QMX

QMX: QRP Labs Multimode Xcvr (transceiver)

Virtual U3S operating manual, firmware 1_04_008, 27-Aug-2026

For ALL QMX-series transceivers

Contents

1. Introduction

The Virtual U3S feature in QMX incorporates the whole functionality of the real Ultimate3S QRSS/WSPR/Weak signal modes standalone beacon kit, designed by QRP Labs in early 2015 and produced continuously since then. The Virtual U3S supports all the modes of the real U3S: QRSS, FSKCW, DFCW, Hell, DX Hell, Slow Hell, FSK, CW, CW ID, WSPR, WSPR-15, Opera, PI4, JT9, JT65, ISCAT, TX CW, TX FSK and customisable patterns. WSPR Extended modes are supported (type 2 and 3 messages for 6-character Maidenhead grid subsquare locators, and Callsign pre/suffix). Additionally QMX supports a RTTY beacon mode.

Some weak signal mode types have a fixed data payload. Others (QRSS, for example) transmit a free-format message. In the Virtual U3S, these free-format messages are sourced from the QMX Messages menu.

Certain information from the QMX such as GPS latitude and longitude can be inserted dynamically into messages, which in the real U3S are known as “Message tags”; the same functionality is available in the Virtual U3S.

A comparison of some differences and similarities between the Actual U3S and the Virtual U3S:

Feature	**Actual U3S**	**Virtual U3S in QMX**
Band Coverage	Depending on which LPF you choose	The band coverage of your QMX version
LF (2200m, 630m)	Yes – depending on chosen LPF	No – limited to your QMX version’s band selection
VHF (4m, 2m, 222 MHz)	Yes – depending on chosen LPF	No – limited to your QMX version’s bands selection
TCXO	Yes - Option	Yes – standard feature
GPS discipline	Yes – Option	Yes – Option (internal QLG3 on QMX+)
Multi-band	Yes – Option, if you include the relay-switched filter kit and choose LPFs for it.	Yes – standard feature, depending on your QMX version’s band selection
Power output	Around ¼ Watt (depends on band and PA configuration. More power optional with additional transistors and higher supply voltage. 5W PA kit was discontinued	Up to 3-6W depending on the power output of your QMX on the band in question. Adjustable in settings.
Configuration via LCD and buttons	Yes	Yes
Configuration via Terminal interface	No	Yes
CAT control	No	Yes
SWR protection	No	Yes, configurable
Available factory assembled	No	Yes
Dynamic message content (Message tags)	Yes	Yes

2. Weak signal mode summary

The Virtual U3S is capable of transmitting a variety of slow/weak signal modes standalone (no PC required for encoding).

The older weak signal modes such as QRSS (very slow CW) are visual modes which are decoded using audio spectrum analzyer software such as the very old but still popular Argo by Alberto I2PHD. Other modes such as WSPR (Weak Signal Propagation Reporter) are decoded by computer software.

The table provides a summary, more details are provided in coming sections.

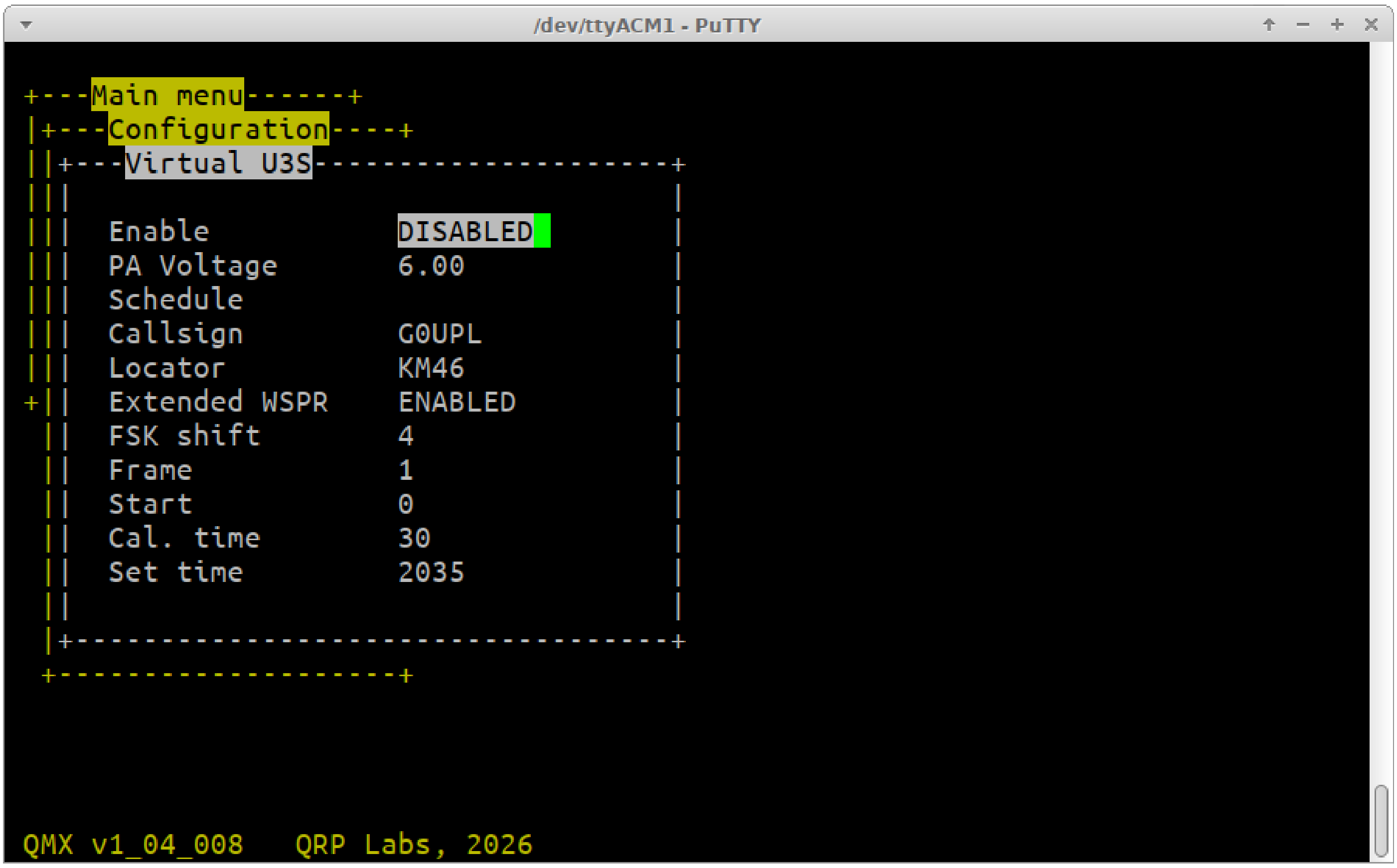
Mode	**Features**
QRSS	Oldest on/off keyed CW, with very slow dit speeds lasting several seconds.
FSKCW	Slow speed CW, a continuous transmission but where the Dah's and Dit's are shifted a few Hz for key-down.
Custom	Customizable patterns are configurable in the Virtual U3S, like the real U3S, allowing a sequence of frequency shifts to "draw" patterns on the frequency spectrum screen (e.g. Argo).
DFCW	Dual Frequency CW: also very slow, but dits and dahs are the same duration, with dah's shifted a few Hz relative to dits.
CW	Standard CW beacon at standard speed (configurable)
FSK	Full speed CW but continuously keyed, with Dah shifted up a configurable number of Hz.
CW-ID	CW transmission of the configured callsign at standard speed (configurable)
WSPR	Weak Signal Propagation reporter: transmits callsign, locator and power in a very narrow 6 Hz bandwidth during a 2 minute transmission. VERY popular!
WSPR-15	8x slower, 8x narrower bandwidth, was used sometimes on LF bands to increase SNR
Opera	Manchester-encoded on/off keyed Callsign transmission
PI4	(PharusIgnis4), a 4-tone error-corrected beacon transmission designed for UHF, often 6m. See https://rudius.net/oz2m/software/pi-rx/
JT9	WSJT-X free-text transmission mode with 5 speeds (1, 2, 5, 10 and 30 minute duration)
JT65	WSJT-X free-text transmission mode, with three tone separations (A, B, C modes)
Hell	Hellshreiber "text transmission" mode like an early fax, developed by Rudolf Hell a centry ago (1920s – 1930's). "DX Hell" doubles every column of pixels to make bold text, more DX-suitable. QMX Virtual U3S, like the real U3S, uses genuine pixel patterns recovered from the manual of an actual mechanical Hellshreiber machine.
Slow-Hell	Frequency shifted slow Hellshreiber with narrow bandwidth, suitable for weak signals and display on Argo spectrum analysis software
ISCAT	ISCAT modes A and B (ionosperic scatter)
RTTY	Radio Teletype: 45.45 baud 170 Hz standard RTTY beacon

3. Configuration Parameters

The Virtual U3S feature in the QMX is configured via the parameter settings in a sub-menu named, unimaginatively but also unconfusingly, “Virtual U3S”. In this section the configuration parameters are described in detail. They control every aspect of how the Virtual U3S will behave. The exception is the actual transmitted content of free format message modes such as CW, which are hosted in the “Messages” menu of the QMX.

Editing parameters in the Virtual U3S follows all the same conventions as the rest of the parameters in the QMX. Refer to the QMX operating manual to familiarize yourself with the requirements for editing configuration parameters.

Just like all the other QMX configuration parameters, the Virtual U3S settings are available in the Terminal login to QMX, as well as on the LCD/Controls of the QMX front panel itself.



3.1 Main configuration menu parameters

Enable

When ENABLED, the Virtual U3S will run.

When this parameter is enabled, the Virtual U3S will run automatically as soon as the QMX is switched on.

You can escape from the Virtual U3S mode by pressing the right button (a.k.a. the “Exit” button) on the QMX control panel. However, the next time you power up QMX, or the next time you enter the QMX configuration menu to make changes then exit it, the Virtual U3S will again run automatically.

PA Voltage

```
PA Voltage
6.0
```

The PA voltage you set here, determines the power output of the Virtual U3S. The actual PA voltage seen at the BS170 drains at full amplitude is slightly below this configured value. To be clear, what the PA Voltage setting actually means is that the PA is set up as if the overall supply voltage to the whole radio was this voltage.

The default value is 6.0V. Remember that power is the square of voltage. Therefore on a 12V-built QMX, setting the PA voltage to 6.0V will be expected to produce about ¼ of the full power output. So something around 1W give or take. Bear in mind for 9V beacon, that would need to be 4.5V.

Many weak signal modes consist of long periods of key-down, which will cause higher levels of PA transistor heating than lower duty cycle modes such as CW or SSB or even other Digital modes such as FT8. For this reason it is recommended to keep the Virtual U3S power somewhat below the full available power.

Callsign

```
Callsign
G0UPL
```

The callsign setting is an alphanumeric setting, which may be up to 14 characters long. It can theoretically include punctuation but clearly that isn't normally used in a callsign. The callsign is NOT used for the normal message transmission in most modes.

The callsign is used only for three purposes:

1) The Callsign is used for generating the message encoding in WSPR, Opera and PI4 modes
2) The Callsign is sent in the CW ID mode, which can be used to send your callsign in standard CW at pre-programmed intervals, as a station identifier (read your license conditions). The Power/speed parameter sets the CW speed in Words Per Minute (wpm).
3) The Callsign can also be inserted automatically into the standard Message transmission, if you include the “#CS” tag in the message.

Note that when using the WSPR and Opera modes, the callsign must satisfy certain rules. Firstly, it can only be 4 – 6 characters long. The callsign must consist of the following:

1) One or two characters consisting of A-Z or 0-9

2) One character which must be a number, 0-9

3) Two or three characters which must be A-Z

These restrictions are placed on the callsign in order to satisfy the requirements of the WSPR and Opera encoding algorithms. PI4 callsigns are 8 characters, any of A-Z, 0-9, space or '/'. **Be sure to finish your 4, 5 or 6 character callsign with a string terminator, using the "Enter" symbol (see main QMX operating manual). Do not be tempted to pad the remaining string size with spaces!**

The callsign must not contain any spaces. Any necessary padding with spaces will be carried out automatically by the system when encoding the WSPR message.

On occasions where you need to use a callsign prefix or suffix in your WSPR message, this is supported; please read the "Extended WSPR" configuration parameter setting below. Opera modes do not support this.

Locator

The locator setting is only used in WSPR and PI4 modes, and is one of the inputs to the WSPR encoding algorithm. It is six characters, the first two are always A – R, the second two are numbers in the range 0 – 9, and the final pair are always A – X. The locator is not needed for any other mode than WSPR and PI4, and may be left blank if not using these.

If a GPS module is connected with a correctly configured serial data stream, then the locator is set automatically from the GPS data and does not need to be entered manually.

WSPR messages ordinarily just encode 4-character Maidenhead locators such as IO90 in this example. The 5 th and 6 th fine detail location specifiers ("AB" in this example) are ignored. On occasions where you need to send the full 6-character locator in your WSPR message, this is supported; please read the "Extended WSPR" configuration parameter setting below.

The 4-character or 6-character locator can also be dynamically inserted into the Message (for transmission modes that read the Message string) if you use the "#M4" or "#M6" tags, respectively.

Extended WSPR

This parameter enables the extended WSPR mode when set to 1. In extended WSPR mode, you may use a callsign containing a prefix or a suffix (not both). Furthermore, the full 6-character locator is transmitted providing more accuracy than the usual four. However, extended WSPR mode does have some disadvantages:

1) Two transmissions are required in order to transfer all the information, so a WSPR transmission in extended mode takes 4 minutes. The receiving station must receive both transmissions in order to fully copy all information, or in some cases, even your callsign (which is only sent in one of the two message types). In weak signal conditions a station may not receive more than one decode from you; and you will therefore lose reception reports compared to ordinary WSPR which only takes one transmission (one 2-minute slot).

2) The two transmission types are linked by transmission of a 15-bit hash code. This is sent in the second message instead of the full callsign. It is like a lossy-compression of the callsign. Multiple callsigns would produce the same hash code because there are only 32,768 possible values (15 binary bits). There is therefore a risk of false decodes. If many people switch on extended mode, there is an increased risk of false decodes.

These are limitations of the WSPR protocol, not the U3S kit. The same limitations apply to WSPR sent using a PC and the WSPR program.

For the above reasons, **enabling extended WSPR is not recommended unless you really need to send a long callsign or the full 6-character Maidenhead locator.**

When sending a callsign in extended WSPR mode with a prefix or suffix, three types of extended callsign are allowed, as follows:

1) A single character suffix in the range 0-Z, e.g. G0UPL/P

2) A double digit suffix, numeric only, e.g. G0UPL/26

3) A prefix consisting of between 1 and 3 letters or numbers, e.g. MM/G0UPL

The extended WSPR protocol does not allow both a prefix and a suffix, and it does not allow callsign extensions other than the three types listed above. The U3S kit validates the entered callsign to ensure that it meets the requirements.

FSK shift

This setting determines the FSK size, and may be in the range 0 to 999Hz. For FSKCW and DFCW modes, the specified FSK is the height of the symbol shift. For Slow-Hell mode, it is the height of the text. The parameter does not apply for the other modes.

Generally, you will want a shift of 4 or 5 Hz for the slow-speed modes. Any more than that, will be considered anti-social by other operators. For the fast-speed FSK mode used for beaconing on the 10m band, a few hundred Hz is typically used.

Note that when using WSPR modes, the correct frequency shifts for WSPR are calculated by the kit; the Fine and FSK settings have no effect in the WSPR mode. They also have no effect on CW, CW ID, QRSS, Opera, PI4, JT9, JT65, ISCAT or the Hellschreiber modes: these all have frequency shifts defined already by the mode protocol.

Frame

The Frame parameter is used to define the message frame length.

A technique called “stacking” has been developed by QRSS'ers to improve signal-to-noise ratio even further. The transmitting station transmits messages defined into a frame of a precise length. The receiving station overlays multiple message reception frames on top of each other on the display. Any message which is repeating at that precise frame length gets re-enforced by the averaging of multiple message receptions; any other non-repeating message or QRM gets reduced.

As an example, consider a message which in the chosen QRSS mode and speed happens to take 8 minutes to transmit. You could set the “Frame” setting to 10. This means that after the message transmission has completed, the microcontroller will wait until 10 minutes after the message start, before starting the next transmission. Receiving stations who are using QRSS decoding software with the stacking feature, and have it set to 10 minutes frame length, will be able to collect six (say) transmissions over one hour and average them together to improve the signal-to-noise ratio.

This parameter is also used for the WSPR, PI4, JT9, JT65 and ISCAT modes. In these modes, the Frame parameter defines how often the encoded transmission data is transmitted. WSPR transmissions take just under 2 minutes and are always aligned to the start of an even minute. Therefore in WSPR mode, the Frame parameter must be an even number (a multiple of 2), e.g. 10 minutes (9 minutes is not possible).

Typically, Frame should be set to 10 minutes for WSPR. 10 minutes is also a popular frame length for “stacking” QRSS reception, as long as your message transmission fits within 10 minutes. If it does not, the controller will wait until the next 10 minutes have elapsed.

In WSPR-15, Frame should be a non-zero multiple of 15 minutes, typically 15 minutes.

When using GPS frequency stabilisation, you must choose a non-zero Frame parameter. This is because the frequency stabilisation must take place when there is no transmission from the kit, i.e. during a pause between transmissions. The frequency stabilisation takes a configurable number of seconds and occurs just after the end of a transmission. Therefore you need to ensure that the Frame parameter is sufficiently large that there is a minimum gap equal to the configured calibration time (see below), between the completion of your message transmission and the start of the next one. The duration and type of the frequency stabilisation is determined by the “Cal” parameter, described later.

Default: the default value is 00 which means no frame size is defined, and the stacking feature is not used. All mode/frequency combinations are sent repeatedly in sequence. Frame must be set to a non-zero value for WSPR, WSPR-15, PI4, JT9, JT65 and ISCAT modes.

Start

The Start parameter is used in conjunction with the Frame parameter. It defines when the transmission cycle starts. For example if Start is set to 03, and Frame is set to 10, then the message transmission will start on the when time reaches 00:03 (or 3 minutes past any hour), and will repeat at exactly 10 minute intervals.

Again for WSPR, the Start parameter must be even (a multiple of two). For WSPR-15, a multiple of 15.

Default: the default value is 00 which means the Start parameter is not used. Frame parameter is used to define the message frame length.

Cal. time

Cal. time configures the duration of the calibration process which follows the completion of all configured message transmissions in the schedule. Calibration measures the frequency of the QMX's 25 MHz TCXO clock. The error measurement is then used to accurately adjust the transmitted frequency.

In order for the calibration feature to work, a GPS must be connected to the QMX (or example, the internal QLG3 GPS option in the QMX+) and it must be correctly configured in the QMX System Configuration menu. In the case of the QMX+ this means you must select the internal GPS (as opposed to the default which is the an external GPS plugged into the paddle port).

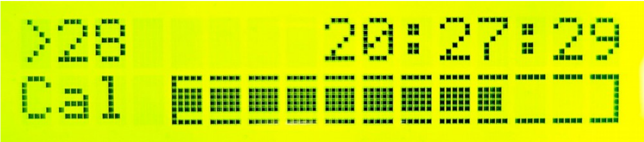
The frequency calibration method does not, internally, work the same on the QMX as on the actual Ultimate3S. One the QMX we have the luxury of the built-in TCXO reference as standard (which is an option on the Ultimate3S). The 25 MHz TCXO is already very accurate, typically within around +/- 5Hz.

The calibration procedure in the QMX works by using the 1pps signal from the GPS to gate an internal counter which measures the frequency of the 168 MHz oscillator. The ratio between the internal 168 MHz CPU clock and the 25 MHz TCXO is 6.72. Therefore even in a one second measurement the resolution of the calibration is 1 / 6.72 = 0.15 Hz. However the system averages the measurements for as many seconds as you configure in the Calibration time setting. Therefore higher accuracy is achieved by using longer calibration times. But generally it is not necessary to go overboard.

The maximum possible Cal Time is 250 seconds. If the configured calibration time is longer than 250 seconds then 250 will be used.

If the calibration time setting is insufficient to fit into the remaining empty time before the start of the next frame of transmissions, then when the next frame schedule begins the calibration is aborted.

During the calibration process, a nice progress bar is displayed on the QMX screen, just like on the old Ultimate3S screen:



On completion of the calibration, the measured error (difference between the actual TCXO frequency and the nominal 25.000000 MHz frequency) is shown on the screen for ten seconds:

In this case indicating an actual measured frequency of 25,000,003.19 Hz for the TCXO frequency.

After ten seconds this display is replaced by a display showing the GPS status:

In this case this indicates 'A 3D' (valid 3D satellite fix), with 13 satellites used in the fix computation, of a total of 18 being tracked. 40 dB is the average SNR of the fix satellites.

Set time

The Virtual U3S uses the QMX real time clock, which is used for the Frame (QRSS stacking) and for timing WSPR, PI4, JT9, JT65 and ISCAT messages. This configuration parameter allows you to set the real time clock. The clock setting takes effect, and the seconds (internally counted but not displayed) are set to zero, when the RIGHT button is pressed after setting the minutes. Accurate timing is critical for the WSPR (etc) modes. The 25 MHz TCXO reference in QMX is so accurate that QMX will keep aaccurate enough time for WSPR for several weeks, even without GPS.

Note that if using a GPS module with and if there is a valid serial data stream properly decoded, then the time is set automatically, and there is no need to adjust this Set time parameter. An accurate real time clock is not required other than for the WSPR, PI4, JT9, JT65 and ISCAT modes.

Even if the real time clock is not set, it will start from 00:00 when the system is powered up. This is sufficient for stacking if enabled (i.e. if Frame is non-zero). If the QMX+ is used, with CR2032 battery backup, then the real time clock is the battery-backed-up clock in the CPU and will continue operating correctly even when the QMX is powered down.

Starting the Virtual U3S

Once you have everything set up just how you want it, and you've set the "Enable" setting to "ENABLED", then all you have to do is exit back out of the entire QMX menu system, to the normal operating mode of the QMX. The Virtual U3S takes over the operation of the QMX and shows its progress on the LCD in the same way the actual Ultimate3S kit did.

3.2 Configuring the transmission schedule

On the Virtual U3S, the schedule of transmissions is set up in a sub-menu table of settings. A view of this in the terminal interface is shown below. Of course, the settings may also be edited directly on the QMX using the LCD and buttons. In this case please refer to the main operating manual if you are unfamiliar with the method for scrolling between columns of a table menu such as this. Note that to move columns left or right, you have to select a row with a number; if it's a list type (such as ENABLED/DISABLED) then the arrow keys will move that.

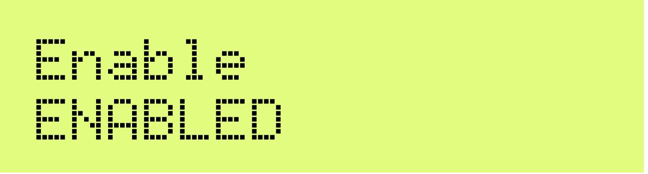
```
/dev/ttyACM1 - PuTTY

+---Main menu------+
|+---Configuration----+
||+---Virtual U3S---------------------+
|||+---Schedule--------------------------------------------+
||||                                                       |
||||  TX #           0        1        2        3        4        |
||||  Enable         ENABLED  DISABLED DISABLED DISABLED DISABLED |
||||  Frequency      14113900 14060900 14060300 14060300 0        |
||||  Mode           CW-ID    RTTY     FSK      DFCW     WSPR     |
+|||  Power/Speed    12       23       16       1        0        |
 |||  Message index  0        0        0        0        0        |
 |||                                                       |
 ||+-------------------------------------------------------+
 ||  Cal. time      30           |
 ||  Set time       2117         |
 ||                              |
 |+------------------------------+
 +-------------------+

QMX v1_04_008   QRP Labs, 2026
```

The menu contains 16 columns of settings. Each column configures one transmission in the schedule. Each of the transmissions may be on a different band, frequency, and use a different mode and message transmission (for modes supporting free text). The settings in each column correspond to the Ultimate3S mode setting screens. The columns are labeled with a TX # from 0 to 15. The setting rows for each column are now described.

Enable



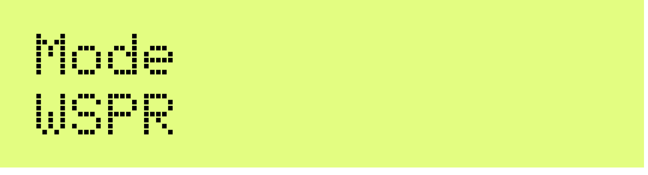
The Enable setting just configures whether the column is used or not. If DISABLED the column is ignored. If set to ENABLED, the column specifies a transmission and the rest of the parameters must be correctly set up, to avoid generating an error message.

Frequency

This is the Frequency used for the transmission. The Virtual U3S will automatically find the correct Low Pass Filter to use for the transmission, according to the limits set in the Band Configuration table.

For most modes, Frequency specifies the lower of the frequencies (where frequency shift is concerned). Not all. For example, for WSPR the specified frequency is the CENTER frequency of the transmission, not the frequency of the lowest tone.

Mode

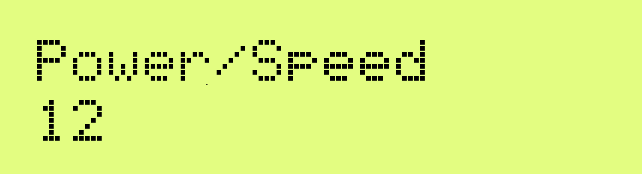


Here the transmission mode of this transmission slot is specified. The available modes are: WSPR, WSPR15, CW, CW-ID, FSK, FSKCW, QRSS, DFCW, Slow Hell, Hell, DX Hell, Opera 05, Opera 1, Opera 2, Opera 4, Opera 8, Opera 32, Opera 65, Opera 2H, PI4, JT9_1, JT9_2, JT9_5, JT9_10, JT9_30, ISCAT_A, ISCAT_B, JT65A, JT65B, JT65C, RTTY, RX.

Note that RX is a special case which is 2 minutes of reception on the specified frequency. It isn't a transmission mode as such.

The transmission modes will be described in more detail later in this document.

Power/Speed



The meaning of this parameter changes depending on the transmission mode being used.

For WSPR, this is the Power being encoded into the WSPR transmission. Note that this is a configured power which YOU are telling the encoder; it is not measured from your system. In WSPR the allowed values for power are 00, 03, 07, 10, 13, 17… etc. dBm up to 60. Any other value produces an error. This is part of the WSPR protocol specification.

This table shows the permissible powers for encoding in the WSPR message, in both dBm and Watts:

Power dBm	Power Watts
0 dBm	1 mW
3 dBm	2 mW
7 dBm	5 mW
10 dBm	10 mW
13 dBm	20 mW
17 dBm	50 mW
20 dBm	100 mW
23 dBm	200 mW
27 dBm	500 mW
30 dBm	1 W
33 dBm	2 W
37 dBm	5 W
40 dBm	10 W
43 dBm	20 W
47 dBm	50 W
50 dBm	100 W
53 dBm	200 W
57 dBm	500 W
60 dBm	1000 W

An appropriate value for the QMX if the PA Voltage parameter is set to 6.0, would therefore be 30 dBm (1 W). Note that the values are intended to be approximate. Don't worry about small differences.

The power setting is part of the schedule configuration screen because you may wish to transmit at stepped power levels, using a relay-switched attenuator. In this case you would require the WSPR message to be re-encoded for each transmit frame, and this configuration allows that possibility. You may naturally find that the power output varies on different bands, and this setting allows you to accurately transmit your measured power in the message.

For modes such as CW, the Power/Speed parameter specifies the Speed of the transmission in words per minute. CW-ID is always sent at 12 wpm and the Power/Speed parameter is ignored. For Slow-Speed modes such as QRSS, the Power/Speed parameter specifies the duration of each “dit” in seconds. More details on how to set up the various weak signal modes are in the mode chapter, below.

In modes such as Opera where the speed is fixed, and power is not part of the encoding, this parameter is just ignored.

Message index

Finally, in modes such as CW or QRSS where a free-text message is transmitted, the Message index parameter specifies the location of the message. This allows a different message to be transmitted in different transmission slots (columns of the schedule table). The Message index parameter refers to stored messages in the QMX Message menu. Index 0 is labeled “Message 1”, Index 1 is labeled “Message 2”, Index 2 is labeled “Message 3” and so on. There are 12 messages in the Messages menu, so 12 different texts can be transmitted by the various slots in the schedule.

In modes such as WSPR or Opera where there is no free-text transmission, the Message index parameter is simply ignored.

3.3 Error checking and reporting

There are various opportunities to misconfigure the Virtual U3S, and error checking is designed to point you to the problem so you can fix it easily. The cause of the errors is always well defined and resolvable.

This error occurs when you have enabled the Virtual U3S in the main Virtual U3S menu, but you have not enabled any columns (transmission slots) in the schedule! With nothing in the schedule, the Virtual U3S has nothing to work with.

```
U3S error: 02
Speed
```

This error informs you that the Speed/Power parameter cannot be zero for the transmission mode in TX index #2. The “02” on the top row indicates the TX ID number is 2; which means the third column of the schedule table (0, 1, 2) labeled “2”. Review the mode you’ve chosen and it’s requirements.

```
U3S error: 01
Power
```

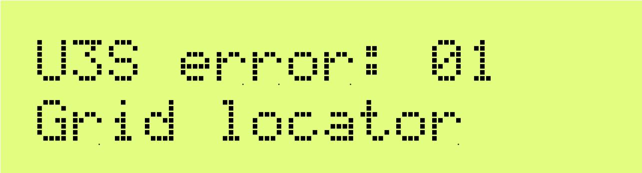
This error informs you that the Speed/Power parameter is inconsistent with the chosen transmission mode in TX slot #1. This happens when you’ve chosen WSPR as the transmission mode, but you’ve entered a power which is not allowed in the WSPR protocol. Please refer to the table in the previous section which shows the allowed WSPR dBm power levels 0, 3, 7, 10, 13, 17 etc.

```
U3S error: 01
Callsign
```

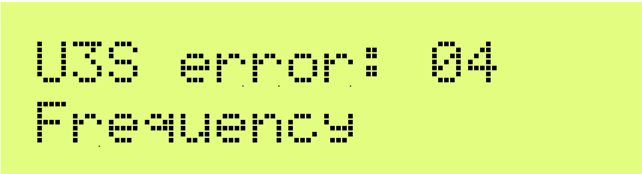
This error informs you that the contents of the Callsign parameter (in the main Virtual U3S menu) are inconsistent with the requirements of the transmission mode in slot #1.

WSPR and Opera have distinct requirements for the callsign. Only 6 callsign characters are allowed, unless you are using WSPR’s Extended mode in which case a prefix OR suffix (not both) is allowed. Please refer to the above sections describing Callsign and Extended WSPR parameters, to understand the requirements for the modes you’re using.

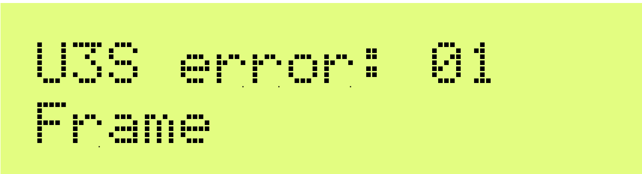
Note also that since the “Callsign” parameter is applied to all slots in the schedule, in some cases it is possible to choose callsigns that work for one mode but not for another. For example if you happened to have a Callsign with a prefix, for use with Extended WSPR mode, this would preclude the use of Opera in any slot since Opera does not support prefixes or suffixes on callsigns.



This error informs you that the Locator setting is empty or invalid, but it is required for the transmission mode in the transmission slot #1. (applies to WSPR, which uses the locator parameter for the message encoding.



This error informs you that the Frequency in TX slot #4 in the schedule is zero or invalid.



This error informs you that the Frame parameter is inconsistent with the mode in transmission slot #1. For example, WSPR transmits in two minutes, and therefore the Frame parameter must be a multiple of 2. Odd minute Frame parameters are not possible, for a Schedule containing a WSPR transmission.

4. Message tags

Certain 2-character tags can be inserted in the transmission message, these will be substituted with dynamic data during the transmission.

For example, if the callsign setting is "G0UPL" and the Maidenhead locator setting is "IO91AB", and the message setting is "#CS #M4 " then the actual transmitted message will be "G0UPL IO91". GPS parameters can also be used. The following list describes the available tags. Any tag not recognised is just sent as a single blank space. If you try to send longitude or latitude when the GPS does not have a satellite lock, you also get a single blank space.

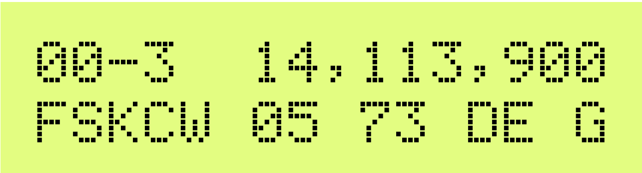
#AT Altitude - all the characters from the GPS string
#GV GPS Validity flag - 'A' indicates a valid fix has been computed
#LT Latitude – degrees and decimal minutes: all the characters from the GPS string, plus N/S suffix
#LN Longitude – degrees and decimal minutes: all the characters from the GPS string, plus E/W suffix
#M4 4-character Maidenhead locator e.g. "IO91"
#M6 6-character Maidenhead locator e.g. "IO91AB"
#CS Callsign configuration entry (just inserts whatever is in the Callsign parameter)

Additional tag parameters will be added in future.

5. Mode description

The Virtual U3S is capable of transmitting a range of QRSS/Weak Signal modes, which are useful for various applications or for experimentation. WSPR is the most commonly used mode in the U3S.

The following image shows a typical display during the slot transmission and the elements of the display are explained below:



At the top left corner, 00 indicates the TX Slot ID (column in the schedule table). In this case it is slot #00 (the first column of the schedule table).

The next character, in some modes, indicates whether a dit or dah is being sent. This applies for example, to CW and its variants. A – shown here indicates a dah is being sent; a . shown here indicates a dit is being sent.

The ‘3’ character at the 4th character position of the top row, actually indicates the Low Pass Filter selected for the configured frequency. This is consistent with the behaviour of the actual Ultimate3S when used with the relay-switched filter board kit. On the QMX the LPF is selected automatically according to the frequency, and the LPF switching uses PIN diodes.

At the top right is the transmission frequency, as configured in the schedule column corresponding to this Slot ID. The frequency normally indicates the lower frequency of any FSK modes, with some exceptions. For example on WSPR mode, it is the CENTER frequency of the 6 Hz bandwidth WSPR transmission.

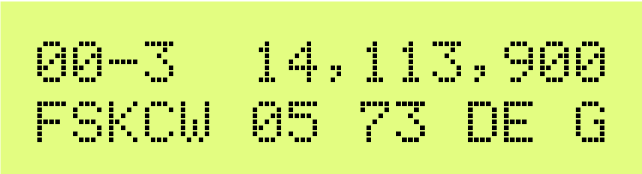
Just to the left of the frequency, is a “heartbeat” which flashes with the GPS 1 Pulse Per Second (PPS) signal, if you have a GPS connected. The typical GPS (QRP Labs QLG2 or the internal QLG3 module option in QMX+) has a 100ms high duration PPS signal. So the heartbeat flashes on for 0.1 seconds, and off for 0.9 seconds. It’s a little quick for the display and you may need to adjust the display contrast and look at the display from a front 90-degree viewing angle.

On the bottom line, details of the mode are displayed. The mode is shown on the bottom left. In this case, FSKCW. In some modes where there is a configurable speed or power, this is shown next. In this case, 05, means the “dit” duration of the FSKCW is 5 seconds. Finally the message is shown; in this case “73 DE G0UPL” was being sent and so the first part of this message is displayed. As the message gets transmitted, the display is updated to show the remaining parts of the message only. So once the ‘7’ has finished being transmitted, this part of the display will now show “3 DE G0”.

On some modes such as WSPR where an encoded stream of FSK symbols are sent, this message section of the display at the bottom right is replaced by two numbers: the first is the symbol number, the second is the tone number being transmitted. These numbers update as the message is transmitted.

In the following sections a screen display is shown for each mode, and any exceptions or specific details are described.

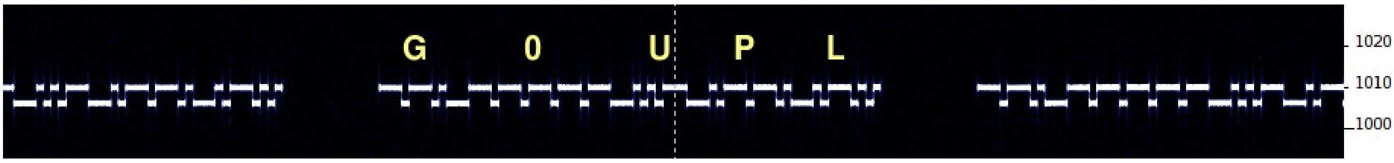
5.1 FSKCW



In Frequency Shift Keyed CW (FSK/CW) the dahs and dits have the same timing as traditional (slow) morse i.e. QRSS, but the carrier is always on and “key down” is indicated by a slight upwards shift of a few Hz. FSK/CW is the most popular of the slow-signal modes.

The Power/Speed parameter in the schedule should be set to the duration of a dit, measured in seconds. The Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc).

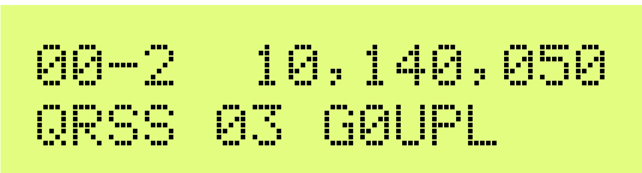
In the following image, which shows how the transmission appears on Argo, the decoded letters have been drawn above the received signal in yellow, to show clearly how the mode is to be decoded.



Good for: The transmitter being continuously on, avoids any “chirp” (frequency pulling) of the oscillator, which did used to be a problem, particularly in more simple circuits (but not a concern in the QMX with its fine TCXO). The signal is very readable in the presence of QRM or in weak conditions, because of the continuously present carrier. Decoding the signal is intuitive, just by reading the top line like ordinary CW.

Bad for: Because the carrier is continuously on, it will take more battery power (if that's a concern), than an on/off keyed mode like plain CW. It also takes more bandwidth than the single sub-Hz bandwidth of plain QRSS.

5.2 QRSS

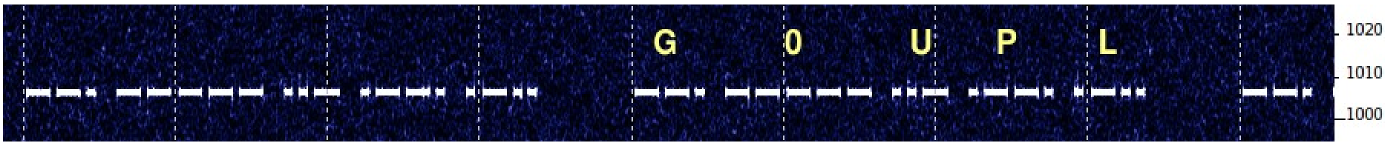


The original and oldest plain QRSS mode, which is simply plain CW, but massively slowed down, typically with dit speeds from 3 seconds or slower.

Here QRSS with 3 second dits is used, on frequency 10,140,050 (LPF #2), and the message is just “G0UPL”.

The Power/Speed parameter in the schedule should be set to the duration of a dit, measured in seconds. The Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc).

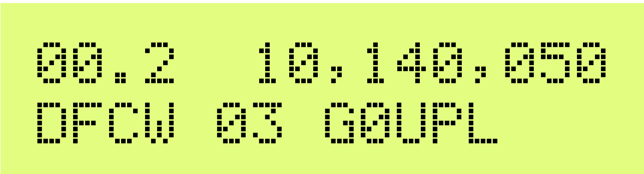
In the following image, which shows how the transmission appears on Argo, the decoded letters have been drawn above the received signal in yellow, to show clearly how the mode is to be decoded.



Good for: Simple to decode by eye, since it is the same was what we can imagine hearing. Very narrow bandwidth and band occupancy, sub-1Hz.

Bad for: In weak signal conditions or where there is a lot of QRM, an interfering weak carrier may also look like an intermittent horizontal line, so the wanted signal can be hard to distinguish from the QRM. Fading (QSB) can cause interfering carriers to look like QRSS too. In simpler circuits, “chirp” (oscillator pulling) can be a problem.

5.3 DFCW



DFCW means dual-frequency CW. The dits and dahs are the same length, but dahs are shifted upwards by the FSK amount. A 1/3 dit-length gap inserted between symbols improves the readability.

The Power/Speed parameter in the schedule should be set to the duration of a symbol (dit or dah), measured in seconds. The Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc).

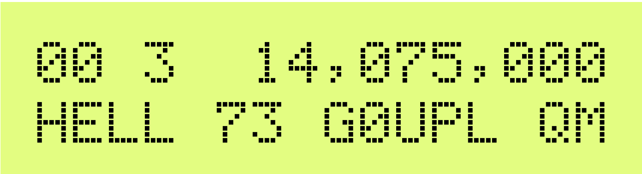
In the following image, which shows how the transmission appears on Argo, the decoded letters have been drawn above the received signal in yellow, to show clearly how the mode is to be decoded.

Good for: The mode is faster than QRSS, with the same signal-to-noise ratio improvement, because the dah's are the same length as the dits. In QRSS dah's are three times longer. So the message transmission is faster.

Bad for: Not very intuitively easy to read as QRSS is. If the signal is a bit weak and hard to decode, the eye/brain may have more trouble “filling in the gaps”, because it is not so intuitive.

5.4 Hellshreiber (also called FeldHell)

Hellschreiber is a fax-like mode designed in the 1920's by Rudolf Hell in Germany. It was later used in WWII. The literal translation into English would be "Light Writer" and of course rhymes with its inventor's name. Each character is made up of a 7-row, 5-column grid. Technically there are 14 half-rows, but there are no lonely half-pixels in the characters, half-pixels are only ever seen in groups of at least two. This was an ingenious idea to minimise transmission bandwidth yet improve readability. The baud rate of a standard Hell transmission is 122.5. Most of the characters fit into a 10-half-row, 5-column grid, but the numbers are extended above and below this in some cases.

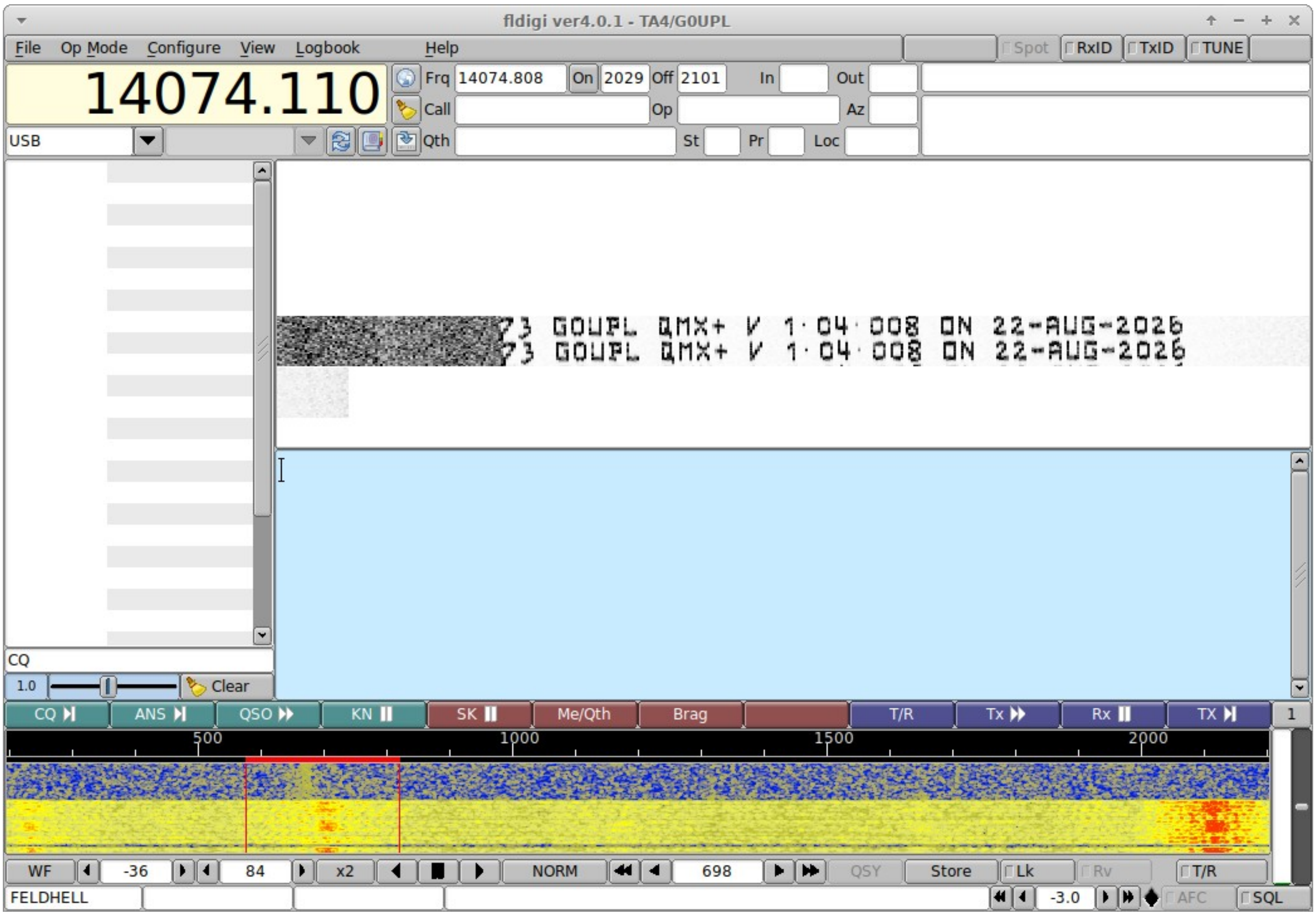


The display set-up is as usual, with the message scrolling across the bottom right part of the screen during the transmission.

The Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc). The Power/Speed parameter is not used.

The screenshot fragment below is from IZ8BLY's excellent Hellschreiber receiving (and sending) software.

This is a screenshot from the popular Fldigi program:



Good for: Fun, history, and easy readability.

Bad for: Bandwidth. Hell is not a slow-signal mode; the bandwidth is said to be around 400Hz.

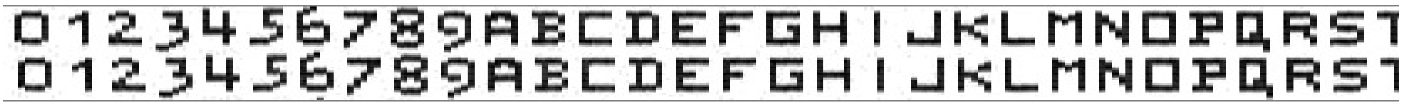
I am proud to be using the original Hell font in this kit as carefully designed by Rudolf Hell. I know this because Pierre ON5SL, who owns a real antique FeldHell machine, kindly sent me a scan of one page of the operator's manual. The following page shows the drum pattern of the machine.

Various software is freely available that can send and receive Hellschreiber, using any Windows font. Specially designed fonts are available. One called “FeldReal.fon” claims to use the original Hellschreiber character design but I am not convinced – if you are interested, look for example, at the K. FeldReal.fon's K actually looks nicer, but it doesn't quite match the one in the machine, look at the top right of the K.

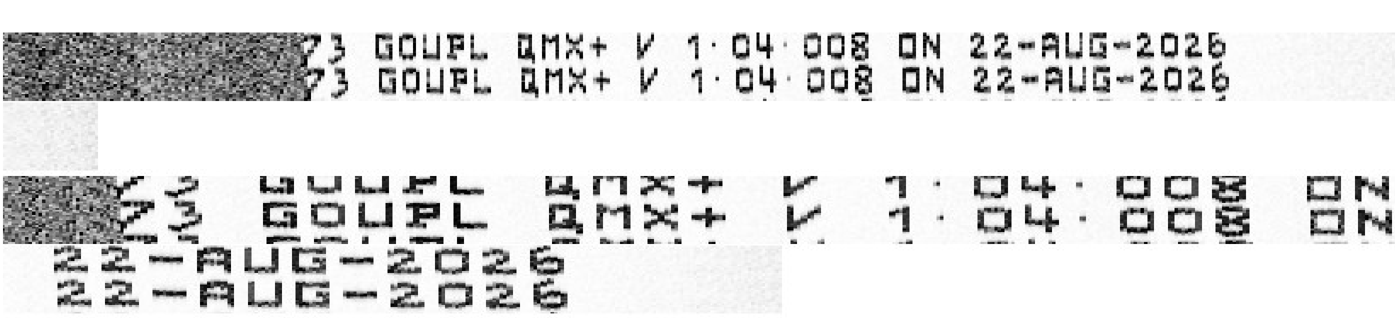
5.5 DX Hellshreiber

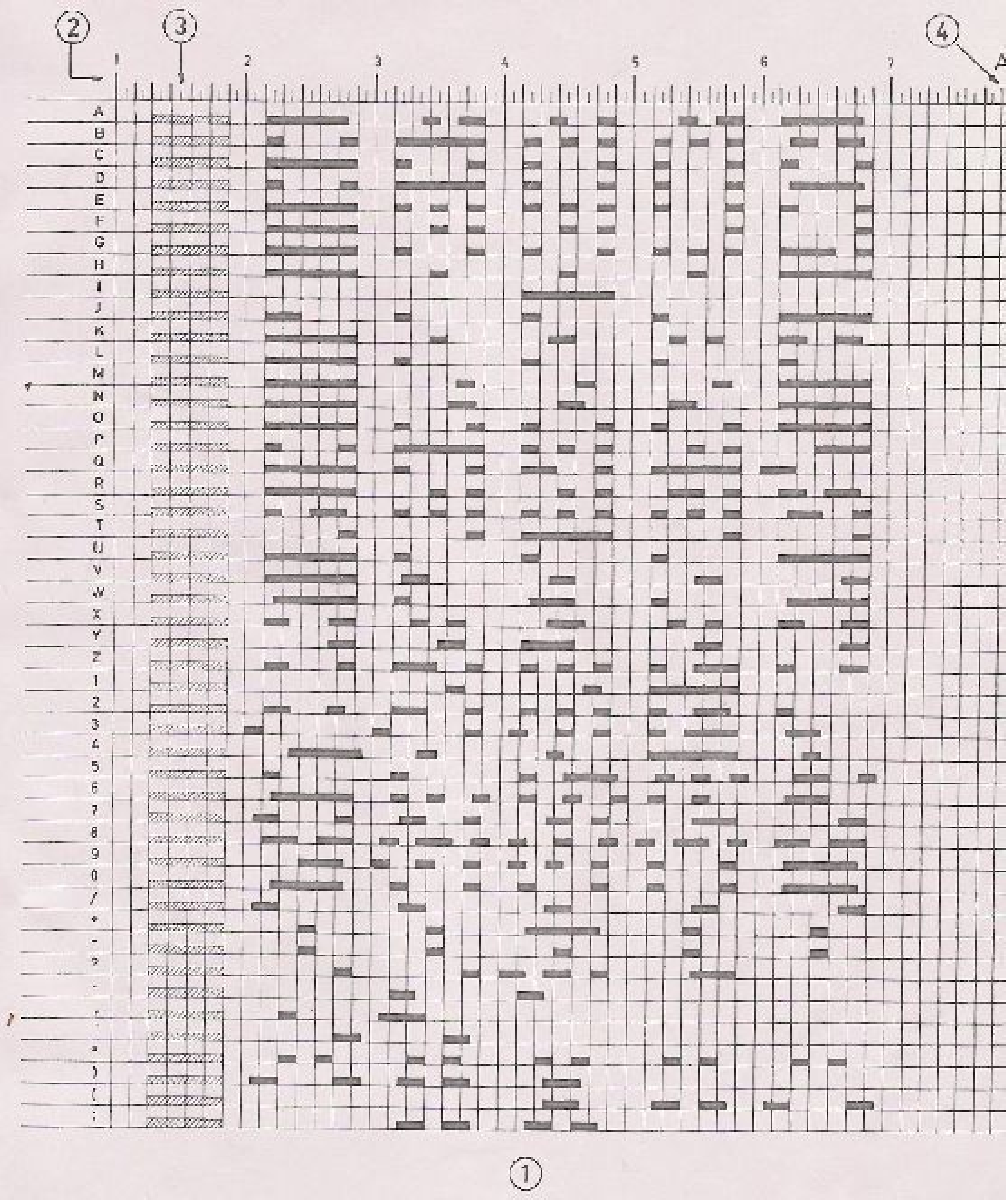
So-called DX Hellschreiber is the same as ordinary Hellschreiber described above, except that every column is sent twice. The characters are therefore twice as wide, the message takes twice as long to send, but it has the advantage of potentially being more readable in weak conditions.

The Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc). The Power/Speed parameter is not used.



The image below shows a side-by-side comparison of Hell and DX-Hell on Fldigi. The regular normal Hellshreiber is the upper trace; the below “bold looking” one is DX-Hell, where every column of pixels is sent twice for greater readability.





15

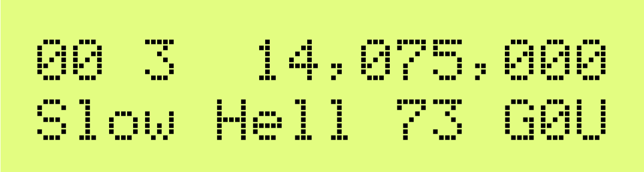
① Abwicklung der Sendenocken

② Einteilung der Meßscheibe

③ Anlaufschritt

④ Nullstellung der Senderachse

5.6 Slow Hellshreiber



Slow Hellschreiber is the slow-signal Hellschreiber equivalent. Each Hellschreiber letter row is shifted slightly in frequency. Typically the single pixel rate is very slow, for example 1 second per pixel (0.5 seconds per half-pixel). The character pattern is "scanned" one column at a time from bottom to top, shifting the frequency as the row increases up the character. In this way an image of the character is built up at the receiving station.

In Slow Hellshreiber mode, certain configuration settings in the kit require special attention:

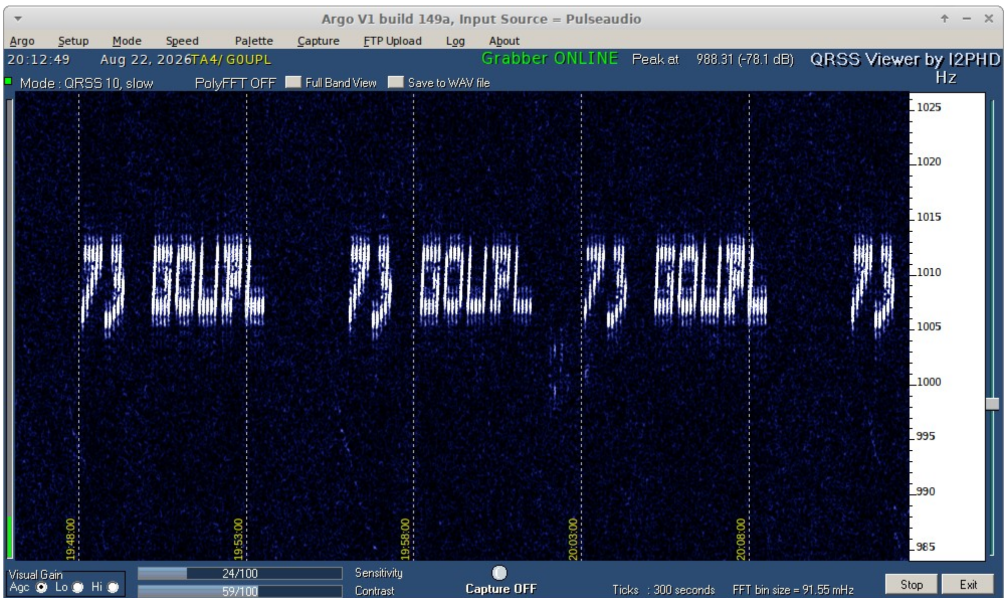
Speed: The speed setting is the number of seconds required to transmit one whole character. Recall that a Hell character is composed of 7 rows (14 half-rows) and 5 columns; there is also one empty column as an inter-character space. Therefore each character is composed of 42 "pixels". Therefore if you want the transmission time to be 1 second per pixel, set Speed to 042. If you wanted 1.5 seconds per pixel, you would set Speed to 063, and so on.

FSK shift: The FSK size configuration specifies the height of each character, it is the number of Hz for 5 rows of the character. Recall that most Hell characters, such as the letters, fit on a 7-row, 5-column grid. FSK (Hz) specifies the size of that grid; some of the numbers extend above or below the grid. For example an FSK (Hz) setting of 05 would result in characters 5Hz high, i.e. 1Hz per row (pixel).

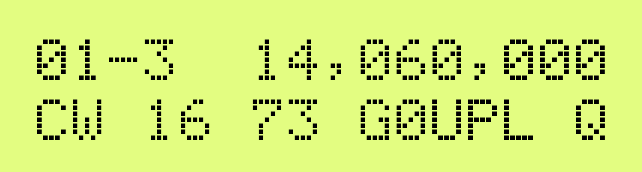
Be aware of fundamental limitations of information theory. If the pixel rate is 1 second (Speed = 042), the minimum theoretical bandwidth is 1Hz. Therefore choosing a pixel size of less than 1Hz, i.e. FSK (Hz) = 05, does not make any sense. Faster transmissions require larger bandwidth, and you will have to set the FSK size higher to match. If you do not, the image at the receiving end will be blurred.

Again the Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc).

In the image below the text is "73 G0UPL", with Speed set to 42 and a 6Hz letter height. Argo's slow 10 second mode is being used.



5.7 CW



CW mode is just plain, ordinary on/off keyed CW (morse code) at ordinary speeds. The display's top line shows the slot number in the schedule, band and frequency. In this example, it is slot 1, band 3 and 14,060,000 Hz. The bottom line shows CW followed by the speed – in this example 16 wpm. Finally the message is shown. The leftmost character is the one currently being sent. “–“ after the sequence number, in the 3rd character of the top row, indicates a dash is being sent. Naturally a “.” will appear if a dit is being sent. As the transmission is sent, the message will scroll to the left.

In CW mode, the Speed setting is the words-per-minute speed, e.g. 12 for 12 wpm. The Message index parameter specifies which message from the Messages menu to use for the text (0 = Message 1, 1 = Message 2 etc).

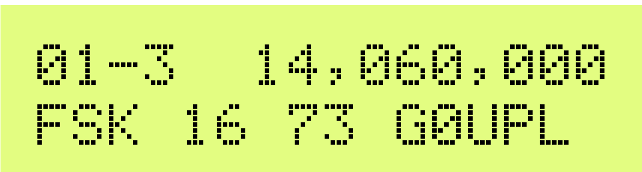
5.8 CW-ID

The CW ID mode can be used for transmitting your callsign identification periodically if you wish, if you think that it helps you to comply with licensing conditions.

This mode is similar to “CW”, it is just ordinary on/off keyed CW. However, instead of transmitting the contents of the “Message” setting, it simply transmits the “Callsign” configuration parameter from the main Virtual U3S screen. The Speed is controlled by the transmission slot’s speed setting. The “Message Index” parameter is ignored.

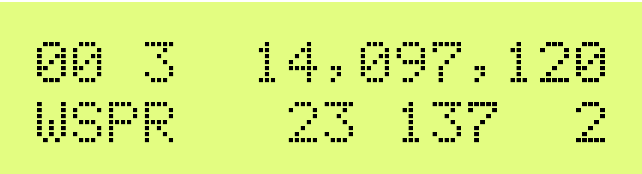
Note that if a mode setting is set to CW ID, the Callsign parameter must not be left empty, and you must enter a valid Callsign parameter for CW, containing only characters A-Z, 0-9, Space or /. An empty Callsign or invalid character in the Callsign will result in an error message.

5.9 FSK



This mode is designed for wideband fast FSK. It is in common use for 10m beacons using FSK shifts of several hundred Hz. The speed setting is interpreted as a words-per-minute speed, e.g. enter 12 for 12 words-per-minute. The Message index setting specifies the message to be sent (0 = Message 1, 1 = Message 2 etc).

5.10 WSPR



The WSPR mode display is composed of several elements. The top row is as usual (Slot ID#, LPF number, Frequency). The bottom row shows the mode (WSPR), the power configuration (23 dBm in this case), followed by the symbol number and the tone. WSPR has a total of 162 symbols, each of which is one of four different tones, separated 1.46 Hz. Therefore in this case, tone # 137 is being transmitted which happens to be tone 2.

Note that the Frequency specified is the CENTER FREQUENCY of the WSPR transmission. It is not the base frequency of the lowest tone. It is DEFINITELY NOT the USB dial frequency! A common mistake is to enter the USB dial frequency of the center band of WSPR operations from the WSPRnet home page. This is what you would set a radio in Upper Sideband mode to, when using WSJT-X (or similar) to generate the WSPR tones. But here we specify the ACTUAL WSPR transmission frequency in Hz.

WSPR stands for Weak Signal Propagation Reporter. The message consists of three parts: the operator's callsign, Maidenhead locator (4-character, e.g. IO90) and two digits specifying the power. At the receiving station, messages are decoded and uploaded to a central internet database.

The WSPR message is encoded into a set of 162 symbols, each may be 0, 1, 2 or 3, using a compressed data format with forward error correction. The symbols are transmitted as tones, each tone separated by 12,000 / 8,192 Hz i.e. approximately 1.46Hz. The duration of each symbol is the reciprocal of the tone spacing, which is approximately 0.683 seconds. WSPR messages take about 110.6 seconds to transmit, and always start at even minutes past the hour.

While a WSPR transmission is in progress, the “WSPR” text shown in the above example screenshot will change intermittently to the 4-character Maidenhead locator being transmitted. This can be useful during mobile operation if a GPS receiver is used, since the WSPR message is dynamically encoded with the fresh location data from the GPS unit, and the display will show the actual transmitted locator for the current location.

In WSPR, timing is critical, so when using WSPR you must set the time configuration parameter as exactly as possible, if you are setting it manually! Be sure to exit editing the Time parameter when the seconds of your accurate clock tick over from 59 to 00. This will ensure the seconds are in sync with your real clock time. If careful attention is given to setting the frequency and the real time clock, then successful WSPR reports will be obtained. Of course these things are easier if you are using a GPS module: the Maidenhead locator will be calculated from the received latitude and longitude, and the time decoded nicely from the GPS serial data stream.

The microcontroller in this kit takes care of the WSPR message encoding algorithm, without any assistance from a PC host computer. It also calculates the tone spacing and symbol duration.

The Virtual U3S also suppports encoding of the extended WSPR message protocol, which can send a callsign with a prefix or suffix, and sends the Maidenhead locator in full 6-character accuracy. This is controlled by the “Ext. WSPR” parameter and described more fully in that section, see above.

Note that the WSPR message is re-calculated at every transmission frame in the sequence, since each frame contains its own power setting. This allows the kit to be used to transmit a series of different powers, if external attenuators are connected, for comparison purposes.

In between message transmissions, the display will show instead just a clock (see below), while we wait patiently for the next WSPR transmission to begin, according to the settings of the configuration parameters Frame and Start. This is useful for checking that the time on your kit is accurately set. The display also shows the minute at which the next frame will start transmitting. In the example below, the time is 13:04:53 and the next frame will start at 13:10:00. During calibration, the bottom line will additionally show a status bar indicating the progress of the calibration (see section above detailing the calibration process, in the description of the Cal Time parameter).

On Argo, WSPR messages look ugly, like this local reception copy (or worse, because you probably are over-driving your receiver when receiving your own signal!):

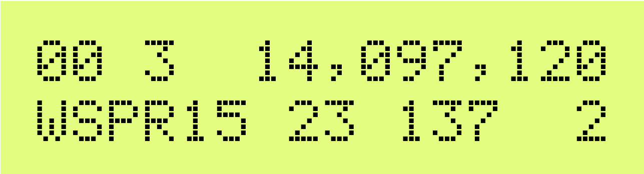
<u>WSPR transmission standard frequencies:</u>

Choose a transmission frequency somewhere in the 200Hz-wide standard WSPR bands:

2200m:	0.137400 – 0.137600	17m:	18.106000 – 18.106200
600m:	0.475600 – 0.475800	15m:	21.096000 – 21.096200
160m:	1.838000 – 1.838200	12m:	24.926000 – 24.926200
80m:	3.570000 – 3.570200	10m:	28.126000 – 28.126200
60m:	5.288600 – 5.288800	6m:	50.294400 – 50.294600
40m:	7.040000 – 7.040200	4m:	70.092400 – 70.092600
30m:	10.140100 – 10.140300	2m:	144.490400 – 144.490600
20m:	14.097000 – 14.097200		

For WSPR decoding, most people use the popular WSJT-X program. See https://wsprnet.org for the home of WSPR, maps of reception, links to WSJT-X download, forum and more.

5.11 WSPR15



WSPR15 is a relatively unpopular experimental mode that is a variation of the common WSPR mode. It is used on the LF bands where the higher signal-to-noise ratio it offers is advantageous. WSPR-15 uses the same encoding algorithms as WPSR, but the tone spacing is 8 times closer (approximately 0.18Hz) and the speed is also 8 times slower.

In WSPR15, the Frame parameter must be a non-zero multiple of 15 minutes, and the Start parameter must be zero (usually) or a multiple of 15 minutes.

WSPR15 is not likely to be useful on a QMX because the QMX is not configured for LF operation; it is included mostly for backward compatibility with the Ultimate 3S kit.

5.12 Opera modes

Opera encodes the callsign (from the callsign parameter field) into a forward error corrected on/off keyed carrier with 239 symbols. There are eight Opera modes, Opera05, Opera1, Opera2, Opera4, Opera8, Opera32, Opera65, Opera2H. The suffix indicates the approximate number of minutes required for the message transmission, with Opera05 meaning 0.5 minutes (30 seconds); Opera2H takes 2 hours to send the callsign. The speed may be chosen to suit the band in use. Slower Opera modes e.g. Opera8, Opera32 are more suited for MF and LF bands.

Opera does not require any time synchronisation, transmissions are successfully decoded regardless of the start time. Opera ignores the Power/Speed setting as well as the Message Indes setting: it always transmits messages encoded from the Callsign parameter which must be set up correctly.

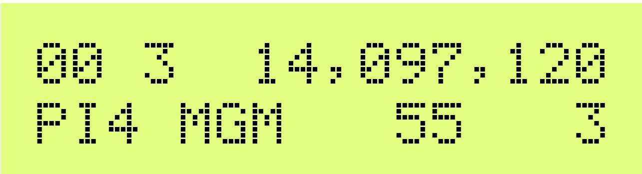
More details on Opera can be found here: http://rosmodem.wordpress.com/

In operation the mode (including time for transmission, 0.5 minutes in this case) is shown in the bottom left, with the symbol number (here 23) and the tone, which in the case of Opera is 0 or 1. Opera is Manchester encoded with on/off keying.

The table below gives standard Opera modes and transmission frequencies (centres of activity). (Note: reception dial frequencies will be 1.5kHz below these for USB with 1500Hz audio).

Carrier Frequency	Mode	Carrier Frequency	Mode
137,500	Opera32	18,106,500	Opera2
478,500	Opera8	21,075,500	Opera1
1,837,500	Opera4	24,926,500	Opera1
3,548,500	Opera2	28,071,500	Opera1
5,290,500	Opera2	50,071,500	Opera1
7,039,500	Opera2	70,094,500	Opera05
10,136,500	Opera2	144,181,500	Opera05
14,063,500	Opera2		

5.13 PI4



PI4 is a beacon mode with a 1-minute sequence. The transmission consists of:

- Callsign encoded forward-error corrected data transmission, lasting 25 seconds. It has 146 symbols of four tones and is similar in concept to WSPR (more accurately, JT4), but the tone spacing is much wider and the symbol duration is much shorter. Unlike WSPR, it encodes only the callsign, not the locator or power.
- Callsign and Locator are sent in 12wpm FSK CW with 250Hz FSK
- The remainder of the minute is filled with a plain unmodulated carrier

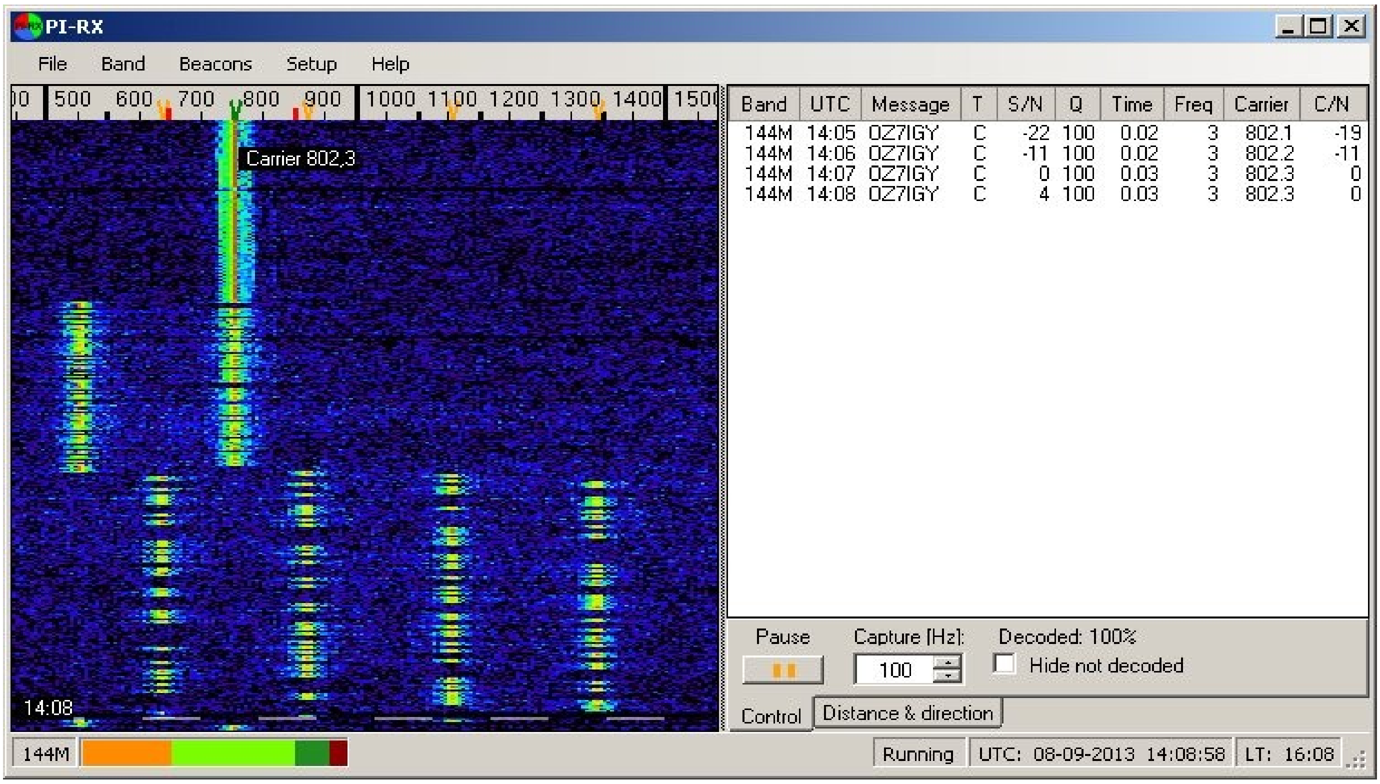
The mode is quite wideband and therefore intended/suited only for beacons for 10m and above.

During the data transmission (also called “MGM”), the screen looks like the image above. Here “55” indicates the 55'th tone is being sent, and “3” indicates the tone being sent (0, 1, 2 or 3).

During the callsign and locator transmission, the display bottom line shows the transmitted text, in the same way as a QRSS or FSKCW transmission. This is either the callsign (for example, G0UPL) or the Locator.

During the plain carrier transmission the display bottom line simply shows “PI4 Carrier”.

The screenshot below shows the PI4 receiver screen and a typical 1 minute transmission. More details on PI4 can be found here: http://rudius.net/oz2m/ngnb/index.htm



5.14 JT9

The JT9 mode is mainly used for short live QSOs. As with the other modes, the bottom right of the screen shows the symbol number and the tone shift.

A JT9 message consists of a 13 character message, containing the characters A-Z, 0-9, space, and + - . / ? characters. The message encoding uses forward error correction (etc.) and produces 85 tone symbols, which are transmitted on one of 9 tone frequencies in a narrow band. The transmitted Message is specified in the Message Index row of the transmission slot column in the schedule table (0 = Message 1, 1 = Message 2 etc). The Power/Speed setting is ignored.

Five sub-types of JT9 are supported: JT9-1, JT9-2, JT9-5, JT9-10, JT9-30. In each case the sub-type (following the hyphen) indicates the approximate number of minutes taken to transmit the message. **Note that only JT9-1 is commonly used nowadays.**

Note that this also imposes limitations on the Frame parameter, since JT9 is synchronised to the start of the minute, and Frame must be set to a multiple of the number of JT9 minutes in the transmission. Similar discussions apply to JT9 as to WSPR, JT65 and ISCAT – the need for accurate time-keeping, in particular.

5.15 JT65

```
00 3  14,097,120
JT65A      37 43
```

The JT65 mode is similar to JT9; JT65 has been used for EME (Moonbounce) communications. There are 65 tones compared to JT9's 9. As with the other modes, the bottom right of the screen shows the symbol number and the tone shift.

A JT65 message consists of a 13 character message, containing the characters A-Z, 0-9, space, and + - . / ? characters. The message encoding uses forward error correction (etc.) and produces 85 tone symbols, which are transmitted on one of 65 tone frequencies in a narrow band. The transmitted Message is specified in the Message Index row of the transmission slot column in the schedule table (0 = Message 1, 1 = Message 2 etc). The Power/Speed setting is ignored.

Three sub-types of JT65 are supported: JT65A, JT65B and JT65C. In each case the sub-type (following the hyphen) indicates the shift – B has a double frequency shift and C has quadruple frequency shift. All take 1 minute to send. **Note that only JT65A is commonly used nowadays.**

5.16 ISCAT

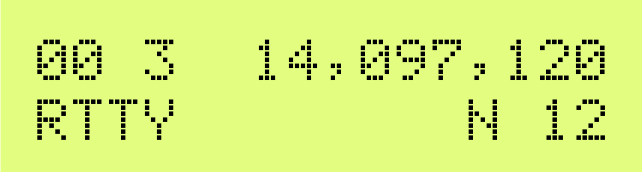
```
00 3  14,097,120
ISCAT-A
```

ISCAT (Ionospheric Scatter) is a specialized communication mode used to make long-distance contacts via weak-signal scattering. It is designed for rapidly varying signals caused by phenomena like meteor scatter, ionospheric scatter, airplane scatter, or reflections off the International Space Station.

ISCAT utilizes a 42-tone Frequency Shift Keying (FSK) system optimized to catch short bursts of reflected signal strength rather than steady, continuous signals. It transmits a free-text message up to 28 characters. The transmitted Message is specified in the Message Index row of the transmission slot column in the schedule table (0 = Message 1, 1 = Message 2 etc). The Power/Speed setting is ignored.

There are two variations of ISCAT: A and B. A runs at a slower data rate (about 21.5 baud) with a narrow bandwidth of roughly 905 Hz, making it slightly more sensitive for steady signals. ISCAT-B is the original sub-mode running at double the speed (about 43.1 baud) with a bandwidth of 1,809 Hz.

5.17 RTTY



The QMX Virtual U3S supports a RTTY mode beacon, which is not implemented in the original physical Ultimate3S kit. During transmission the screen shows the current letter being transmitted (N in this case) from the message, and the Baudot code for that letter.

Amateur radio RTTY (Radio Teletype) is a classic digital communication mode that transmits text messages between radios by shifting a signal between two distinct frequencies. It uses Frequency Shift Keying (FSK) to alternate between a "mark" frequency (binary 1) and a "space" frequency (binary 0). It traditionally uses the 5-bit Baudot code (ITA2), which limits the character set to 32 symbols and requires uppercase letters. Standard amateur operation runs at 45.45 baud (about 60 words per minute) with a 170 Hz shift between tones.

Despite more modern digital modes with higher performance, RTTY remains very popular during weekend operating contests and roundups.

The transmitted Message is specified in the Message Index row of the transmission slot column in the schedule table (0 = Message 1, 1 = Message 2 etc). The Power/Speed setting is ignored.

FLDigi software (and others) may be used for decoding RTTY.

Technically, the baudot code has only 32 values (5 bits). These are framed by one start bit and two stop bits. 32 values is not enough to encode the usual 26-character alphabet plus 10 numbers. Therefore there are two “Registers”, one for the “Letters” and one for “Figures” (which also includes some punctuation characters). A special code 31 switches the receiving machine to the letters register and code 27 switches it to the figures register.

In QMX Virtual U3S the register definitions are:

```
01234567890123456789012345678901

 EnA SIUrDRJNFCKTZLWHYPQOBGfMXVl
 3n- b87re4',!:(5")2$6019?&f./;l
```

Note that the space character is 4 in both registers; value 0 is Null (empty tape). Several special characters exist and you can send these using a special prefix in the message. For example to send a carriage return character r (code 8 in both registers), enter \R in the message. So in all cases, prefix the upper-case version of the lower-case letter shown in the register description above, with a backslash character \ in order to get the special character code.

RTTY special character codes:

Special character	Code	Meaning	In Virtual U3S message put
	0 (both registers)	Null character (empty tape)	Just don't, why would you
n	2 (both registers)	Newline (ASCII 0xA)	\N
b	5 (Figures register only)	Bell	\B
r	8 (both registers)	Carriage return (ASCII 0xD)	\R
e	9 (Figures register only)	Answer back	\E
f	27 (both registers)	Change to figures register	\F
l	31 (both registers)	Change to letters register	\L

Probably the only useful one of these is the Carriage return, which neatly terminates the row of decoded text and moves the cursor down a line, on the receiving end software (for example FLDigi).

Switching between registers is done automatically by the software when it needs to do so.

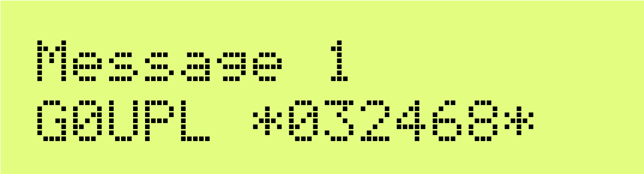
5.18 RX

RX is your opportunity to insert a break in the transmission schedule for listening. It puts a 2 minute receive gap in the schedule, with the Receiver tuned to the specified frequency. The display shows at the bottom right, a count down from 120 seconds which indicates the remaining listening time.

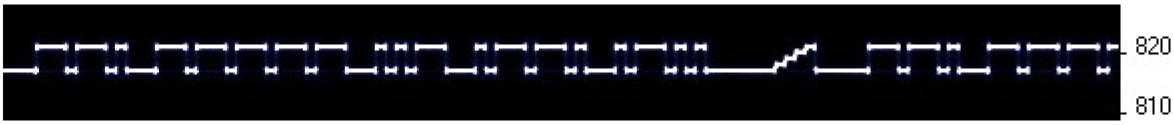
6. Customized message patterns

The Virtual U3S (just like the real U3S) kit has the ability to produce customised message patterns in the FSKCW mode. To enter or leave custom mode, a * character is inserted in the message. The first two characters after the * character specify the symbol duration of each symbol, in seconds. Subsequent characters specify the frequency shift of each symbol one by one, in units of 0.5Hz. For example, 5 means a shift of 2.5Hz. After the numbers, you can continue with letters. So A would mean a shift of 5Hz, B a shift of 5.5Hz, C a shift of 6Hz, and so on up to Z. A space character can be used, in order to cause a key-up (no transmission). Note also that when "Fine FSK" is enabled, the shift is reduced to 1/16Hz steps.

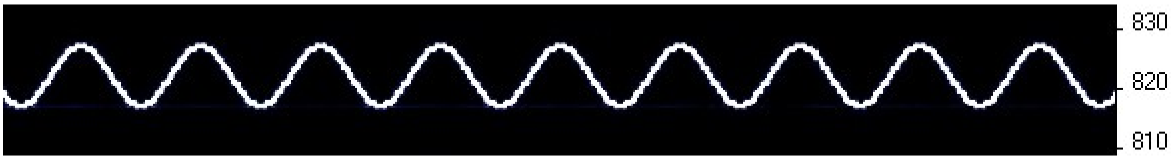
For example consider the following message:



The callsign G0UPL is sent in ordinary FSK/CW with height and speed according to the configuration settings. Next comes a custom message, the duration of each symbol is 03 seconds. The shifts are 1, 2, 3, 4Hz respectively. On Argo, this message produces (with kit configuration FSK (Hz) = 04, Speed = 003):



As an example of what kinds of patterns can be produced, the pattern below is created using message *01ACDFGIJJKKKJJIGFDCA87542110001124578*



7. Resources

Please see the kit page http://www.qrp-labs.com/ultimate3 for any information on the actual Ultimate3S, which has been in continuous production from 2015 to date.

Use the QRSS Knights forum https://groups.io/g/qrssknights to announce your QRSS or FSKCW etc. operation.

For general information and more QRP Labs projects, links to other QRSS resources etc., see http://www.qrp-labs.com.

Argo QRSS decoding software by Alberto I2PHD: http://www.weaksignals.com/

WSPR network home page: http://wsprnet.org

8. Document Revision History

1_04_008	27-Aug-2026	First version, for firmware 1_04_008, the first version with the completed Virtual U3S
1_04_008a	28-Aug-2026	Corrected typos and copy/paste errors (thanks Stan KC7XE and François F4LAD