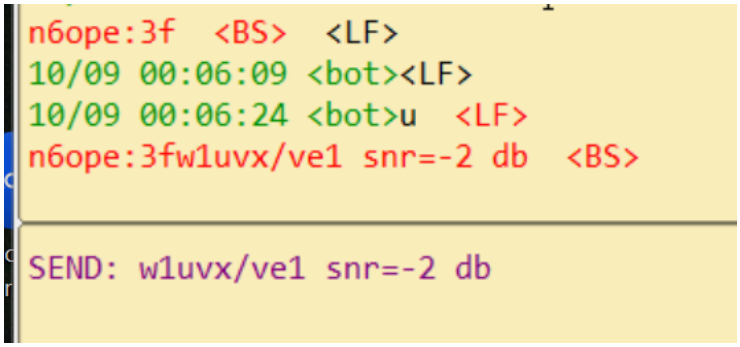
Text Colors – default colors

Text transmitted from your station appears in **RED** in the FSQ Monitor window and the FLDIGI Received window.

Text directed to YOUR station - whether sent to your call sign or to allcall - appears in **BLUE** in the FSQ Monitor window and the FLDIGI Received d window

Text received at YOUR station but which undirected appears in **GREEN** in the FSQ monitor window only. In the directed mode, green is used for time stamps and controls, 'bot' , 'eot'

Traffic that doesn't concern (i.e., wasn't directed to) your station appears in **BLACK** in the FSQ Monitor Window. It *does not* appear in the FLDIGI Directed Text Window.Text that your station is sending out as an automatic response will appear in **PURPLE** in the bottom section (below the horizontal line) in the Monitor Window only. The example to the right shows:



"SEND: w1uvx/ve1 snr=-2 db"

in purple the bottom of the Monitor Window while above it, in red, is the actual traffic which was transmitted: "n6ope:3fw1uvx/ve1 snr=-2db"

This is an example of a query and an automatic response. Station w1uvx/ve1 sent a query to n6ope to ask if they were being heard. Station n6ope sent an automatic response that they were hearing w1uvx/ve1 at a signal to noise ratio of -2 db.

Operating in FSQ

Software Notes & Recommendations[1]

- **FSQ is case sensitive**, so a message directed to (for example) 'W1HKJ' **<u>will not be recognized</u>** by a station when it has set its call sign set to 'w1hkj' in the program configuration settings.

- Lower case is not mandatory for operator call signs, but is encouraged for these reasons:

  1. Lower case is quicker and easier to type.
  2. Lower case is faster to send (most characters are sent in one symbol).
  3. The error rate is half that of upper case characters.

- Call signs that satisfy the parsing requirements will be added to the FSQ heard list.

- ***All FSQ transmissions are designed to be inhibited so long as the software detects a received signal (open squelch).***

- Turn RX ID → **OFF**. This prevents FLDIGI from pulling you off FSQ
- Turn TX ID → **ON**:

Transmitting in FSQ

In FLDIGI an FSQ transmission in is triggered when you either tap the ENTER key or as a result of one of the automatic processes, (such as an automatic reply or acknowledgement). Tapping the "T/R" button at the bottom, right corner of the FLDIGI window <u>will not trigger transmission</u>. Instead it will cause a sounding to be sent. However, you can use the "T/R" button to *stop* a transmission.

"Squelch" Settings in FSQ[2]

Stations can send sentences at any time. In FSQCall (Directed) mode, many of the commands include automated responses, and in addition, Sounding messages may be sent automatically.

[1] Copied directly from the FLDIGI User Manual

[2] Copied from this source: *Tips From* https://amrron.com/wp-content/uploads/2023/02/FSQ-Training-101-1.4.pdf;

So on a busy channel, the potential for transmissions to clash is high. The software operates a Squelch system, which determines (by measuring signal to noise ratio) when the channel is free, and can also determine when to stop printing (therefore limiting junk on the screen). The Squelch is also used to determine when a station may transmit. In non-directed mode, the station will transmit after ENTER has been pressed, only if the Squelch is closed. If it has not closed, the operator must wait until it has and try again.

Some intelligent use must be made of the Squelch control setting, as conditions vary over time. A Squelch setting that allows the transmitter to operate under noisy conditions may also prevent weak signal reception when conditions improve. In Directed mode (the default mode of FSQCall), a conventional network model CSMA protocol decides when the station may have access to the radio channel. This protocol defines three classes of message (chat, responses and soundings), and applies priorities to them. Priority is provided by means of delays after Squelch has closed. In addition, to limit clashes between stations wanting access for the same priority message, random delays are also added.

➢ **Best Practices Recommendation**: Despite all the foregoing, the Great Lakes FSQ net recommends using the PSM[3] (Power Signal Monitor) feature in FLDIGI to attempt to limit doubles and conflicts.

Identifying Your Station

You do not need to preface or end any messages or requests for information sent out from your station with your call sign because FSQ automatically does it for you.

Example: if you wanted to send a message using BPSK (or Olvia or Thor, etc.) you would typically transmit text something like this:

> "de N6OPE, stand by for message transmittal via Olivia 8/500"

In FSQ you would type in the following text in the Transmitted Text Window of FLDIGI and then hit ENTER:

> "allcall pls stand by for msg transmittal via thor 22"

Any stations that can hear you will see this in their Directed Text Window is:

> allcall pls stand by for msg transmittal via thor 22

What will show up in the monitor window is this:

> n6ope:3fallcall pls stand by for msg transmittal via thor 22

[3] The FLDIGI manual's section on PSM can be found here: http://www.w1hkj.com/FldigiHelp/psm_page.html

The syntax FSQ uses is hard to parse at first glance. Something like: **n6ope:3fkd4iz;ab8fj** looks like gibberish. It helps to untangle a string like this when you remember that a colon follows the call sign that has transmitted.

Here are some examples:

What you might see in the monitor window . . .	Sender's call	Syntax	Sender's crc	Command sent to station:	Command	Recipient station
wa1yzn:0d	**wa1yzn**	**:**	**0d**			
ac0wz:e8kd8wck+[01]	**ac0wz**	**:**	**3f**	**kd8wck**	**+**	
n6ope:3fkd4iz;ab8fj	**n6ope**	**:**	**3f**	**kd4iz**	**;**	**ab8fj**

FSQ Message Files: Default Files, aka "Message Files" versus "Named" Files

The FSQ function in FLDIGI has the capability of reading and writing files on another station's computer.[4] The original authors set up the software to use or recognize "default" files versus what I'll call "named" files.

I'm using the term "named" file in this paper as a way to think about how it's content might be distinguished from a file that carries a callsign name. Something like:

- "Red Cross Instruct.txt", or
- "ARES_FQs.txt", or
- "Op Procedures.txt"

However any sort of content can be sent by either method and a 'callsign.txt' can certainly be sent to any other station. Let's say AB8FJ wants to send a copy of the "messages" he's received from W3WT to KD8WCK. He would type:

Kd8wckt#[w3wt.txt]

My understanding is that FSQ's authors intended for it to be a simple system to use: a simple way for stations to exchange information. When sending a default or message file the sender doesn't have to take time to create or name a file: they need only type some sort of content (i.e., message) and their communication is automatically given the file name "sender.txt" where "sender" is the sender's call sign.

[4] The original version of FSQ also had the ability to *delete* default files. This function wasn't included in the U.S. version of FSQCall or in FLDIGI.

***IMPORTANT NOTE*:** both the stand alone FSQ Call software and FLDIGI[5] use the term "message" instead of "default file". Both have functions in their drop-down menu for "Send a message to … (#)" and "Read a Message from … (+)" However, what's actually happening is that the sender is pushing a message to the recipient's system. How that message content is saved depends on configuration choices of the recipient. (See Always Append to File.)

Assuming a transmittal decodes properly a recipient(s) has no choice about whether or not a file that's sent to them is saved on their computer: it will be.

It's also important to know that there is a separate and different method for sending a message that does not involve sending a file. See below: Sending a Message via Space Command.

Two Important Notes:

1. By default, FLDIGI stores files saved in a station's C:\Users\[Name]\fldigi.files\temp directory.[6]
2. **FSQ recognizes only one type of file extension ' .txt ' as a message file.**

Embedding File Directory Info in Your QTC Message

Any station can request a 'print' of your QTC message. This feature can be used very effectively to point other stations to information (files) stored on your computer that might be of interest to them. Examples could be things like:

a. Descriptive list of files available at your station with their file names
b. Operating frequencies / repeaters being used by local / relevant em comm groups
c. Operating structure for a net, i.e. default frequencies, check in times, etc.
d. Addresses for emergency shelters in your community
e. Status of emergency shelters – which are open, which are full

Here's how that can work:

My QTC message currently reads: "File List available at 000.txt"

If another station reads my message saved as "000.txt" they'll get the following text:

```
000.txt     this file     3 lines
001.txt     Kansas City em comm Repeaters    4 lines
002.txt     transcon net instruct    4 lines
```

[5] My understanding is that FLDIGI's authors simply adopted the command structure and functionality of FSQ Call and incorporated it into their software. In other words, FSQ Call existed first and FLDIGI copied it.
[6] FLDIGI assumes the file is located in C:\Users\[Name]\fldigi.files\temp however a user can always navigate to a different directory or folder and select a file from it.

If they then want to see the list of Kansas City em comm repeaters they can retrieve, (i.e., 'read') my file, "002.txt".

Notes: It's your station – you control these 'information' files. Some files could essentially be static – such a list of local repeaters. Others might be subject to updates – like a list of community emergency shelters that are currently open.

There is a potential problem with this scheme, however. You have no control over whether another station pushes (i.e., sends) a file to you or to all call with the hashtag (#) command. If you have a file in your system saved as "dir.txt" it can be appended if another station sends you a file with that same name. This should give you pause as an operator as to what you name a file you send (push) to another station and whether you should select the option to always append incoming files to existing files in the FSQ Configuration set up.

Trigger Commands

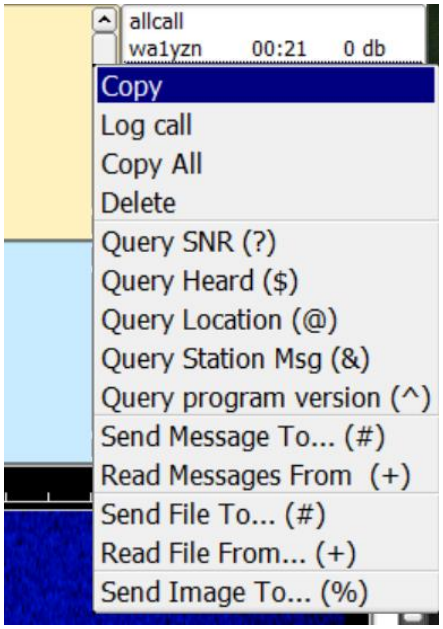
FSQ uses a single character - a punctuation mark - to execute a command. These characters are:

? * ! ~ ; # % + @ & $ < > ^ | and **SPACE**

1. You **MUST** use a callsign in the message body of every sentence for the other station to see a message.
2. The callsign *must be in all lower case letters*
3. The callsign must be followed **immediately** by a trigger character or the command will be ignored. Remember: a space that follows a call sign is considered a trigger.
4. All stations within a network must have unique callsigns.
5. Do not give two commands to the same callsign within the same sentence. If you do, the second command will be ignored.

The FLDIGI Drop Down Menu for Commands

Any FSQ command may be typed by hand in the blue 'Transmitted Text Window' and executed by simply hitting the ENTER key on your keyboard. But FLDIGI offers a short cut way to send ten commands (and which eliminates at least some typographical errors). Access it by **right clicking** on any callsign in your heard list and a list menu of commands will pop up. (And remember 'allcall' is just like any other call sign.)



Important note: The authors of FLDIGI have set up the drop down menu so that sending a "message" and sending a "file" use the same command: the hashtag (#). (Similarly, reading a "message" and reading a "file" both use the plus (+) command.) But in reality there is essentially no difference between a "message" and a "file".

A "message" as used in these commands is a file. It's a file that ends up saved in the recipient's computer and by default it is named with the call sign of the sender and has a .txt file extension. A "file" simply has some other name and ends with .txt.

Query Commands: if you click on one of the five Query commands in the menu FLDIGI will execute the command and that's all you need to do.

Send / Read Message Commands:

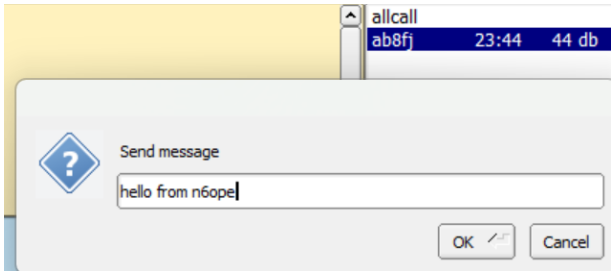
HASHTAG (#) Command: The hashtag command sends a file to another station or stations. But it's more than 'sending' a file; it ***pushes*** a file, causing it to be stored in the recipient(s)' computer whether the recipient wants it or not. *(See always append to files.)*

The drop down menu in FLDIGI gives you the choice of 'sending a message' and 'sending a file' using the hashtag command.

To send a 'message' to another station (or to everyone using allcall) using the drop down menu:

1. **Right click** on any callsign in your heard list and a list menu of commands will pop up.

2. Click on "**Send Message To … (#)**" and a dialog box will pop up on your computer screen that looks like this:



3. Simply type the text of your message in the box and click the "OK" button to send it off. What's actually happening here is that FSQ understands or assumes that you are sending a file with the default name "callsign.txt" where "callsign" is the sender's FCC call sign.

NOTE: sending a "message" with the hashtag (#) command will result in two actions.

a. *First: it will cause a file named [callsign.txt] to be created or appended to in the in recipient's temp file, where 'callsign' is the call sign of the sender. (See file storage)*
b. *As long as the transmission is decoded properly the recipient's station (or recipients' if a message is sent to allcall) will automatically send an acknowledgement back to the sender, (i.e., an "ack").*

Users should be aware that there is an alternative, somewhat simpler way to send a message to another station or to allcall. That's with the SPACE command (SpaceCmd.) However, the FLDIGI drop down menu doesn't include this function – it has to be hand-typed.

Sending a "Message" via the SPACE command versus the HASHTAG command

Whether to send information or an instruction via the SPACE command versus the HASHTAG command depends on the importance of its content, whether one needs a "permanent" record of the transmission, and whether or not a series of acknowledgments (an "ack storm") could be an annoying consequence of the choice. Messages sent with the SPACE command print in the recipients' Directed Text Window but are not acknowledged automatically.

A good example of a circumstance when one might use the SPACE command is to advise other stations to change frequencies. That command would simply be:

allcall qsy to 20m

To Send 'File' To (#) another station using the drop down menu in FLDIGI:

1. Right click on either an individual callsign in your heard list or allcall
2. Click on "Send File To… (#)" from the drop down menu
3. In a Windows system FLDIGI opens up your C:\Users \(name)\fldigi.files\Temp directory by default so you can simply click on one of the files listed there. See the File Storage section later in this document for more details on file storage.

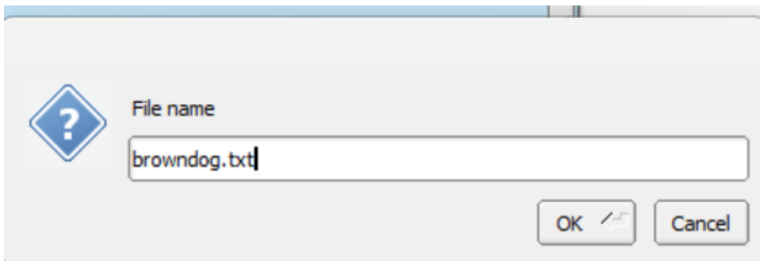
Note: you can always navigate to another directory to select a target file but it must be a .txt type file!

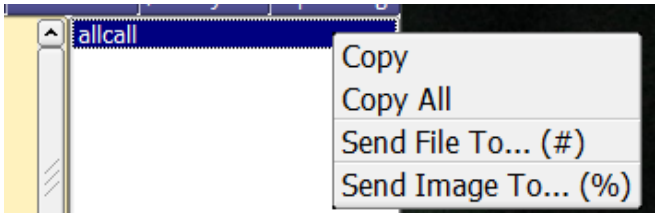
Read Message From (+)

1. Right click on an individual callsign in your heard list (you cannot retrieve or read a message from allcall). Let's say for example you want to read the message file from station w1hkj.
2. Click on "Read Message From (+)" from the drop down list; the command is executed.
3. FLDIGI *automatically* knows that you want to retrieve and read the message file stored in your computer as "w1hkj.txt".

Read File From (+)

1. Right click on an individual callsign in your heard list
2. Click on “Read File From . . . (+)” from the FLDIGI drop down menu
3. Type the name of the file in the box that opens. **NOTE:** You MUST include the file extension ‘.txt’. If you do not you’ll get a return message from the target station that the file you wanted is “not found”.





Since there is no ‘QTC’ or ‘QTH’ for allcall nor does allcall hold any files, the commands available for ‘allcall’ as a callsign are much more limited. However they work the same way as commands sent to an individual station

FSQ Command Summary

- For the following: "message" is a string of text characters. It could be '*happy birthday'*, it could be '*stand by for message transmittal via FLAMP using Thor 22 at 2000 on the waterfall'*
- "Callsign" is either (a) a standard ham radio call sign **OR,** (b) 'allcall'
- "Printing" means that text will appear in the Directed Text Window of the recipient station(s)
- target_call means the call sign of the recipient of a command
- **format** is used with image transfers only. It can have one of three values: **'S'** (small), **'L'** (large), or **'F'** (fax)

Commands that retrieve information from another specific station *(as long as the station hears and decodes your signal)*	
Trigger Command	**Command Result**
callsign? (? example)	Causes station **callsign** to automatically send a signal report
callsign@ (@ example)	retrieves / prints the QTH message stored by station **callsign**
callsign& (& example)	retrieves/prints the QTC message stored by station **callsign**
**callsign$** ($ example)	retrieves/prints the [entire] heard list from station **callsign**
callsign$n ($n example)	retrieves the latest 'n' number of entries from station **callsign's** heard list

Commands Related to Message Exchanges	
callsign message OR allcall message OR cqcqcq message (example)	**message** prints at station **callsign** (or at all stations that hear you). R*ead this as 'callsign'[space]message,* where the space functions as a command. **IMPORTANT NOTE:** This method of sending a message functions sort of like "chatting". Your transmission WILL NOT be automatically acknowledged by a receiving station(s).
callsign!message (! example)	Causes station **callsign** to retransmit **message** back to you
callsign~message (~ example)	Causes station **callsign** to retransmit **message** back to you after a delay
callsign;target_call message (; example)	Causes station **callsign** to *relay* **message** to the **target callsign** and relays back a response from them

Commands Related to Message Exchanges	
callsign#[aaaa]message OR allcall#[aaaa]message (# example)	Causes station **callsign** OR **allcall** stations to *store* a **message** as (1) a text file named 'aaaa.txt', OR (2) if a file named 'aaaa.txt' already exists, it will append **message** to that existing file if you have selected that option in FSQ configuration settings. *(see Always Append to Files)* ***Note: this can be thought of as "pushing" a message to other call signs. They have no choice to accept it or not.*** The stations that hear you and properly decode your transmission will automatically send back an acknowledgement.
callsign+ (+ example)	reads a *default* file named 'sender.txt' which is stored on **callsign's** computer and where 'sender' is the sender's callsign
callsign+[bbbb.txt] (+ example)	reads a file named 'bbbb.txt' which is stored on **callsign's** computer
callsign|message (| example)	causes an alert box to display with '**message**' at station **callsign.** *Note: this Alert box will stay on **callsign's** screen until that station closes it or until another, newer alert is sent to the station.* **Note:** if you want an 'Alert' message to stay on your computer screen until you dismiss it then you should set the "Notify Time Out" counter to zero in the 'message logging' section of configuration settings for FSQ. *(see Notify Time Out)*
callsign% format OR allcall%format	Sends an image (saved in sender's directory) to **callsign** OR all stations that hear you in format, 'S', 'L', or 'F' *Note: the command syntax calls for a space after '%' but before 'S', 'L', or 'F'*

Miscellaneous Commands	
callsign* (* example)	Turns FSQ to ACTIVE state if it has been asleep at station **callsign**
callsign> (> example)	causes station **callsign** to move to a faster speed and confirms
callsign< (< example)	causes station **callsign** to move to a shower speed and confirms
callsign^ (^ example)	causes station **callsign** to report its FSQCall software version

Commands – Detailed Examples

The drop down 'command' menu available in FLDIGI is very convenient, but the following examples illustrate what could – alternatively – be typed in the blue Transmitted Text Window of FLDIGI in order to execute a command. And, in some cases, these commands aren't available alternatives in that drop down menu.

Remember: to execute a command you type text and then hit the ENTER key. As soon as the command executes it disappears from the blue Transmit Text window. The following illustrations show you what to expect to see in the FLDIGI window and the Monitor Window for both the sender and recipient after the command executes. In the SENDER's FLDIGI Directed Text Window, (the yellow FLDIGI pane), you'll see the outgoing command in **red** text.

Sounding

'Sounding' is not a command but is still one of the most common transmissions one sends in FSQ. For this example, N6OPE is sounding.

Text entered in Transmit Text Window: *nothing, to sound just position your cursor in the blue pane and hit ENTER.* (There's nothing in the Directed Text (yellow) FLDIGI window because there isn't any text entered in the Transmit Text window when you sound.)

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> **n6ope:3f <BS>**	

Any stations capable of receiving her station see the following:

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> **10/20 17:45:33 <bot> n6ope:3f <BS><eot>**	

? Command – Asks another station if they are hearing the sender's station

N6OPE asks station AB1PH if he can hear her station.

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> **n6ope:3fab1ph? <BS>**	**ab1ph?**

AB1PH's response (generated and transmitted automatically).

Note: the colors in the Recipient's (AB1PH's) FLDIGI window. N6OPE sends a command directed to AB1PH. That shows up in **blue.** The text returned [automatically] by AB1PH's station shows up in **red**.

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> **10/20 18:27:35 <bot>n6ope:3fab1ph? <BS> <eot>** **.sf <LF>** **ab1ph:efn6ope snr=8 db <BS> <LF>**	**n6ope?** **n6ope snr=8 db**

@ - QTH Query

AC0WZ asks station WB0VGD for their location (QTH info). WB0VGD's station automatically sends a response. *Note: your QTH doesn't <u>have</u> to be a city/state; it can be descriptive.)*

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> ac0wz:e8wb0vgd@ <BS> <LF> 10/20 19:16:11 <bot>wb0vgd:72n6ope Red Cross Stn; City Center, Omaha, NE <BS><eot>	wb0vgd@ wb0vgd: Red Cross Stn; City Center, Omaha, NE

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
wap <LF> 10/20 19:15:58 <bot> ac0wze:72wb0vgd@ <BS> <eot> ui <LF> wb0vgd:72n6ope Red Cross Stn; City Center, Omaha, NE <BS>	ac0wz:@ ac0wz Red Cross Stn; City Center, Omaha, NE

& - QTC Query

N6OPE asks station WA1YZN for their QTC message. WA1YZN's station automatically sends a response.

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> n6ope:3fwa1yzn& <BS> frp <LF> 10/20 19:16:11 <bot>wa1yzn:0dn6ope see file 000.txt for stored file list <BS><eot>	wa1yzn@ wa1yzn: see file 000.txt for stored file list

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
wap <LF> 10/20 19:15:58 <bot> n6ope:3fwa1yzn@ <BS> <eot> ui <LF> wa1yzn:0dn6ope see file 000.txt for stored file list <BS>	n6ope:@ n6ope see file 000.txt for stored file list

$ command -- Request a station's heard List

KD4IZ asks N6OPE for their station's heard list

SENDER's Monitor Window		SENDER's FLDIGI Window
kd4iz:7dn6ope$ <BS>		n6ope$

N6OPE station responds <u>automatically</u>:

RECIPIENT's Monitor Window		RECIPIENT's FLDIGI Window
10/20 18:16:47 <bot>kd4iz:7d6ope$ <BS> <eot> <LF> n6ope:3fkd4iz Heard:<LF> kd4iz, 18:10, 8 db<LF> kd8wck, 18:16, 3 db<LF> ac0wz, 18:17, 15 db<LF> wa1yzn, 18:21, -3 db<LF> w3wt, 18:22, -2 db<LF> k0kex/a, 18:28, 20 db <BS>		kd4iz:$ kd4iz Heard: kd4iz, 18:10, 8 db kd8wck, 18:16, 3 db ac0wz, 18:17, 15 db wa1yzn, 18:21, -3 db w3wt, 18:22, -2 db k0kex/a, 18:28, 20 db

$n command -- Request the last 'n' number of stations in a heard List

This command works just like the '$' command except it asks for a list of the last (most recent) 'n' stations heard by the recipient. For this example KD4IZ asks N6OPE for a list of the last 3 stations in their list.

SENDER's Monitor Window		SENDER's FLDIGI Window
kd4iz:7dn6ope$5 <BS>		n6ope$5

N6OPE station responds <u>automatically</u>:

RECIPIENT's Monitor Window		RECIPIENT's FLDIGI Window
10/20 18:16:47 <bot>kd4iz:7d6ope$5 <BS> <eot> <LF> n6ope:3fkd4iz Heard:<LF> wa1yzn, 18:21, -3 db<LF> w3wt, 18:22, -2 db<LF> k0kex/a, 18:28, 20 db <BS>		kd4iz:$5 kd4iz Heard: wa1yzn, 18:21, -3 db w3wt, 18:22, -2 db k0kex/a, 18:28, 20 db

Space command

For an explanation of the difference between sending a message to another station (or stations) please refer back to Sending a Message via the SPACE command versus the HASHTAG Command

VA3PC sends a message out to a specific station – N6OPE - using the **SPACE** command

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> va3pc:46n6ope today is saturday <BS>	n6ope today is saturday

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 18:10:47 <bot>va3pc:46n6ope today is saturday <BS><eot>	va3pc: today is saturday

! Command - asks station to repeat a message

In this example K2RSB asks KD8WCK to repeat a message. The message is: "repeat this message back to me"

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> k2rsb:93kd8wck!repeat this message back to me <BS> np <LF> 10/20 18:52:30 <bot>kd8wck:ee repeat this message back to me<BS> <eot>	kd8wck!repeat this message back to me

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 18:21:46 <bot>k2rsb:93kd8wck!repeat this back to me <BS><eot> <LF> kd8wcke:ee repeat this back to me <BS>	kd8wck: repeat this back to me repeat this back to me
SEND: repeat this back to me	

~ Command -- Repeat Message with Delay

This command has the same function as the ! (exclamation) command above, however a short delay is automatically

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> k2rsb:93kd8wck!~repeat this message with a delay <BS> mp <LF> 10/20 19:13:40 <bot> kd8wcke:ee repeat this message with a delay<BS><eot>	kd8wck!repeat this message with a delay

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 19:13:41 <bot> k2rsb:93kd8wck ~repeat this back to me <BS><eot> <LF> kd8wcke:ee repeat this back to me <BS>	kd8wck: repeat this message with a delay repeat this message with a delay
SEND: repeat this back to me	

; Command – Relay a Message

In this example the sender (N6OPE) uses the relay command (;) to tell station KD4IZ to relay a message to a third station (AB8FJ)

SENDER's Monitor Window	SENDER's FLDIGI Window
n6ope:3fkd4iz;ab8fj today is Sunday <BS> np <LF> 10/20 18:59:06 <bot> kd4iz:7dab8fj[n6ope] today is Sunday <BS> <eot>	kd4iz;ab8fj today is Sunday

Recipient's station (AB8FJ) sends automatic reply

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 18:58:48 <bot>n6ope:3fkd4iz;ab8fj Today is Sunday <BS><eot> <LF> kd4iz:7dab8fj[n6ope] today is sunday	n6ope:;ab8fj today is Sunday ab8fj[n6ope] today is Sunday
SEND: ab8fj[n6ope] today is sunday	

Command – Send a File (message)

The hashtag (#) command is used to send (actually 'push') a file to another station or stations. Again: a recipient has no choice: any file that decodes properly *will be* saved on his system.

To compare the commands for sending a default or message file versus a named file lets assume:

- AB8FJ wants to send a message to W3WT saying, "**QSY to 20m**"
- AB8FJ has a file named "ops_proc.txt" saved on his system that says:

  "**During Nov–Mar we operate on 40m, other months on 20m**"

To send the message or default file AB8FJ would type:

w3wt#qsy to 20m

To send the 'named' file, AB8FJ would type:

w3wt#[ops_proc.txt]

In this example K2RSB sends a "message" file to WB0VGD.

SENDER's Monitor Window		SENDER's FLDIGI Window
<LF> k2rsb:93wb0vgd# thought of the day<BS> <LF> 10/20 20:57:23 <bot>wb0vgd:72k2rsb ack <BS><eot>		wb0vgd# thought of the day

This file will be saved on WB0VGD's system (or appended to an existing file) named: **k2rsb.txt.**

As long as they can decode the transmission property the recipient station(s) will automatically send an acknowledgement they received the file.

RECIPIENT's Monitor Window		RECIPIENT's FLDIGI Window
<LF> 10/20 20:57:06 <bot>k2rsb:93wb0vgd# thought of the day <BS><eot> <LF> wb0vgd:72k2rsb ack <BS>		k2rsb:# thought of the day k2rsb ack
SEND: k2rsb ack		

Send ["Named"] File using Hashtag (#) Command

In the following example AB8FJ is sending a file named 'score.txt' to KD4IZ. (The syntax would be the same if he sent it to allcall.) The contexts of the file are: "Four score and twenty years ago"

Sender's Monitor Window	Sender's FLDIGI Window
qp <LF> ab8fj:9dkd4iz#[score.txt]<LF> Four score and twenty years ago <BS> <LF> 11/14 00:23:35 <bot>kd4iz:7dab9fj ack <BS><eot>	kd4iz#[score.txt] Four score and twenty years ago kd4iz: ack

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> ab8fj:9dkd4iz#[score.txt]<LF> Four score and twenty years ago <LF> <LF> <eol> <BS><eot> <LF> kd4iz:7dab9fj ack <BS>	ab8fj:[score.txt] Four score and twenty years ago ab8fj: ack
SEND: ab8fj ack	

Any/all recipients of the file will end up with a file (or appended to a file) named **ab8fj.txt**.

+ Command – Read a Message

Reading a [default] Message

In this example N6OPE tries to read the default file named n6ope.txt stored on w1hkj's computer. There is NO file with that name on w1hkj's system so the automatic message returned by his station is "not found".

Sender's Monitor Window	Sender's FLDIGI Window
<LF> 10/20 20:41:57 <BS>n6ope:3fw1hkj+ <BS><eot> <LF> w1hkj:efn6ope not found <BS>	n6ope not found

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> n6ope:3fw1hkj+ <BS> <LF> 10/20 20:52:10 <bot>w1hkj:efn6ope not found <BS><eot>	w1hkj+ w1hkj: not found

In this example w1hkj is retrieving a file named 'w1hkj.txt' which is stored on n6ope's system. The contents of the file are:

Received 20241020, 210203
Second thought

Sender's Monitor Window	Sender's FLDIGI Window
<LF> w1hkj:efn6ope+ <BS> dp <LF> 10/20 21:05:37 <bot> n6ope:3fw1hkj [w1hkj.txt]<LF> Received 20241020, 210203 <LF> <eol>second thought<LF> <LF> <eol> <BS> <eot> <LF>	n6ope+ n6ope:[w1hkj.txt] Received 20241020, 210203 second thought

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 21:05:27 <bot>w1hkj:efn6ope+ <BS><eot> <LF> n6ope:3fw1hkj [w1hkj.txt]<LF> second thought<LF> <LF> <BS>	w1hkj [w1hkj.txt] Received 20241020, 210203 Second thought
SEND: w1hkj [w1hkj.txt] Received 20241020, 210203 Second thought	

Reading a General or 'Named' Message File

In this example W3WT is retrieving a file named 'browndog.txt' which is stored on VA3PC's system. The contents of the file are: "jumped over the lazy fox"

Sender's Monitor Window	Sender's FLDIGI Window
<LF> W3wt8eva3pc+[browndog.txt] <BS> <LF> 11/18 22:56:41 <bot> va3pc:46w3wt [browndog.txt]<LF> jumped over the lazy fox <BS> <eot>	va3pc+[browndog.txt] va3pc: [browndog.txt] jumped over the lazy fox

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 21:05:27 <bot>w3wt:8eva3pc+[browndog.txt] <BS><eot> <LF> va3pc:46w3wt [browndog.txt] <LF> jumped over the lazy fox <BS>	w1hkj [w1hkj.txt] jumped over the lazy fox
SEND: w3wt [browndog.txt] jumped over the lazy fox	

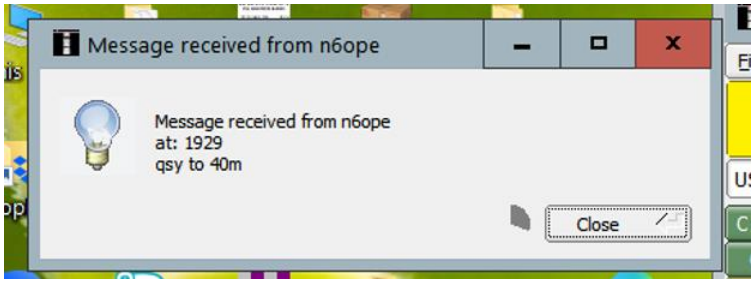
| Send Alert Command

In this example N6OPE sends an alert message to AC0WZ

SENDER's Monitor Window	SENDER's FLDIGI Window
<LF> n6ope:3fac0wz|qsy to 40m <BS>	ac0wz|qsy to 40m

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
Rqp <LF> 10/20 19:28:58 <bot>n6ope:3fac0wz|qsy to 40m <BS> <eot> Message received from n6ope at 19:29 qsy to 40m	n6ope|QSY to 40m

An alert message box like the one below will pop up on the receipient's computer screen. (Probably toward the top, left corner.) How long it remains there depends on the recipient's configuration settings. (See Notify Time Out)



*Command – Turns another Station's FSQ to ACTIVE if it has been asleep

Note: the target station has to have FSQ turned ON. Sender is K6YXH and he executes the command to WB0VGD to go from ASLEEP to ACTIVE.

Sender's Monitor Window	Sender's FLDIGI Window
11/14:01:10:20 <bot>k67xh:86wb0vgd* <BS><eot> <LF> wb0vgd:72k6yxh 4.5 baud <BS> <LF>	

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
11/14 01:10:26 <bot>k6yxh:86wb0vgd* <BS><eot> <LF> wb0vgd:k6yxh ack<LF>	<LF> k6yxh:86wb0vgd* <BS> k6yxh ack

> Command – Causes Target Station to Move to Faster Speed

K6YXH sends a command to increase transmit speed to station WB0VGD. WB0VGD's speed increases from 3.0 to 4.5.

Sender's Monitor Window	Sender's FLDIGI Window
<LF> k6yxh:86wb0vgd> <BS> <LF>	k6yxh>

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
11/14 01:02:47 <bot>k6yxh:86wb0vgd< <BS><eot> wb0vgd:72k6yxh< <BS> <LF> 11/14 01:03:41 <bot>k6yxh:86wb0vgd 2.0 baud	k6yxh< k6yxh: 2.0 baud

< Command – Causes Target Station to Move to Slower Speed

Sender's Monitor Window	Sender's FLDIGI Window
<LF> k6yxh:86wb0vgd< <BS> <LF>	wb0vgd<

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
11/14 01:02:47 <bot>k6yxh:86wb0vgd <BS><eot> wb0vgd:72k6yxh< <BS> <LF> 11/14 01:03:41 <bot>k6yxh:86wb0vgd 2.0 baud	k6yxh k6yxh: 2.0 baud

^ Command - Send request for Station's Version of FLDIGI

Here, W1HJK asks N6OPE what version of FLDIGI she's using.

Sender's Monitor Window	Sender's FLDIGI Window
<LF> w1hkj:efn6ope^r <BS> rp <LF> 10/20 20:43:45 <bot>n6ope3f:w1hkj fldigi 4.2.05 <BS><eot>	n6ope^r n6ope:fldigi 4.2.05

RECIPIENT's Monitor Window	RECIPIENT's FLDIGI Window
<LF> 10/20 20:46:46 <bot>w1hkj:efn6ope^r <BS><eot> <LF> n6ope:3fw1hkj fldigi 4.2.05 <BS>	w1hkj:^r w1hkj fldigi 4.2.05

Sending Images Via FSQ

You can send images via FSQ. The process is very similar to how you might send an image file using the MFSK modem in FLDIGI.

I have nothing to add to, nor can I improve upon the instructions written up by Murray Greenman and Con Wassilieff which can be found here:

https://www.whitemesa.net/fsqcal/doc/FSQCAL_024_6_US/IMAGEhelp.htm

However these instructions were written for the separate, stand-alone software, FSQCall which isn't exactly functionally the same as using FSQ in FLDIGI. Most particularly I note that in this paper the authors state:

> *Note: The recipient callsign for image transmission can be a single callsign, such as zl2abc%, a list of callsigns (such as zl2abc% zl2cde% zl3fab%), and even allcall% can be used to send the image to everyone.*

Based on tests in FLDIGI, *I don't think* that a list of users can be "chained" together in order to send an image (or a file) to multiple stations. *I think* that you have only two choices: sending to a single station or sending to allcall. *(N6OPE, Oct. 2024.)*

➢ **Best Practices Recommendation**: According to the folks who participate in the weekly Great Lakes FSQ net you probably need a SNR of 10 or better with another station to reliably receive images sent via FSQ on JF.

File Storage - Logging

FLDIGI's FSQ saves a continuous log of all Heard stations, and an Audit log that includes all received data. These logs are named **fsq_heard_log.txt** and **fsq_audit_log.txt** respectively. The logs are never cleared unless the operator does so manually. Data is appended to each log as it occurs. These logs are located in the FLDIGI temp folder:

XP	C:\Documents and Settings\<username>\fldigi.files\temp
Vista	C:\Users\<username>\fldigi.files\temp
Win7, 8, 10, 11	C:\Users\<username>\fldigi.files\temp *Note: If you have set up multiple configurations for FLDIGI in a Windows system you will need to go to the specific directory you've set up for the FLDIGI config specialized for FSQ communications. For the author, that directory is named:* C:\Users\<username>\fldigi_FSQ.files\temp.

	Be aware that if you occasionally use FSQ in your “other” or “general” FLDIGI config then the ‘temp’ directory for that configuration will contain FSQ Audit and Heard logs.
Linux	/home/<user>/.fldigi/temp
OS X	/home/<user>/.fldigi/temp
Unix	/home/<user>/.fldigi/temp

You can elect to disable the logging functions. You may also rename the log to suit your particular operations.

Audit Log Example

Your Audit Log captures everything that you would see passing through your Monitor Window – which can and will include a great deal of noise gibberish. In the sample below I’ve highlighted actual traffic / commands with **bold** text.

```
================================================
Audit log: 20240205, 031058
================================================
 khhcmpMuagePhmlapmbopatvvttxtosho. rhip <LF>
k0rgb:bc  <BS>  qo sp <LF>
kd0nbh:da  <BS> =opwxoykp <LF>
w0nrp:28  <BS>  xpeqvdzpxw]g.hbrwlimbbfrqnup <LF>
w0kcn:f0  <BS>  kpopot.uhb<LF>
^rejhrbidpoqheu.or.aioeehkmikghnqwzovbey>yvu<LF>
<BS>latzsoclcuwop` en pfkrp <LF>
w0kcn:f0n0mtv&  <BS>  mdp <LF>
n0mtv:b0w0kcn n0mtv  no traffic  <BS>  ipopop.si:pamzcc<LF>
muuophzweoqnmruyqxxonvp <LF>
w0nrp:28cqcqcq  <BS>  poSnp@dKVhxakgjfete<LF>
ychapovwwp <LF>
```

Heard Log Example

These were the last (bottom) four entries in my heard log one afternoon. They form the record for 8/16/2024 (one entry), 8/20/24 (two entries), and 8/21/2024. I *think* the date and time in the “header” entry for the log reflect the time that my station was tuned to FSQ.

- On Aug. 16th I apparently started FSQ at 00:23:22 UTC. I heard only one other station and that was at 00:43:35. (Amazingly enough a station in China = bh1fj.)
- On Aug. 20th I must have started FSQ at two different times. Once at 00:02:24 UTC, and again about an hour later at 00:03:35. No stations heard
- On Aug. 21st I started up FSQ at 00:04:17 and heard four other stations (one twice) between 00:41:53 and 00:44:50 UTC.

==========
Heard log: 20240816, 002322
==========
20240816,004335,bh1fj,-7 db
==========
Heard log: 20240820, 000224
==========
==========
Heard log: 20240820, 000335
==========
==========
Heard log: 20240821, 004117
==========
20240821,004153,kd4iz,10 db
20240821,004223,wa1yzn,12 db
20240821,004251,ab1ph,-4 db
20240821,004333,ab1ph,-3 db
20240821,004450,va3pc,11 db

Resources

https://www.whitemesa.net/fsqcal/doc/FSQCAL_024_6_US/Syntax.pdf

http://www.w1hkj.com/FldigiHelp/fsq_page.html *(This is the section of the FLDIGI manual for FSQ.)*

http://fsqcall.org/