Quansheng UV-K5(8) Handheld Radio

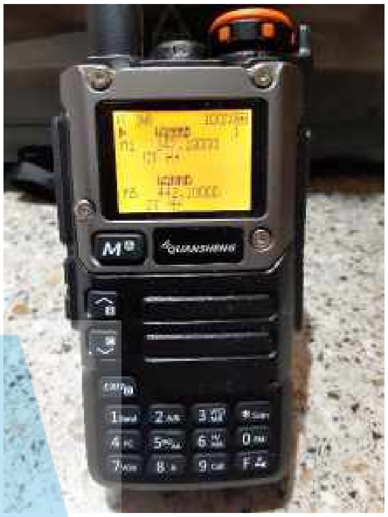


Figure 1 - Quansheng UV-K5(8)

A little while ago, one of my fellow GCARC members came to me with a new handheld radio that he liked. That radio (Figure 1) is the Quansheng UV-K5(8), but not quite the way it came out of the box. This is an extremely inexpensive unit, coming in at that time at an Amazon price of $30.50, which includes in addition to the body of the radio (Figure 2) a standard three-foot USB-A to USB-C cable, a hard-wired 117VAC charging cradle, a battery, a belt clip, and of course an antenna. There is also an instruction manual, which is actually fairly well written.

The radio is listed on Amazon as being an *“AM/FM/DTMF walkie Talkie 200Three Frequency Receiving walkie Talkie NOAA Weather Forecast”* (*sp*) device. As of this writing, the radio shows as being in-stock and available following a brief out-of-stock period around the first of February. For some unknown reason, this little radio is the hottest thing on the web, and several of our Club members have now purchased them. Some of the attraction is the fact that it is a fully software-defined unit, having capabilities that are not often seen in handheld radios, especially those in this price range.

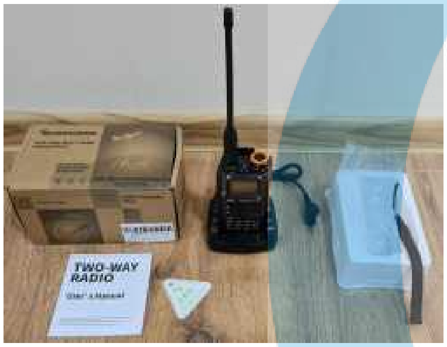


Figure 2 - What's in the box

There are multiple options available as to firmware for this radio, with various differences between the firmware versions and even between the different releases of a given firmware set. I ordered my radio on a Saturday and it was delivered the next day. However, I did not even open the box until Monday, as I wanted to investigate the different firmware choices that were available. After much research and viewing several online videos, I opted for the *egzumer* firmware, which was current at release v0.21 at that time. It has now been updated to release v0.22, though I have not yet installed that release.

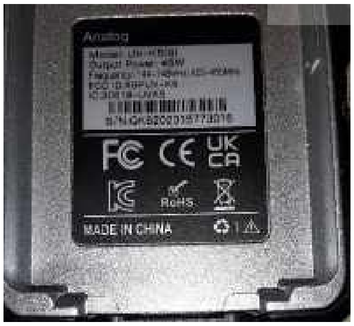


Figure 3 - Factory label

The radio factory label (Figure 3), located inside the battery compartment (under the battery pack), gives no indication of the radio being of any specific design level or variant. Apart from the Recycle logo, the “do not dispose” symbol, and logos of various approving bodies, it simply provides the model number, the US FCC ID number, the Canadian IC ID number, the rated output power, the serial number, and a barcode. There is also a statement of the country of origin on the label.

I will take a moment right now to voice a caution to my readers. Make sure that you really want to install and use the *egzumer* firmware, as it is quite difficult to return to the factory firmware. The factory firmware is available for download at the Quansheng

Download Center page for the UV-K5(8) radio, http://en.qsfj.com/support/downloads/3268. On that page, you will find a firmware file, a USB driver for the Quansheng programming cable, and an executable file for the installation of a firmware updating application. The catch is that if an attempt to roll back the firmware to that factory release using the factory firmware updater, an error message about a version mismatch is returned. The firmware file is a standard *.bin* file, and it can be installed to the radio using other processes, but caution must be exercised to avoid bricking the radio.

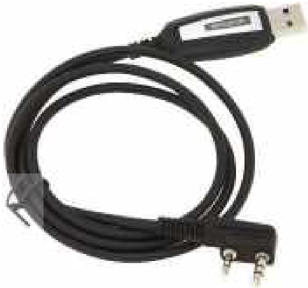
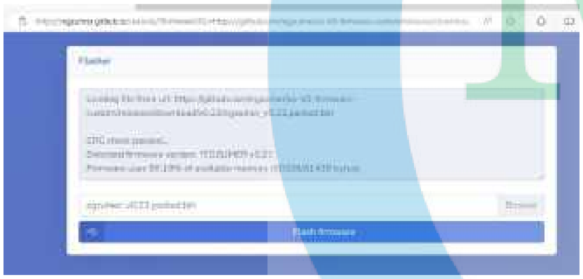


Figure 4 - Programming cable

As to the need for the USB driver, it must be understood that this radio will use any common Baofeng-type programming cable (Figure 4), so if you already own such a cable and the driver for it is already installed to your operating system, the Quansheng cable and its driver will not be needed.

Because I had decided to run the *egzumer* firmware, I never so much as powered on the radio or looked at the display until I had installed the new firmware, which process I will discuss shortly. As a result, I cannot describe or comment on the features offered by the factory firmware. I will be rolling the radio back to the Quansheng firmware at some point in the future solely for the purpose of exploring and then discussing that firmware. However, that will not be done any time soon, which means that it will most likely spawn a second article about the radio, specifically dealing with the factory firmware and the radio's native features.



Flasher

[illegible]

CRC check passed.

[illegible]

Figure 5 - Browser-based firmware flash utility

The road to the *egzumer* firmware starts at https://github.com/egzumer/uv-k5-firmware-custom/wiki/Flashing-the-firmware. The exact installation instructions are given on that page, as is a link to the firmware releases page. The single most attractive thing about this firmware update process is its simplicity. Unlike most firmware update routines, this one is performed directly in the browser (Figure 5), so long as the browser is a Chromium-based browser, such as Chrome, Edge, or Opera. Flashing the firmware requires use of the standard programming cable and placement of the radio into its firmware update mode. That is done by pressing and holding the PTT key while simultaneously powering up the radio. The flashlight will illuminate to indicate that the radio is ready to receive the update. This action can be cancelled *before the download is started* by simply powering down the radio. However, once the update download is initiated, *do not switch off the radio!* The update progress is reported on-screen as a percentage of completion, and the process takes about ten to fifteen seconds once it is initiated. The radio will automatically restart once the firmware update is complete.

Because I performed the firmware flash at the time that I first powered on the radio, I had no programming data in place about which to be concerned. However, those who may have already programmed their UV-K5(8) radios before deciding to change to the *egzumer* firmware will be relieved to learn that the original programming of the radio's memory slots will be untouched by the *egzumer* firmware installation. In addition, certain other settings may also survive the firmware change, depending upon how and where those items are implemented in the firmware structure. When and where a given option is retained in the new firmware, is stored in the same memory location in both firmware images, and that option's values are the same or are translatable to the new firmware, that option will most likely be preserved through the firmware change. Whether the radio had been programmed and its optional settings adjusted or not, it is almost a given that the user will want to do some customization of the radio. While most if not all of the programming and setting of options can be accomplished via the keypad and display, many users will prefer using the familiar *CHIRP* programming interface. The Quansheng brand is represented in the *CHIRP* supported radio list in the form of the TG-UV2+ and the UV-K5. The UV-K5 option will successfully read from and write to the radio, meaning that it can be used to program the radio…*almost.* In order to really – fully – program this radio with the *egzumer* firmware and to set any or all of the new options that do not even appear under the standard UV-K5 driver, the user will need to use a version of *CHIRP* that supports the *egzumer* variant of this radio. This can be done through any one of three different ways, which will be discussed later.

Before getting into the programming process. I want to talk a little bit about the radio's specifications and its feature set, as these become important during the programming of the radio.

SPECIFICATIONS

I can address only those specifications that are publicly available. This includes any specs published in the owner's manual as well as any found online but attributable to the UV-K5(8) in some reliable and verifiable manner. This discussion will deal only with the specifications of the "FCC" or USA-sales version of this radio. The Owner's Manual makes reference to two other versions of the UV-K5(8), with the "CE" version having frequency ranges of 144~146MHz for VHF and 430~440MHz for UHF. The "NORMAL" version includes 136~174MHz for VHF, and 350~470MHz for UHF, split into two bands with the division at about 400MHz.

GENERAL SPECIFICATIONS	
Memory Channels	200
Stored FM Radio Channel Quantity	20
NOAA Channels	10
Frequency Stability	±1ppm
Modulation Mode	FM:11KφF3E(12.5KHz), 16KφF3E(25KHz)
Dimension	115mm x 60mm x 37.5mm
Weight	234g
Operating Temperature	-20℃+60℃
Antenna Impedance	50Ω

RECEIVER		
Sensitivity – FM (12dB SINAD)	F1 (50~76MHz)	-121dBm
	F2 (108~135.9975MHz)	-121dBm
	F3 (136~173.9975MHz)	-123dBm
	F4 (174~349.9975MHz)	-123dBm
	F5 (350~399.9975MHz)	-123dBm
	F6 (400~469.9975MHz)	-123dBm

	F7 (470~599.9975MHz)	-121dBm
WFM (20dB SINAD)	WFM (76~108MHz)	-110dBm
AM (10dB S/N)	F2 (108~135.9975MHz)	-113dBm
Audio Frequency	≥0.5W	
Audio Distortion	≤10%	

TRANSMITTER		
Specification	UHF	VHF
Frequencies	420~450<MHz	144~148MHz
Output Power	≤5W	≤5W
Emission Current	≤1.5A	≤1.5A
Maximum Frequency Deviation	≤5kHz (WFM)	≤5kHz (WFM)
	≤2.5kHz (NFM)	≤2.5kHz (NFM)
Modulation Distortion	≤5%	≤5%
Stray Power	≤7.5µW	≤7.5µW
Adjacent Channel Power	70dB (WFM)	70dB (WFM)
	60dB (NFM)	60dB (NFM)
Residual Modulation	40dB	40dB

Quite unusual among handheld radios, especially those in this price range, is the fact that apart from the frequency ranges offered, the transmitter specifications are the same for UHF and VHF operations. Usually, we see a difference, especially as regards the output power on the UHF band. This sameness shown in the UV-K5(8) is undoubtedly due to the radio being an SDR type as much as is possible throughout its architecture.

FEATURES

According to the Quansheng UV-K5(8) User Manual, the radio has the following features and capabilities:

- 200 Channel
- Cross-Band Intercom
- 1750HZ Call Tone
- Program On Site
- Emergency Alert
- FM Radio
- 10-Group Scrambler
- Multi-Scan
- CTCSS / DCS
- Remote Kill / Revive
- Keypad Lock
- Jacklight
- PC Programmable
- Busy Channel Lock
- VOX
- Dual-watch operation
- Time-Out-Timer
- Reset
- Large LCD Display
- Voice Prompt
- 10 Emergency Weather Channels
- Power-on Password Protection
- AM / FM Aviation Band Receiving
- Reverse Frequency Function
- Type-C and Charger-Base Charging
- Wide / Narrow Bandwidth
- Reminding Switch
- Multi Display of Channel No. / Channel Frequency / Channel Name
- Multi-Band Transmitting & 50∽600 MHz Receiving

- High-Capacity Battery / Long Standby Time
- Offset Frequency Direction Set
- Wireless Radio Replication
- Squelch Level Adjustable
- Repeater Forwarding Confirmation Function
- Frequency and Channel Mode Shift
- Offset Frequency Setting
- Channel Scan and Add
- DTMF Calling
- DTMF ANI
- DTMF Select Call (single call, group call, all call)
- Backlight Auto Off Time Selective
- Automatic Signaling Search
- Multi Step Frequency
- One Key Call Channel
- Fast Copy One Channel
- H (High), M (Medium), L (Low) Output Power Selective
- Receive / Transmit Code Setting Separately

Bear in mind the fact that the above list is a factory feature list. Earlier I said that I cannot comment on the factory firmware, and I am still not doing so other than to provide a list as shown in the manual and to say that some of these features have been supplanted by new features in the *egzumer* firmware. The firmware image in the radio is hard-limited at a total of 60kB. Due to the limited storage space set aside for radio personality within the firmware structure, in order to add some new features other existing ones had to be disabled. The disabled features list is actually rather short:

- ENABLE_AIRCOPY := 0 (disables Wireless Radio Replication);
- ENABLE_NOAA := 0 (disables 10 Emergency Weather Channels);
- ENABLE_VOICE := 0 (disables Voice Prompt);
- ENABLE_ALARM := 0 (disables Emergency Alert);
- ENABLE_TX1750 := 0 (disables 1750HZ Call Tone); and
- ENABLE_PWRON_PASSWORD := 0 (disables Power-on Password Protection).

The *egzumer* firmware replaces the above features with other desirable ones, though the level of desirability is a matter of personal opinion and taste. The menu structure also changes under the *egzumer* mod. The original menu contained a total of 51 items, which are all described in the user manual. While the *egzumer* menu keeps many of the original menu items, the menu has now grown to a 60 (69) item menu. I know – that is confusing. Which is it, sixty items or sixty-nine items? Well, the answer is *both.* The basic power-on menu contains 60 items. In addition, there is a rarely-used nine-item extension to the basic menu, which explains the *60 (69)* hedge. To access the menu extension, starting from a power-off state, press and hold both the **PTT** key and the **SK1** key while powering on the radio. The **SK1** key is the side key #1, located directly beneath the **PTT** key. The radio display screen will prompt you to *"RELEASE ALL KEYS"* in order to continue. When the keys are released, the menu will be active and in its extended version. Once launched in this manner, the extended menu will remain available for re-entry via the **M** menu key until the radio is powered off. After a normal power-up, the menu will revert to the standard 60-item menu.

Whenever the radio is placed in *Menu* mode, the menu position is shown at the bottom of the display screen, either an $x.60$ or an $x.69$ indication, with x being replaced by the menu item number of the current or active menu item. The user can scroll through the menu items in either direction using the up and down arrow keys, and the menu list will scroll "around the corner" when either extent of the list is reached. This is convenient when the user wants to access a high-numbered menu item directly after launching the menu. To do so easily, simply scroll down past the *1.60* or *1.69* indication to reach *60.60* or *69.69*, depending upon which menu has been launched.

I should probably also point out that the menu system uses a sort of reverse-hierarchy in that *1.60* or *1.69* is the *highest* menu item while *60.60* or *69.69* is the *lowest* menu item. This means that to scroll directly from *1.60* to *60.60,* the user would press the ▲ key, while to move from *1.60* to *2.60,* the user would instead press the ▼ key. While in *Menu* mode, specific menu items can be quickly reached by directly entering the menu item number into the numeric keypad. For example, to access the *Mic* (microphone sensitivity) menu item, which is assigned menu item number 31, the user would first launch the menu by pressing the **M** key, and then press, in turn, the **3 VFO/MR** key and then the **1 BAND** key.

Menu items, while scrolling through the menu, are selected by pressing the **M** key. Within that selected menu item, values are chosen by scrolling through those available until the desired value is reached, which is then confirmed by pressing the **M** key again. Scrolling is done via use of the ▲ and ▼ keys. The menu system can be closed at any time by pressing the **EXIT** key. Note that the **M**, ▲, ▼, and **EXIT** keys are also marked A, B, C, and D respectively.

The list below is the basic menu structure that exists under the *egzumer* firmware release v0.21. The grey highlighted text is the name of the menu item as it shows in the radio. In some cases, the existing submenus or major value options are listed. This is done in those cases where clarification is needed, as the menu item name is not fully descriptive of what that menu item will control. The number that prefaces each highlighted text snippet is the menu item number as found in the firmware. Those numbers can therefore be used to short-cut your way to specific menu items without the need to scroll through the list to get there.

***EGZUMER* Menu List**

1. Step - step of the frequency (in kHz), up/down buttons change frequency by this value, also you can only set a frequency that is multiple of half of this value.
2. TxPwr - radio output power (LOW/MID/HIGH)
3. RxDCS - receiver Digital-Coded Squelch, if you enable this, squelch will only unlock if this code is being received. You can start a DCS/CTCSS scan while you are in this menu option by pressing * SCAN button
4. RxCTCS - receiver Continuous Tone-Coded Squelch System, squelch will only unlock if this code is being received. You can start a DCS/CTCSS scan while you are in this menu option by pressing * SCAN button
5. TxDCS - transmitter Digital-Coded Squelch, radio will send given code while transmitting
6. TxCTCS - transmitter Continuous Tone-Coded Squelch System, radio will send given code while transmitting
7. Tx6ODir - transmitter frequency offset direction
8. TxOffs - transmitter frequency offset value
9. W/N - bandwidth used by transceiver
    a. WIDE - 25kHz
    b. NARROW - 12.5kHz
10. Scramb - scrambler, distorts the audio so it would be harder to understand for other listeners, if two radios use the same setting they can communicate

11. BusyCL - busy channel lockout, blocks radio from transmitting when signal is being received
12. Compnd - compander (compressor/expander), allows signals with a large dynamic range to be transmitted over facilities that have a smaller dynamic range capability, improves audio quality, both radios should use this option
13. Demodu - demodulator mode, default is FM, AM/USB can be used for listening only
14. ScAdd1 - add channel to scan list 1
15. ScAdd2 - add channel to scan list 2
16. ChSave - save current setting to a memory channel
17. ChDele - delete memory channel
18. ChName - modify memory channel name
    a. Use up/down keys to select a channel to edit
    b. Press the Menu button again to enter edit name mode
    c. Use up/down keys or digits (0 ~ 9) to cycle the letters etc.
    d. Press the Menu button to move to the next character position
    e. Repeat above two steps till you reach the end
    f. When "Sure?" pops up, press Menu to save, or Exit to cancel
    g. Press Exit at any time to cancel the edit and return to main menu.
19. SList - selects which channel is used by memory channel scanner
20. SList1 - channels assigned to scan list 1
21. SList2 - channels assigned to scan list 2
22. ScnRev - scan resume mode
    a. CARRIER - resume scan after signal disappears
    b. TIMEOUT - resume scan after 5 seconds pause
    c. STOP - after receiving a signal, stop the scan
23. F1Shrt - side button 1 short press function
24. F1Long - side button 1 long press function
25. F2Shrt - side button 2 short press function
26. F2Long - side button 2 long press function
27. M Long - menu button long press function
28. KeyLck - auto keypad lock option
29. TxTOut - max transmission time limit
30. BatSav - battery save option, a rate between active time and sleep time
31. Mic - microphone sensitivity
32. MicBar - microphone bar that appears while transmitting
33. ChDisp - channel display style
34. POnMsg - power on message
35. BatTxt - additional battery value on the status bar in % or volts
36. BackLt - backlight duration
37. BLMin - minimal backlight brightness, when the screen backlight turns OFF it will go dim to this value
38. BLMax - maximal backlight brightness, when the screen backlight turns ON it will turn bright to this value
39. BltTRX - backlight activation on TX or RX
40. Beep - keypad press beep sound
41. Roger - roger beep at the end of transmission
42. STE - squelch tail eliminator, eliminates noise at the end of a transmission
43. RP STE - repeater squelch tail eliminator
44. 1 Call - one key call channel, lets you quickly switch to the channel with 9 Call button

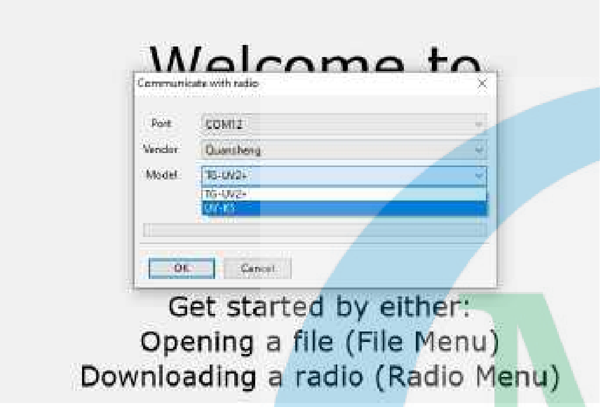
45. ANI ID - DTMF communication radio ID
46. UPCode - DTMF code that is sent at the beginning of transmission
47. DWCode - DTMF code that is sent at the end of a transmission
48. PTT ID - sets if UPCode and/or DWCode should be transmitted
49. D ST - DTMF side tone switch, lets you hear transmitted tones in the radio speaker
50. D Resp - DTMF decoding response
    a. DO NOTHING: do nothing
    b. RING - Local ringing
    c. REPLY - reply response
    d. BOTH - local ringing + reply response
51. D Hold - DTMF auto reset time
52. D Prel - DTMF pre-load time
53. D Decd - enables DTMF decoder
54. D List - list of DTMF contacts
55. D Live - displays DTMF codes received by radio in the middle of the screen
56. AM Fix - activates autogain AM fix function
57. VOX - voice TX activation sensitivity level VOX Setting
58. BatVol - battery voltage and percentage
59. RxMode - sets how the upper and lower frequency is used
    a. MAIN ONLY - always transmits and listens on the main frequency
    b. DUAL RX RESPOND - listens to both frequencies, if signal is received on the secondary frequency it locks to it for a couple of seconds so you can respond to the call (DWR)
    c. CROSS BAND - always transmits on the primary and listens on the secondary frequency (XB)
    d. MAIN TX DUAL RX - always transmits on the primary, listens to both (DW)
60. Sql - squelch sensitivity level

<u>Hidden Menu Items</u>

61. F Lock - sets the TX frequency band plan.
    a. DEFAULT+ (137-174, 400-470) - allows TX on default bands, plus options Tx 200, Tx 350, Tx 500
    b. FCC HAM (144-148, 420-450)
    c. CE HAM (144-146, 430-440)
    d. GB HAM (144-148, 430-440)
    e. (137-174, 400-430)
    f. (137-174, 400-438)
    g. DISABLE ALL - disables TX on all frequencies
    h. UNLOCK ALL - enables TX on all bands (it has additional lock, read a wiki on how to turn that on)
62. Tx 200 - enables TX on 200MHz
63. Tx 350 - enables TX on 350MHz
64. Tx 500 - enables TX on 500MHz
65. 350 En - enables RX on 350MHz
66. ScraEn - enables scrambler function
67. BatCal - battery calibration, measure the voltage on the back of the radio, and adjust the value in the menu accordingly
68. BatTyp - battery type, 1600mAh and 2200mAh battery has very different discharge curve, this is used to calculate battery level percentage

69. Reset - resets radio configuration settings
    a. VFO - removes only channel settings
    b. ALL - resets all radio settings

Now that we have explored the menu a little bit, we can begin to talk intelligently about the programming of this radio using *CHIRP.* As mentioned earlier, there are three options available for *CHIRP* programming of the UV-K5(8) *egzumer* radio. One method involves the use of a custom version of the *CHIRP* executable that contains support for the *egzumer* firmware-equipped radio. The second and third methods involve using an existing installation of *CHIRP* operating in *Developer Mode,* a procedure not highly recommended by Dan Smith, the creator of *CHIRP.* These two methods differ only as to the method used to load the necessary *.py* module to the application.



Welcome to

Communicate with radio

Port: COM12

Vendor: Quansheng

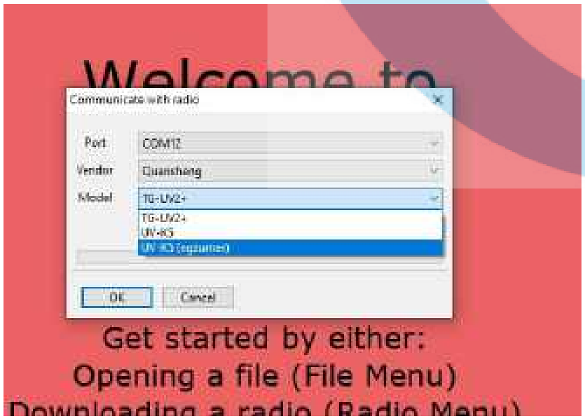
Model: TG-UV2+

TG-UV2+
UV-K5

Get started by either:
Opening a file (File Menu)
Downloading a radio (Radio Menu)

Figure 6 - Standard CHIRP support

Let's talk about the first method listed above as a beginning point. It must immediately be understood that while *CHIRP* has integrated support for some of the Quansheng radios (Figure 6), and it specifically supports the UV-K5, that route will not get you access to the new features provided by the *egzumer* firmware as it is installed in that radio model. As a work-around, the *egzumer* folks have provided a modified *CHIRP* executable that includes support for the radio carrying their firmware. Begin by pointing your browser to https://github.com/egzumer/uvk5-chirp-driver/releases. Under the *ASSETS* heading on that page, you will see a link to the *CHIRP* executable file named ***CHIRP_egzumer.exe***. Click on the link to begin downloading the file. Note that your operating system may inform you that the file could not be downloaded safely or some other such warning. In that case, you will need to right-click the three-dot *More Actions* menu to access your save options. Click *Keep* on the menu that appears. It is most likely that another box will show up warning you again. In this box, click the *More* link about halfway down the box. That will produce another dialog from which you can select *Keep Anyway* to finally save the downloaded file.



Welcome to

Communicate with radio

Port: COM12

Vendor: Quansheng

Model: TG-UV2+

TG-UV2+
UV-K5
UV-K5 (egzumer)

Get started by either:
Opening a file (File Menu)
Downloading a radio (Radio Menu)

Figure 7 - egzumer CHIRP support

To run the special *CHIRP* version offered by that downloaded executable, simply navigate to the location in which the saved file was stored, and then double-click on the file. *CHIRP* will launch, and you can then verify that you are running the correct version of the utility by simply reading the text on the title bar of the application window. If the correct version is running, the title bar will show *"CHIRP egzumer"* rather than just *"CHIRP*" as is displayed by the official *CHIRP* utility.

This version of the *CHIRP* utility will seem no different than the official version, except that when you open the *Radio > Download from radio…* tool and select *Quansheng* as the vendor, there will now be a third model from which to choose. The model *UV-K5 (egzumer)* will now be available. Select this model.

In order to continue with the programming, the programming cable must be inserted into the speaker and microphone jacks of the radio, and must be connected to an active USB port on the computer. By now, you should probably know which COM port the cable has pulled in the computer, as it is most likely the same port that you used when installing the *egzumer* firmware. Make sure that the radio is powered up normally and that the volume is set to at least the 50% point.

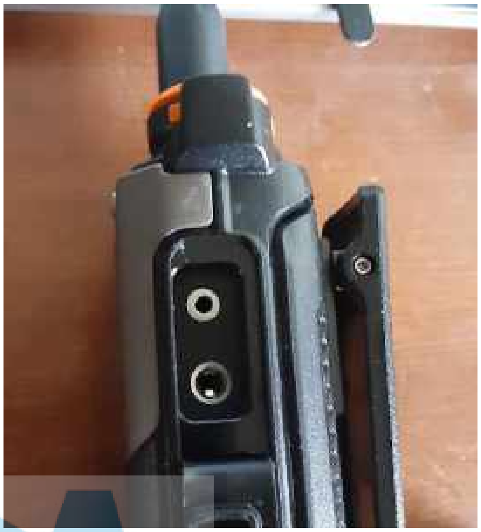
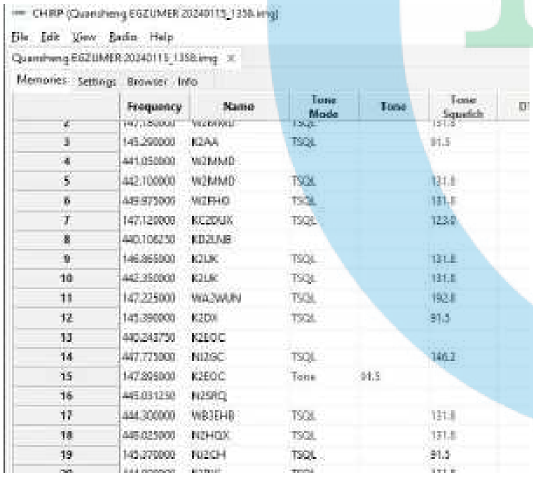


Figure 8 - Mic and speaker jacks on UV-K5(8)

I would like to take a couple of minutes to explain the reasoning behind setting the volume to a rather high level. It must be understood that all of this data communication that occurs during the firmware update and the programming of the radio is done using audio tones. The computer must be able to "hear" the radio's audio output in order to communicate successfully, and *vice-versa.* Thus, having an adequate audio level is key to making the acoustic connection that is so necessary to the process. The *CHIRP* utility automatically manipulates the audio output volume going to the radio, but the user must manage the radio's audio output volume going to the computer.

From this point onward, the programming of the memories is just run-of-the-mill *CHIRP* programming. You can use all of the normal *CHIRP* tools including copy and paste, querying data sources such as *Repeater Book,* and importing *.csv* data files. Remember to make a copy of the existing radio *.img* image file before you make any changes, just so that you have a baseline image to resort to if you do something wrong.

As always, it is also a good idea to make a copy of the completed file that you upload to the radio. Another word of caution is called for here as well. *CHIRP* will automatically generate an image file name when data is downloaded from a radio. That filename will basically consist of the radio make and model and the current date. If you attempt to save multiple images of the same radio make and model on the same day, you will receive a pop-up error warning you that that the file exists and asking if you wish to overwrite the file. *Do not* be tempted to be lazy and click the *Yes* button, because in doing so you will lose your earlier image file, which may be your baseline image for restoration. My habit is to append the file save time in the format *_HHMM* to the automatically-generated filename each time I save an image file. To boot, I have submitted a *new feature* request to Dan Smith to ask if this file naming could be made the standard convention for the software. In that manner, there will usually not be any overwrite issues to deal with, and no needed image files lost to an overwrite.



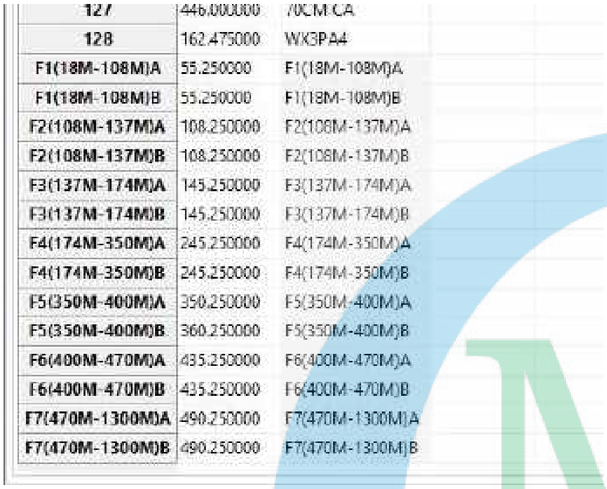
CHIRP (Quansheng EGZUMER 20240115_1358.img)

File Edit View Radio Help
Quansheng EGZUMER 20240115_1358.img ×
Memories Settings Browser Info

	Frequency	Name	Tone Mode	Tone	Tone Squelch
2	[illegible]	[illegible]	TSQL		[illegible]
3	145.290000	K2AA	TSQL		91.5
4	441.050000	W2MMD			
5	442.100000	W2MMD	TSQL		131.8
6	449.975000	W2FHO	TSQL		131.8
7	147.120000	KC2DUX	TSQL		123.0
8	440.106250	KD2UNB			
9	146.865000	K2UK	TSQL		131.8
10	442.350000	K2UK	TSQL		131.8
11	147.225000	WA2WUN	TSQL		192.8
12	145.390000	K2DX	TSQL		91.5
13	440.243750	K2EOC			
14	447.775000	NI2GC	TSQL		146.2
15	147.895000	K2EOC	Tone	91.5	
16	445.031250	N2SRQ			
17	444.300000	WB3EHB	TSQL		131.8
18	448.025000	N2HQX	TSQL		131.8
19	145.270000	NI2CH	TSQL		91.5

Figure 9 - TSQL vs Tone

For those who may be unaware of the difference, using *Tone Squelch (TSQL)* instead of *Tone* (Figure 9) when setting up your memory channels gives the added advantage (in most cases) of blocking potentially unwanted transmissions from being processed through your radio. We are all familiar with the CTCSS system, and we understand that a repeater using a CTCSS or PL tone is using that tone as a means of excluding uncoded transmissions from getting into the repeater. The radio transmits a basically sub-audible low frequency audio tone together with the outbound RF. The specific frequency of that tone is one of a total of fifty possible tones that are assigned to this process. The concept has multiple repeaters within radio range of each other that are using the same frequency pair being assigned different CTCSS tone frequencies. Thus, a transmission that has the correct tone frequency embedded will be allowed into the repeater while those transmissions with a different tone frequency or without a tone at all will be rejected. Not everyone understands, however, that this process also works in reverse. The repeater transmits the same CTCSS tone on its outbound transmissions, and if your radio is set to use Tone Squelch, it will expect that tone, accept transmissions with the proper tone, and reject those that are not carrying the proper tone.



127	446.000000	70CM CA
128	162.475000	WX3PA4
F1(18M-108M)A	55.250000	F1(18M-108M)A
F1(18M-108M)B	55.250000	F1(18M-108M)B
F2(108M-137M)A	108.250000	F2(108M-137M)A
F2(108M-137M)B	108.250000	F2(108M-137M)B
F3(137M-174M)A	145.250000	F3(137M-174M)A
F3(137M-174M)B	145.250000	F3(137M-174M)B
F4(174M-350M)A	245.250000	F4(174M-350M)A
F4(174M-350M)B	245.250000	F4(174M-350M)B
F5(350M-400M)A	350.250000	F5(350M-400M)A
F5(350M-400M)B	360.250000	F5(350M-400M)B
F6(400M-470M)A	435.250000	F6(400M-470M)A
F6(400M-470M)B	435.250000	F6(400M-470M)B
F7(470M-1300M)A	490.250000	F7(470M-1300M)A
F7(470M-1300M)B	490.250000	F7(470M-1300M)B

Figure 10 - Special Band memory Rows.

One thing that the *CHIRP* user must remember is that if you change any existing channels from using Tone to use Tone Squelch (or *vice-versa)*, the tone frequency must be set again, as the software stores these settings as separate values, and they are in separate columns in the user interface.

It is interesting that the *egzumer CHIRP* has some added specialty rows at the bottom of the memory channels list on the *Memories* tab (Figure 10). These rows store the currently-tuned (or last-tuned) frequencies for each of the frequency bands offered by the radio when operated in VFO mode, and there are frequencies stored for each of the two VFO slots. Note that in this aspect, the F7+ band is rolled into the normal F7 band, now showing that band to range from 470.00MHz through 1300MHz. Again, this is available for both the “A” and the “B” VFO.

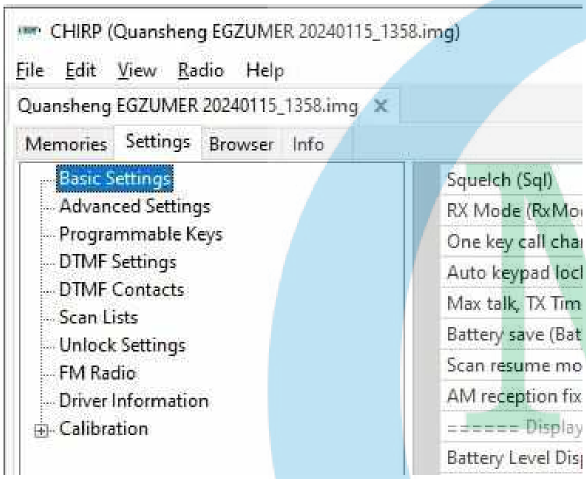
Another point to remember about *CHIRP* is that not all available columns are shown by default when *CHIRP* is launched. In order to make the hidden columns visible, select *View > Show extra fields* from the menu bar. The usefulness of the extra fields or columns is dependent upon the options that control those columns being made active.

Now that the *Memories* tab has been dealt with, let’s discuss the *Settings* tab. In the *egzumer* version of the UV-K5(8), there are ten sub-tabs, and the last of these has seven additional sub-sub-tabs. Not all of these tabs, however, will require any attention at all, or at most will require very minor attention. The bulk of the items in which the average users are interested will be on the *Basic Settings* sub-tab. Other sub-tabs that may interest some users are certain features on the *Advanced Settings, Programmable Keys,* and *Unlock Settings* sub-tabs. Let’s break it down a little bit.

Starting with the *Basic Settings* sub-tab (Figure 11), I made changes to each and every section of the sub-tab. Settings of primary interest to me were:

- Squelch (set to level 4);
- RX Mode (set to DUAL RX RESPOND);
- Max talk time (set to 3 minutes);
- Battery Level Display (set to PERCENT);
- Microphone Bar display (checked box to turn feature on);
- Channel display mode (set to Name + Frequency);
- Power ON display message (set to MESSAGE);

- Message Line 1 (set to my surname);
- Message Line 2 (set to my call sign);
- Backlight Time (set to 1 minute);
- Backlight level min (set to 2);
- Backlight level max (set to 10);
- Backlight on TX/RX (set to TX/RX);
- Voice-operated switch (set to OFF);
- Mic Gain (set to 2);
- Key press beep sound (checked box to turn feature on);
- End of transmission beep (set to ROGER);
- VFO A current channel/band (set to Channel M1);
- VFO B current channel/band (set to Channel M5); and
- Main VFO (set to A).


CHIRP (Quansheng EGZUMER 20240115_1358.img)
File
Edit
View
Radio
Help
Quansheng EGZUMER 20240115_1358.img
Memories
Settings
Browser
Info

Basic Settings
Advanced Settings
Programmable Keys
DTMF Settings
DTMF Contacts
Scan Lists
Unlock Settings
FM Radio
Driver Information
Calibration

Squelch (Sql)
RX Mode (RxMo
One key call chai
Auto keypad locl
Max talk, TX Tim
Battery save (Bat
Scan resume mo
AM reception fix
====== Display
Battery Level Dis|

Figure 11 - CHIRP settings area

It should be noted that the squelch and mic gain levels were set as starting points. They are both easily adjusted via the menu afterwards to dial the radio in to your liking and best performance. I later ended up with both of these values set to “3”, though the mic gain setting via the menu has a different format. In the *CHIRP* interface, the values are sequential levels from zero to four. However, in the menu system, the options are +1.1dB, +4.0dB, +8.0dB, +12.0dB, and +15.1dB. Thus, my mic gain was effectively set at +12.0dB. This value equates to a setting of “3” when viewed in *CHIRP*.

Next up was the *Advanced Settings* sub-tab. Here there are only four values to be set. One of these will only require changing if the battery is changed from the 1600mAh stock battery to the 2000mAh upgrade battery. Because these two batteries have different discharge curves, the type is set here to ensure accuracy of the battery charge percentage display. Going back to the top of the list, the value is *Frequency mode allowed* and the options are either *ON* or *OFF*, selected by a check box. The remaining values are for calibration of the S-meter at the S0 and S9 levels. For a VHF/UHF receiver, these values should be set to -130dBm for the S0 line and -93dBm for the S9 line.

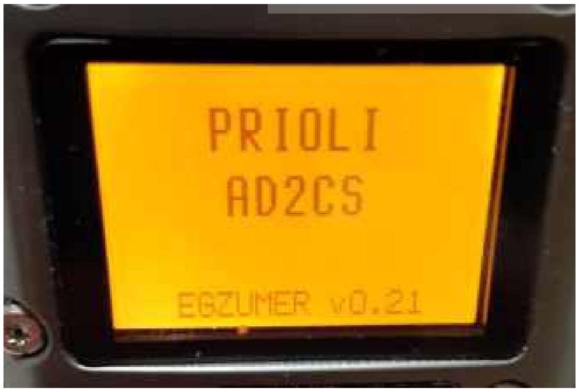


Figure 12 - Startup screen showing firmware version

The radio start-up screen will display the power-on message as entered under the *Basic Settings* area. In addition, the installed firmware type and version number are displayed on-screen at that time (Figure 12).

The next sub-tab, the *Programmable Keys* area, allows the user to customize the functions of the following keys:

- Side key 1 (upper side key);
- Side key 3 (lower side key); and
- Menu key.

The two side keys can be programmed for both short-press and long-press actions, and the Menu key can be programmed for a long-press action. The short-press action of the Menu key is reserved for launching the menu system. I chose actions for these key presses, but what I chose may not be right for other users. There is a drop list for each programmable key that offers the user a choice of eleven different functions. The same list is offered for each of the key actions.

I do not currently use DTMF on my handheld radios, so the *DTMF Settings* and *DTMF* Contacts sub-tabs were ignored. Also ignored was the *Scan Lists* sub-tab, as I simply do not use the scan function of most of my radios.

The next sub-tab of interest is the *Unlock Settings* sub-tab. This is the place where this radio can get you into trouble if you are not careful. On this sub-tab, the user can unlock the TX function on the 174-350MHz band, the 350-400MHz band, and the 500-600MHz band. It is also possible to unlock the RX function on the 350-400MHz band. Here, I changed the first value, the *TX Frequency Lock* line to a setting of *FCC HAM (144-148, 420-450)* as a means of keeping myself legal. I cleared the check boxes on the three TX unlock values, but left the 350-400MHz RX function active. Finally, and I *strongly* recommend that all FCC-licensed users do this, I set the *Scrambler enabled* value to OFF by clearing that check box.

The next sub-tab allows the user to pre-program a number of broadcast FM radio stations into channel slots, of which twenty are available. Just to test the functionality, I set up three Philadelphia stations to which I occasionally listen, one oldies station, one country station, and one classic rock station. These channels are populated by station frequency. I assigned the first three channels to my three stations, in ascending frequency order just because I am obsessive in that way.

The *Driver Information* sub-tab is next. This one is an information-only area, with no capability to edit the values. This sub-tab simply reports the data as it finds it.

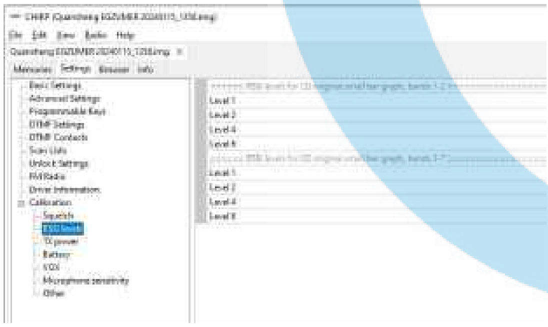


Figure 13 - Settings RSSI values

That brings us to the *Calibration* sub-tab. This sub-tab can also get the user into trouble, but mostly as regards proper functioning of the radio. It is critically important to understand that many of the values in the Calibration area are custom set at the factory to the individual radio *and are not the same for other individual radios!* Another point is that is also important to make a backup of the Calibration data as it was read from the radio. Finally, as a precaution, the Calibration data is not uploaded to the radio by default. In order for this data to be sent to the radio, the *Upload calibration* check box must be marked. When this check box is marked, *only* the Calibration data is sent to the radio, without any other programming data. It is equally critical to understand that you must *not* use a *CHIRP* configuration file from another radio, as the *CHIRP* configuration file contains a complete EPROM dump of the radio from which it was read. As a hedge against such problems, the *egzumer* folks recommend making a backup using a utility that does not depend on the *CHIRP* driver, such as *k5prog-win.* That utility is available online at https://github.com/OneOfEleven/k5prog-win/raw/main/k5prog_win.exe.

I cannot emphasize the need to back up the Calibration data strenuously enough. Because this data is at least in part unique to each radio, it is not possible to restore proper functionality by copying the Calibration date from another radio. The *egzumer* folks caution us to make changes in this area at your own risk!

As stated earlier, there are seven sub-sub-tabs found here. The first of these goes into the details of the specific values for each of the ten squelch “levels” selectable for each of the various frequency bands, broken into two band groups, that are covered by the radio. I left these settings alone, as it was too early to start changing these values, until I had some time in on the radio and had an idea of its performance.

The next area is the RSSI – Received Signal Strength Indication – area (Figure 13). These values affect the display of the signal strength bar graphs displayed on-screen for the two band groups as above. I saw no immediate need to adjust anything here, either, at this point in time.

The *TX power* sub-sub-tab lays out the individual transmit power factors for each of the seven bands covered by the radio. For each band, there is a section for the low-, the medium-, and the high-power setting, and for each of those three levels, there is a further breakdown into low, center, and upper areas of those levels. Not yet fully understanding the metrics in play here, I left all of this alone until I have a chance to gain a better understanding level. From a review of the data read from my radio, it is obvious that the values here go up as the power levels go up, but I have no real idea specifically what each or any value means in the measurable world.

What I *did* learn is that there is very little real information in the wild regarding the UV-K5(8) calibration values and their meanings. It is stated that the calibration data is raw data and is not equated to any units such as volts, watts, dB, dBm, and so forth. Because I do not understand all of it, I choose to leave it alone.

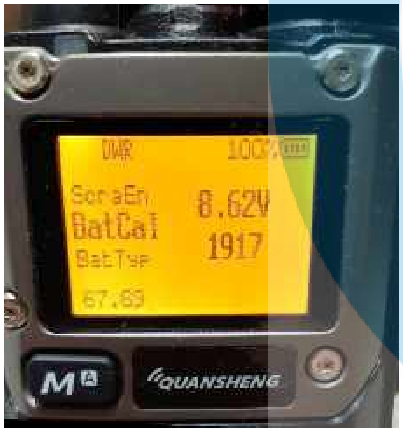


Figure 14 - BatCal menu item 67.69

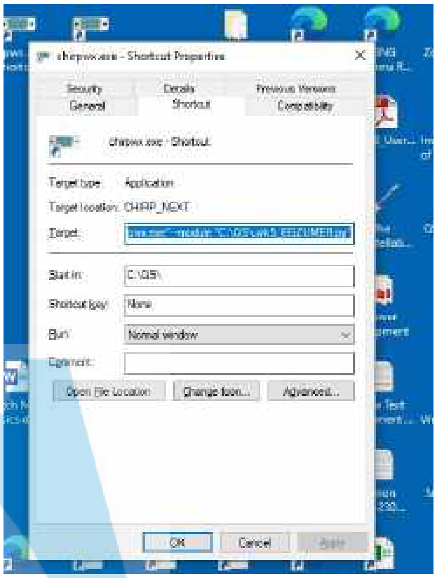
The next sub-sub-tab is that for the battery. One of the values is flagged as a battery calibration value. I *do* know where that value originates, but I am unsure how it relates to the other values other than to say that they are somehow derived from a combination of the battery calibration data and the battery type. One of the items in the hidden menu system, *67.69 BatCal,* is the source of the data on the battery calibration line. The menu notes tell us to measure the voltage at the back of the radio and to enter that value as the *BatCal* value under the *67.69* menu item (Figure 14). Changing the voltage level in that menu item produces a change in the displayed calibration value, which is also shown on-screen in the *67.69* menu item. That value coincides with the value shown in the *Battery* sub-sub-tab of the *Calibration* sub-tab.

After the battery values, we come to the *Microphone sensitivity* values. Here we see a list of five values, identified as lines 0 through 4, as we saw earlier when looking at the mic gain options. The values ascribed to these lines increment as do the mic gain values discussed earlier, though they are not the same values. Again, this is raw data of some sort and is not equated to any measurable units.

Finally, we arrive at the *Other* sub-sub-tab, which contains just three lines, the *Xtal frequency low* value, the *Volume gain* value, and the *DAC gain* value. Once again, we are looking at raw data with no simple way to equate the data points to familiar units.

All of this comes together to say *"If you don't care about bricking your radio, go ahead and play with this data."* My inclination is to leave it all to someone else to experiment with. I can be quite content at times in learning through the experiences of others.

Way back in the early parts of this article, it was mentioned that there were three different methods of getting *CHIRP* to communicate properly with a radio running on the *egzumer* firmware. We already discussed what is probably the more common method, but the other two methods are valid, so we will discuss them now. Both of these remaining methods involve downloading the *uvk5_egzumer.py* driver module, and then loading that module into the *CHIRP* instance. The driver module is readily available online at https://github.com/egzumer/uvk5-chirp-driver/blob/main/uvk5_egzumer.py?raw=1 and is quite compact at only about 83kB on the hard drive. In order to use this driver module with *CHIRP*, your *CHIRP* installation *must* be one of the *CHIRP-next* releases and not a legacy release. To reiterate, the driver module will not load into the legacy releases of *CHIRP.*


chirpwx.exe - Shortcut Properties

chirpwx.exe - Shortcut

Target type: Application
Target location: CHIRP_NEXT
Target: pwx.exe" --module "C:\QS\uvk5_EGZUMER.py"
Start in: C:\QS\
Shortcut key: None
Run: Normal window
Comment:

Figure 16 - Modified shortcut Target line

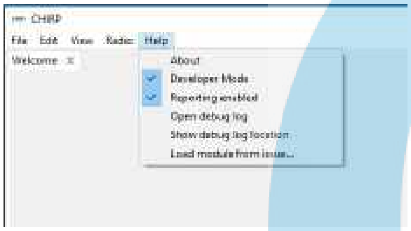


Figure 15 - Launch Developer Mode

Once you have downloaded the *.py* driver module, you will want to move the file to a specific location that you will create for it. As an example, you might choose to create a folder directly off the root of the C:\ drive, and then name that new folder *QS.* Whatever you call it and wherever that new folder is located, move the downloaded *.py* file into that new folder. Make a note of the folder name and location, as you will need it shortly.

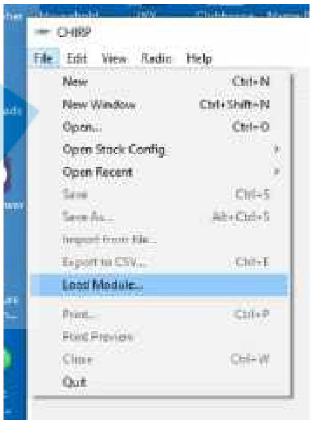


Figure 17 - Load module...

The first method to load the driver module is as a command line argument appended to the shortcut used to launch the CHIRP executable. Start out by locating the standard *CHIRP* executable (program) file in the folder where it was placed by the *CHIRP* installation routine. In most Windows® installations, the executable file will be found in *C:\Program Files (x86)\CHIRP_NEXT* and the filename is *chirpwx.exe*. Right-click on this file and select *Copy.* Next, navigate back out to the Desktop, find a clear spot on that Desktop, right-click there and select *Paste shortcut.* When the dust settles, right-click on the new shortcut and select *Properties.* The Properties sheet should open to the *Shortcut* tab. If not, select that tab now. On the *Shortcut* tab, click in the *Target:* line immediately to the left of the last quote mark. Insert a space there, followed by the text string *--module "C:\QS\uvk5_eqzumer.py"* and then click *Apply* and then *OK (Figure 15).* Note that there are *two* hyphens directly preceding the word *module* with no space after them. The syntax must be exactly correct or the program will not load the module and will instead return error messages.

Once *CHIRP* launches, you will notice that the interface background is an orange-red color. At this point, the *CHIRP* instance can be used in the normal manner, and that the application window title bar reads *CHIRP **Module Loaded***.

The final alternative for launching the *egzumer* support into a *CHIRP* instance requires that the user first put *CHIRP* into *Developer Mode* so that the driver module can be manually loaded into the application instance. Start out by launching *CHIRP* in the normal fashion. Once *CHIRP* is running, select *Help > Developer Mode (Figure 16).* NOTE: If a check mark is already displayed next to *Developer Mode* in the *Help* menu, then the program is already running in developer mode. In that case, skip ahead to the module loading step. If no check mark was already displayed there, go ahead and click the *Developer Mode* menu item. A warning dialog will be displayed. You must accept the warning by clicking *Yes* in order to continue. The next thing you will see is a notice that *CHIRP* must be restarted for the changes to take effect. Go ahead and close *CHIRP,* and then launch it again*.*

Once the *CHIRP* utility has been restarted in *Developer Mode*, it is time to load the driver module. To do so, select *File > Load Module…* (Figure 17) and accept the warning that will be displayed. Browse to the location of the stored driver module file, select the *uvk5_eqzumer.py* file, and then click *Open* to complete the module load. Once this is done, the *egzumer* version of the UV-K5 radio will now be available under the model list for the *Quansheng* vendor. From here forward, continue on with *CHIRP* in the normal manner.

<u>Analytic Testing of the UV-K5(8)</u>

In order for any article about this radio to be complete, we must also discuss the performance of the radio, both under actual usage conditions and under bench-test conditions. The full array of the bench testing done includes battery discharge rate testing, output power level testing, spectral purity testing, transmitted frequency accuracy testing, CTCSS tone accuracy and purity testing, and peak deviation testing.

Let’s start with the battery discharge testing. This test was made as a means of determining the anticipated life expectancy of a full charge on the 1600mAh factory standard battery. Testing was done on the basis of measured current draw by the radio under three different modes of operation: (1) idle in receive mode, (2) actively receiving an incoming transmission, and (3) transmitting. Further, the transmit mode was measured under the conditions of each of the three basic transmit output power levels. Current drawn by the radio under each of those operational modes was as follows:

- RX idle – 21mA;
- RX active – 78 mA;
- TX low power – 175mA;
- TX medium power – 205mA; and
- TX high power – 235mA.

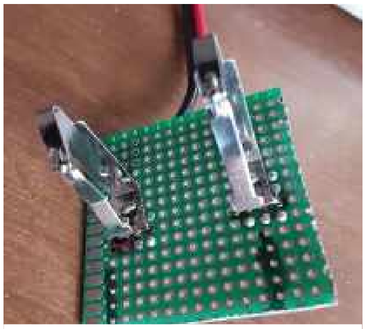


Figure 18 - Discharge cradle board

The standard battery that ships with this radio is a 7.2-volt, 1600mAh, 11.52Wh lithium-ion battery. This battery is best charged, according to the UV-K5(8) User Manual, through use of the 117VAC-powered charging cradle. However, the manual does permit use of the included USB-C cable to charge the battery via a standard USB port, or presumably, a common “wall wart” cell-phone type of charger. Charging in this manner is recommended as usable only for “emergency charging”. It stands to reason that the battery cannot be brought to a state of full charge via the USB-C port as the USB system and its associated similar chargers are limited to 5VDC. The 5-volt charge is clearly insufficient to bring a 7.2V battery up to full charge. It will, however, restore enough of a charge to permit short-term use of the radio when the battery has been depleted to a low state of charge. In order to derive a reasonable "fully discharged" point, in the absence of any advice from Quansheng on this matter, we are going to fall back on some widely-accepted values. The battery in use in this radio, as already stated, is a nominal 7.2V battery, and it is of the lithium-ion technology. A basic li-ion cell produces a nominal 3.6 volts, so this battery is therefore at least two li-ion cells taken in series. I say "at least" because there may actually be a series-parallel arrangement of, say, four cells ganged in such a manner to produce the requisite current capacity. Not knowing the basic capacity of the underlying cells, we are forced to go with what we *can* deduce and call it two cells in series. The basic li-ion cell is widely accepted to be considered "discharged" when its open-circuit voltage reaches a voltage of 2.5 volts. Note that the loaded voltage will be slightly lower than that level.

Armed with that information, I set about the process of systematically discharging the battery under the various load levels discussed above. I used my Kunkin KP-184 electronic load center to do the job under controlled conditions. A modified charging cradle was used to make connecting the battery to the KP-184 easier and to do so in a non-destructive manner.

I began by harvesting the battery contact clips from the Quansheng charging cradle. I did so in a non-destructive manner so that I could later re-assemble the original charging cradle. Next, I cut a piece of proto-board to size and bored it to accept the battery clips (Figure 18). Finally, I soldered a pair of leads to the clips and assembled the whole thing back into the original charging cradle housing. This gave me an easy means to connect the battery to the load center for the discharge testing (Figure 19).

The test procedure involved placing a fully-charged battery into a connection with the KP-184, setting the KP-184 to the desired discharge rate per the list above. The battery voltage was monitored, with a target discharged open-circuit voltage level of 5.0 volts being sought. The testing required pulling the battery down to some voltage below the target open-circuit voltage, as the normal test procedure calls for allowing the battery to rest thirty minutes after discharge activity before measuring the open-circuit voltage. Experience showed that the approximate loaded voltage that would yield a 5-volt open-circuit voltage under those test conditions was about 4.3 volts.

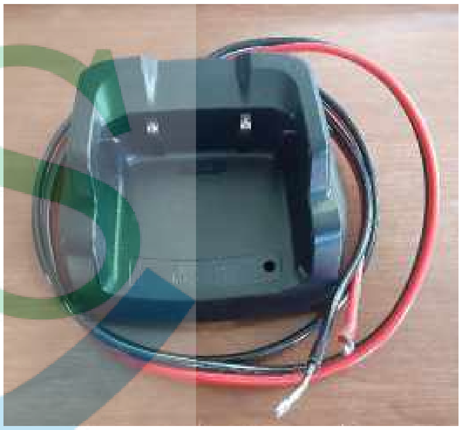


Figure 19 - Discharge cradle

The first test, which was the RX-idle mode at a continuous 21mA, took 116.5 hours to draw the battery from a full state-of-charge down to the 4.3V (loaded) level, which after the rest period produced an open-circuit voltage of 4.97 volts. This implies that the radio can operate in the RX-idle mode for more than four and three-quarters days.

The remaining tests, all performed in the same manner but imposing the different loads as per the current draw list above, produced the test results as below.

- RX active – 78 mA drawn took 31.25 hours to discharge the battery to 4.38 volts.
- TX low power – 175mA drawn took 13.75 hours to discharge the battery to 4.27 volts.
- TX medium power – 205mA drawn took 12.0 hours to discharge the battery to 4.4 volts.
- TX high power – 235mA drawn took 10.25 hours to discharge the battery to 4.29 volts (Figure 20).

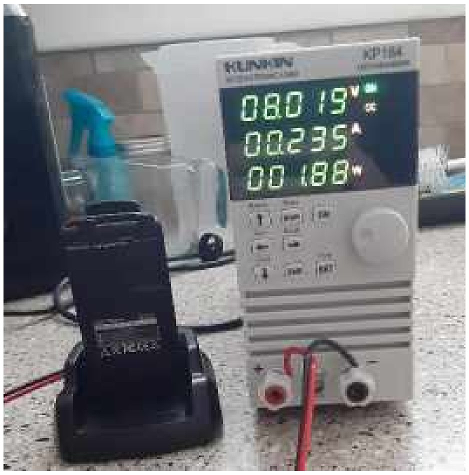


Figure 20 - Battery under 235mA discharge

As can easily be seen from the results above. The battery will provide some extended operational times for the radio. However, normal use of the radio involves some mix of the various operating modes. In order to more realistically test under a simulation of that operating condition, another test was made with varying current draw levels under a planned duty cycle. This plan included a mix of 60% RX-idle ops, 25% RX-active ops, and the remaining 15% was TX-high time. The test was a “rolling” mix of current draw values, rather than drawing one level continuously before switching to the next, and then finally the third level. The rolling mix was intended to more closely simulate real-life use of the radio. The end result of this test was that it took 35.25 hours to discharge the battery to the 4.25 volts (loaded) level, which recovered to 4.98 volts after the thirty-minute rest. This test would then indicate that the radio can be operated under similar conditions for almost a day and a half. Naturally, if the use mode mix were to be different, that time will change, but this was a good test of the durability of the battery in use.

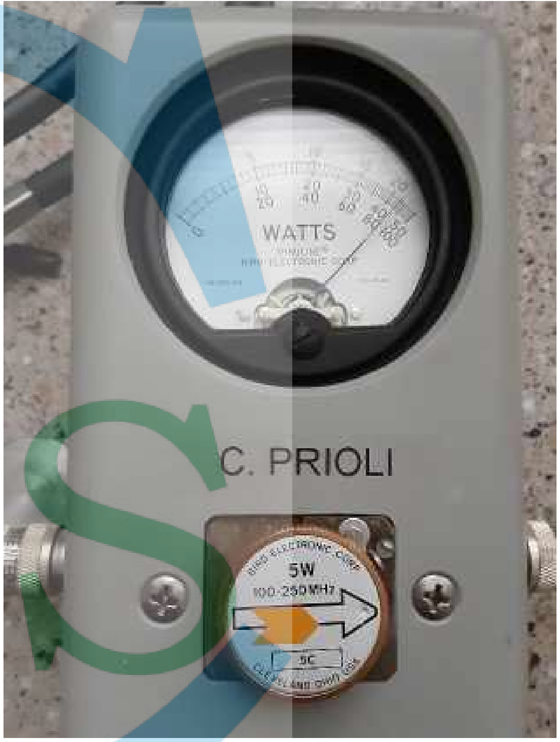


Figure 21 - VHF power output

The next test to be discussed is the output power test. This test was made using a Bird 43 directional watt meter and a 100-watt dummy load. Appropriate cables and adapters were used to connect the radio to the Bird meter. In each test, a 5W element was used, appropriate for the frequency range under test. This was a straight-forward test of the measured output power, with the results shown below:

- VHF –
  - low power – 2.4W;
  - medium power – 2.8W;
  - high power – 4.6W (Figure 21);
- UHF –
  - low power – 2.375W;
  - medium power – 2.95W (Figure 22); and
  - high power – 3.5W.

This test was simple and there is not much to say about it. The output powers are what they are. It is likely that they can be modified slightly by playing with the *Calibration* settings in the programming application, but I have to wonder if it is really worth risking the radio for a little more output power, especially if the existing power out is doing the job that you need done.

Spectral purity testing was done using a Siglent SVA1015X combination spectrum analyzer and vector network analyzer. While there were no wildly spurious spikes, spurs, or signals evident, there *was* some harmonic output on both bands. Testing on the 2-meter band showed the output frequency to be right on target, but the test did show the presence of the second harmonic at about -40dBm and the third harmonic at -45dBm. Similarly, testing on the 70-centimeter band showed good fundamental positioning, but the second harmonic at -42dBm as well as the third harmonic at -40dBm were again present. I would have expected better here, especially with the software-defined nature of the radio and the fact this type of spurious signal is easily handled by software filtering.

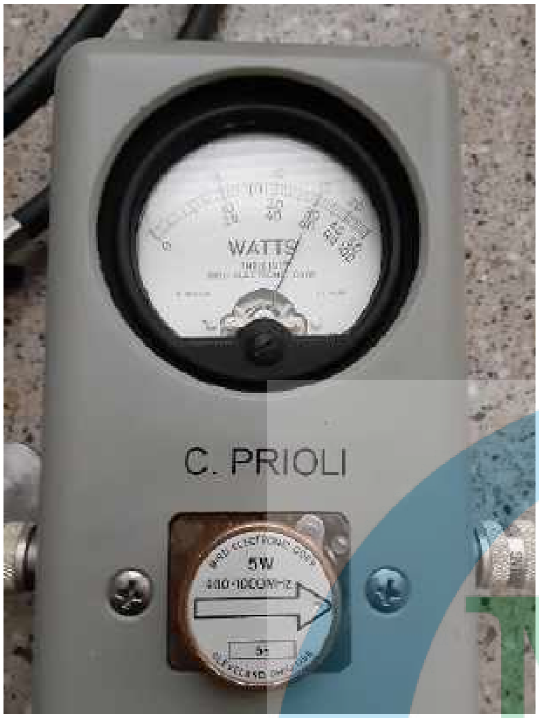


Figure 22 – UHF power output

The transmit frequency was again examined both on an oscilloscope and later with a frequency counter. An attenuated sample of the RF signal was pulled and fed to my Siglent SDS1202X-E 200MHz digital storage oscilloscope and also to the BK Precision 1803D frequency counter. Both instruments agreed within about 5 hertz as to the transmitted signal's frequency on 2 meters, and within about 15 hertz on the 70-centimeter band.

While the oscilloscope was set up, the presence of the CTCSS tones on both bands were confirmed, with perfect accuracy being noted. In this case, both programmed frequencies under which the tone test was made, the GCARC 2-meter and 70-centimeter repeaters, were carrying the 131.8Hz (#21) tone. As a check of the radio, I modified the settings to look at a low of 67Hz (#1) and a high of 254.1Hz (#50), and both of these frequencies were on the money as well. As such, I would then anticipate that the entire range of CTCSS tones would be accurately produced by the UV-K5(8).

The final test made was the peak deviation test. This test was made twice, once using my very basic Heathkit IM-4180 FM Deviation Meter in conjunction with my CMS SA-03 0-65dB step attenuator, and also via the Cushman CE-4B communications monitor. While there were very minor differences noted between the test results, the results were close enough to be able to call them the same. The results were not quite what I anticipated based upon the specifications table. The specs call for a maximum deviation of ≤5kHz on the WFM (25kHz BW) transmissions and ≤2.5kHz on the NFM (12.5kHz BW) transmissions. What the testing showed was 4.2kHz on WFM VHF and 3.8kHz on WFM UHF. However, on both bands, the peak deviation for NFM was slightly higher than the 2.5kHz specified peak at just over 3kHz.

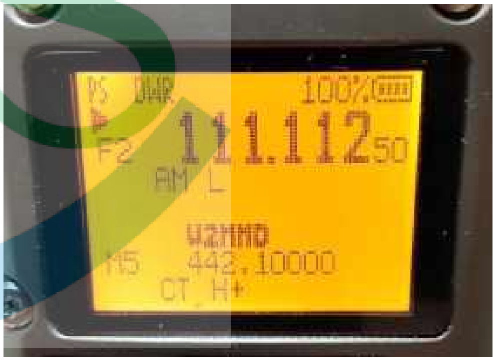


Figure 23 - AM demodulation set

Radio Operation

After the testing was done, it was time to just operate the radio and see exactly what it will really do. One of the selectable options is the capability to set the *Demod* or demodulation type to be either *AM (Figure 23)*, *FM*, or *USB (Figure 24)*. This setting can be changed on a channel-by-channel or in VFO mode on a VFO (either VFO A or VFO B) basis, using menu item *13.60*. In addition, for FM operations, the channel or VFO, depending upon operational mode, can be individually set to either *Narrow* or *Wide*, using menu item *9.60 (W/N).*

While the demodulation can be set to either of the three options listed above, it must be noted that these are ***demodulation*** and not ***modulation*** options. This radio can transmit *only* on FM. This is not really the drawback that it may at first seem to be. Sure… it would be great to have a thirty-dollar HT that can transmit USB or AM, but would we *really* get much use out of such a radio? I for one already have a radio that can be construed as a five-band five-mode handheld (as it is configured with its battery box). That radio has a maximum power output right in the five-watt area, as is the UV-K5(8), and it was a lot of fun *when I first built it!* But… the novelty quickly wore off, and I find myself no longer turning that radio on except when I am showing it to somebody. So… would we really get much use out of that capability if the radio did offer it? I tend to think not, but I could easily be mistaken.

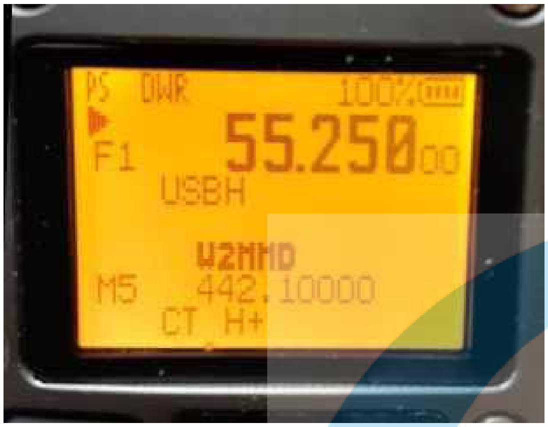


Figure 24 - USB demodulation set

The radio is comfortable to hold, and has a solid feel. The display is clear and easily read, with a wealth of information presented there. The *egzumer* firmware provides several display options for each channel in *Memory* or channel mode. The options include the option to display only the frequency, only the channel number, only the (channel) name, or the name & frequency together. This option is available at the *33.60 ChDisp* menu item.

The display backlight is adjustable across a range of arbitrary values from 0 to 10 via menu items *37.60 (BLMin)* and *38.60 (BLMax).* It can also be set to illuminate the display either not at all, on TX only, on RX only, or on both TX and RX. This is set via menu item *39.60 (BltTRX)*. Finally, the backlight time can be set to a value of *OFF, 5 sec, 10 sec, 20 sec, 1 min, 2 min, 4 min,* or *ON.* This value is set via menu item *36.60 (BackLt).*

Although the *CHIRP* programming routine has an option that would seem likely to be a default channel display for each VFO in memory mode, these values are instead the actual state of the radio when read. If this value is changed in the software and then uploaded to the radio, it will certainly make those channels the default displayed channels, but that value is a floating value. What this means is that in reality, the radio will start up after a power-down in whatever state the radio was left when it was powered down. Thus, if the user changes to a different memory channel before powering down, the radio will start up again on that new channel. Similarly, if the radio is placed into VFO mode before power-down, that is where it will return when powered up again.

Transmitted audio quality is good, and has repeatedly been reported as being a full and rich sound by listeners. As an afterthought to the testing done previously, I decided to check the radio's transmitted audio for harmonic distortion. This was not reported with the other testing as this was not directly a test of the radio *per se,* but is instead a test of the received signal transmitted by this radio.

The test process involved connecting the audio output of another radio receiver into the audio input of a calibrated Heathkit® IM-58 Harmonic Distortion Analyzer. The radio was placed in the receive mode on a 2-meter simplex frequency. The radio audio gain (volume) was set to approximately the 50% point. Next, I made several short transmissions of clean audio tones from my Siglent SDG1032X signal generator via the UV-K5(8). The audio tones used were 500Hz, 1kHz, 5kHz, 10kHz, and 15kHz unmodulated sine waves. I repeated the process twice more, using different receiving radios each time, for a total of three test sets. The radios used to receive the transmissions were a TIDRadio TD-H8, an Icom ID-4100, and an Anytone D878UVII-PLUS. The results of each test were noted on the IM-58 and compared for differences.

Across the range of transmitted and received signals, a maximum of approximately 2% total harmonic distortion was observed. While it is impossible to say if that distortion, or what part of it, came from each radio involved, the test nonetheless showed that when received by a range of different receivers, the audio quality was very close to the original modulating signal. I was hard put to find another more practical means of testing this, and I was very satisfied with the results obtained. Because of the fact that no speakers or microphones were involved – the signal was directly-coupled into the transmitter and the analyzer in every instance – any distortion present was a function of the electronic aspects rather than the physical aspects.

The *egzumer* firmware provides for the following “normal” frequency band set:

- Band F1 – 18.0000 MHz to 108.0000 MHz;
- Band F2 – 180.0000 MHz to 136.9999 MHz;
- Band F3 – 137.0000 MHz to 173.9999 MHz;
- Band F4 – 174.0000 MHz to 349.9999 MHz;
- Band F5 – 350.0000 MHz to 399.9999 MHz;
- Band F6 – 400.0000 MHz to 469.9999 MHz; and
- Band F7 – 470.0000 MHz to 999.9999 MHz).

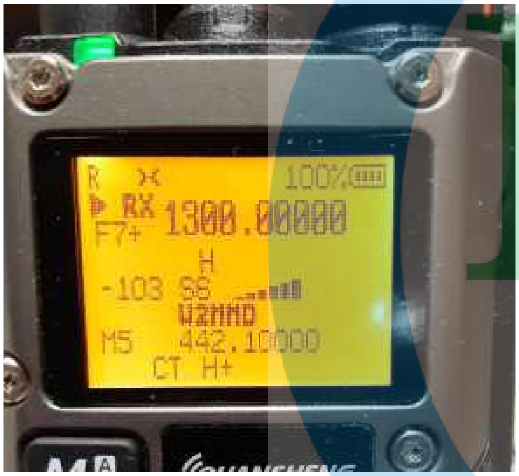


Figure 25 - Band F7+ display

In addition, there is another band, labeled as Band F7+ and starting at 1,000.000 MHz, that goes up from there with an upper limit of 1,300.000 MHz (Figure 25). It is very unusual to see this array of frequencies available on a low-cost HT, and the majority of these frequencies are RX-only frequencies. When the TX options are set to the FCC ham bands, the only frequencies on which transmission is permitted are from 144.0000 MHz to 148.0000 MHz and from 420.0000 MHz to 450.0000 MHz. The *egzumer* firmware provides for toggling through the bands in sequence by repeatedly pressing first the **F** and then the **1 Band** keys. Each time that key combination is pressed, the radio will advance the band progression from the current band to the next-higher band.

The range of the radio is about what one would expect from a five-watt handheld on the 2-meter or 70-centimeter bands. While this it completely unscientific and arbitrary, and is widely subject to conditions, I generally have no trouble reaching the GCARC repeater from inside the Clubhouse, a distance of 4.27 miles in a straight line. I am able to reach the repeater on either the 2-meter or the 70-centimeter band, and these repeaters are co-located at their Pitman, NJ location.

The antenna provided with the radio is the ubiquitous “rubber ducky” antenna (Figure 26) rated, according to the tag on its base, for 136-174MHz and 400-480MHz. It uses a female SMA connector on the antenna, leaving the male half, with the pin and the female threads, as the half on the radio. Note that this is *not* a reverse-polarity connection, though it is opposite of the more common method of installing the female side (with male threads) to the radio.

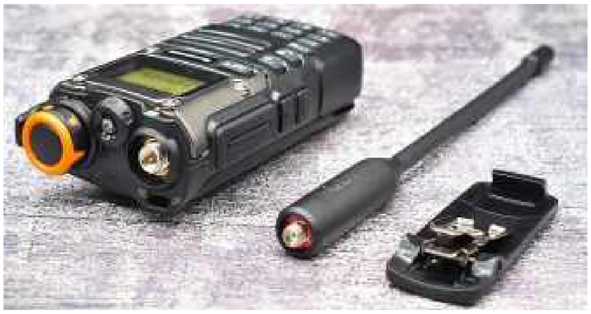


Figure 26 - Antenna base illustration

The “flashlight” or, as it is sometimes called, the “jacklight” on top of the radio is interesting. Though I have never used one of these lights for any reason, this one does have a rather unique feature. While most of these lights on this type of radio will operate in either one of two modes – either “on steady” or “on flashing”, this one operates in a third mode. The first press of the key that is programmed to the flashlight will turn it on to a steady state. The second key press will make it flash rapidly. The *third* key press, however, will cause the light to flash *“SOS”* in Morse code. As is usual, a final key press will turn the light off.

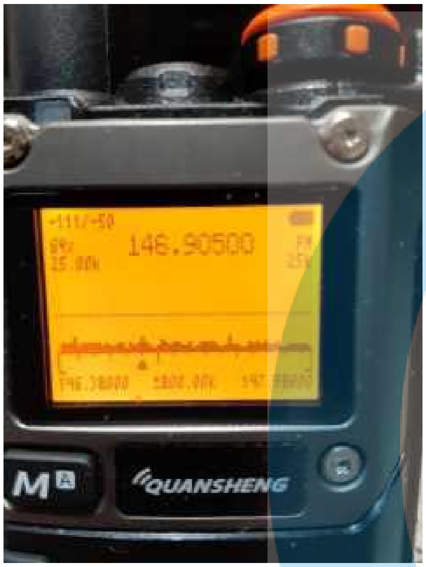


Figure 27 - Spectrum analyzer display

At some point in the life history of this radio model (or family), it was noticed that there was an apparent problem with the operation of the AGC system when AM demodulation was selected on any given channel or frequency. Under strong incoming signal conditions, the AGC locks in at index -1 and no further AGC action occurs. This can be corrected by toggling the *Monitor* function, which restores normal AGC operation until the next strong signal is received. However, as a better alternative, menu item *56.60* (*AM Fix*) can be activated. When this option is set to *ON,* the AGC system is disabled when the demodulation type is set to *AM.*

I could go on to describe some of the other optional settings, like the tuning step, or the DTMF options, or any one of several others, but instead, I will wrap this up by bringing up one of the most popular features of the *egzumer* UV-K5(8). This radio has a pretty nice “spectrum analyzer” feature (Figure 27), with multiple display settings and/or options. While it *does* take some getting used to when it comes to reading and understanding the display, it is nonetheless a useful and fairly accurate feature. A concise quick-start guide to this feature is available at *https://github.com/egzumer/uv-k5-firmware-custom/files/13242741/QuanSheng.UV.K5.Spectrum.analyzer.guide.EN.pdf*. This guide, as accessed via the above link, is a translation from the original Spanish document, and as a result, the embedded captioning in the graphics remains in that language. However, it is actually quite easy to intuit the meanings of the various captions either from the words themselves, from the context, or from a combination of the two.

Browsing to https://github.com/egzumer/uv-k5-firmware-custom/wiki/Spectrum-analyzer will bring the user to the *Github* page covering the spectrum analyzer feature. That page offers some condensed views of the SA display in its two basic modes of operation. There is also a summary of the radio button functions for each of the operational modes located on that page.

All things considered, and we considered many things here, this is a remarkable radio for its price. Of course, I cannot speak to the factory firmware, but in the *egzumer* iteration, there is far more good than bad about its features and operation. I would not hesitate to recommend this model – and this firmware – to anyone who is looking for a capable HT for everyday use. Further, I would also recommend the *egzumer* UV-K5(8) to anyone who is in the market for a good off-the-beaten-path radio for exploring some bands outside the normal FCC amateur bands.