

VK3CV 122GHz display

Modified Arduino sketch functionality

Mark Hughes GM4ISM

As per Barry's G8AGN's original sketch on which this is based, the software allows us to display the status of the VK3CV 122GHz 'transverter' by reading and decoding the various switch settings.

There are slight modifications to what is displayed in some switch conditions, (eg Indicating Full Duplex and renaming LO as TX in that condition)

In addition it displays supply voltage(s) using Arduino pins A6 and A3 and the status of whether the RX frequency is high or low of the Local Oscillator is calculated. Arduino pin A4 is an output that can be used to control a switchable Quadrature combiner. It will have the status of low impedance to ground or high impedance, dependant on whether Rx is Hi or Low of the Local Oscillator.

A CW hang time on the display updating of the KEY line has been added to improve the display with external 'hand sent' CW.

2 pushbuttons are implemented (**INC**rement and **Ent**er) to allow user interaction with the software. Arduino pins A7 and A5 are used.

The Arduino can be used to control the channel selection. It can detect if the channel switches are already set and won't assert control unless external channel switches are selecting CHO (all open). It also relinquishes control if any of the channel switches are set differently from the software commanded channel.

In control mode, the Arduino presents either a high impedance (~ 100 MOhms) relying on external pull-ups on the VK3CV board, or a low impedance to ground.

There is a CW auto keyer function, able to send looped CW messages in 'standard' QSO format with callsigns up to 11 characters long. Message string elements can be easily changed without a PC. The format is hard coded.

The Maidenhead Locator for the keyer is set by the GPS. If GPS is not present it can be manually set. It will be overwritten when GPS lock is achieved (but not mid message)

To save space in programme memory, the CW character definitions are stored in EEPROM. The sketch loads these into EEPROM at startup but will not overwrite stored CW message strings.

The use of EEPROM.update ensures that the EEPROM is not actually overwritten at every boot.

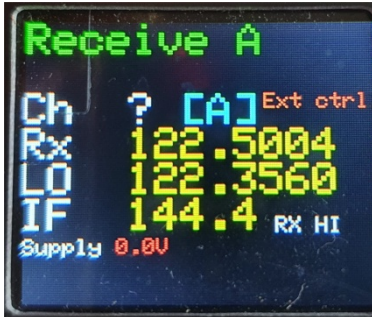
It is possible to save space in the compiled sketch by commenting out the call to the EEPROMload() function in void setup() but it MUST be allowed run once on a new Arduino.

Both the CW Keyer and channel control are accessed by the **INC** and **Enter** buttons. Button functions are displayed to the user at startup.

Please note, the CW keying transistor has a max collector current of 100mA and collector voltage rating of 45V. It should be able to drive a small relay or most transceiver Morse Key input via the breakout jumper on the board. If driving a relay, there is no protection against back EMF from the relay coil, this should be fitted externally.

## Channel setting

To set the channel from the display, 'long press' on **Enter** will cause the channel number (must be 0) to be replaced by a white '?' pressing (or holding) the **INC** button will increment the channel number by 2 (only even channels are selectable on the VK3CV board, odd Channel Tx Rx pairs are always the inverse of the preceding even channel and are accessed using the A/B switch)



When the desired channel is reached, a press on the **Enter** button will assert channel control and return the display to 'monitoring mode'. If **Enter** is not pushed, this happens after 5 seconds.

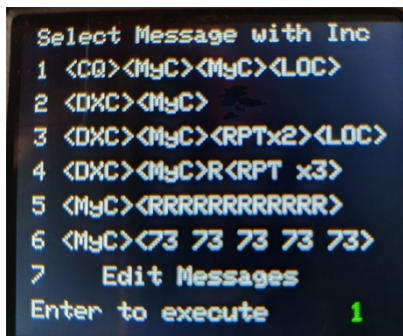
The Ext ctrl message displayed will disappear.

If external channel control switches RB2, 3 & 4 (eg the BCD switch on the 122GHz Board) are not open cct (ie a channel other than 0 is selected), the Arduino will not attempt to assert control and will display a warning that the channel is under external control.

If the display unit is in control of the channel and an external channel switch (that is not already pulled low) is changed, the Arduino will detect this and relinquish control to the hardware channel switches and simply read the switch states.

## CW Keyer

The built in CW keyer has six fixed format messages that can be selected from a menu that appears when **Enter** is 'double clicked'. The 6 messages format is that of a 'standard' QSO exchange, starting with CQ or 2 calls, then calls and report, calls and R Report, RRR and lastly 73s



```
Select Message with Inc
1 <CQ><MyC><MyC><LOC>
2 <DXC><MyC>
3 <DXC><MyC><RPTx2><LOC>
4 <DXC><MyC>R<RPT x3>
5 <MyC><RRRRRRRRRRRR>
6 <MyC><73 73 73 73 73>
7 Edit Messages
Enter to execute 1
```

Once you enter this menu, there is no timeout (lack of code space) if you want to exit quickly, ensure the menu item is one of the 6 messages and hold the **Enter** Key for a couple of seconds, the CW Keyer will launch and then exit.

The menu is accessed by 'double clicking' the **Enter** button. **INC** then selects the menu item.

Messages are composed of user define elements. Callsigns, Report and locator (if not set by GPS) can be entered via menu item 7. These are then stored in EEprom.

The display will indicate which message element is being edited and display the present string, flashing the cursor at the first letter. **INC** will change that character one ASCII value at a time, allowing only letters, numbers, / and : The colon represents a CW word space.

Once the desired character is indicated, **Enter** will move the cursor to the next character.

Continuous pressing the **Enter** button will skip through the message element characters reasonably fast. When you reach the string length limit, an arrow will appear at the end of the line. Release the button and the next message element will be ready to edit.

Callsign elements are up to 11 characters (blank characters are simply skipped)

The Report element is up to 7 characters to be able to add a contest serial eg 529:003

Letters are allowed here (as much because there is so little space left in memory as the desire to be able to send 55S to indicate 'Scattered' tones)

The last maidenhead locator stored can be overwritten manually if GPS is not present.

Only 8 digit locators can be stored and sent via CW

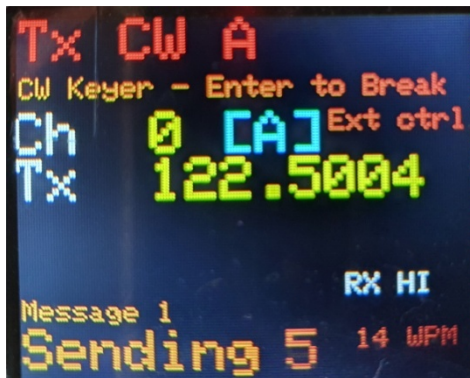
The last setting is CW speed, which can be set between 6 and 20 WPM

Once all message elements have been updated (or kept as is), the CW menu will be displayed again

Select the CW string you want to sent with the **INC** button and hit **Enter**

The keyer will then send the CW using the open collector keying cct which will normally be looped back to the Key line of the display and 122GHz board.

The keyer will sent the message string in a loop until it times out after 5 minutes, or keying is interrupted with the **Enter** button. Please note that Display of the status of the switches cannot update during the keying sequence. This CW keyer is not intended to be used while in FM mode, The VK3CV board will interpret PTT and Keying as a command to enter CW Beacon mode.



The 122GHz 'transverter' internal beacon keyer can be accessed in the normal way and is completely independant.

Another minor tweak to the original sketch provides a hang time on the monitoring of the Key line. This prevents the display constantly flashing if you dare to plug a Morse key in and actually sent CW by hand.