

W0EB/W2CTX Software Operating Manual for the MicroBITX Transceiver Using The BITEensio Micro-Controller with a Color TFT Touch Screen Display Manual Version 1.30

Copyright © by W0EB/W2CTX, Aug.24, 2018, All Rights Reserved.

(This document may be freely distributed as long as no changes are made to it and
the authors are given full credit for their efforts.)

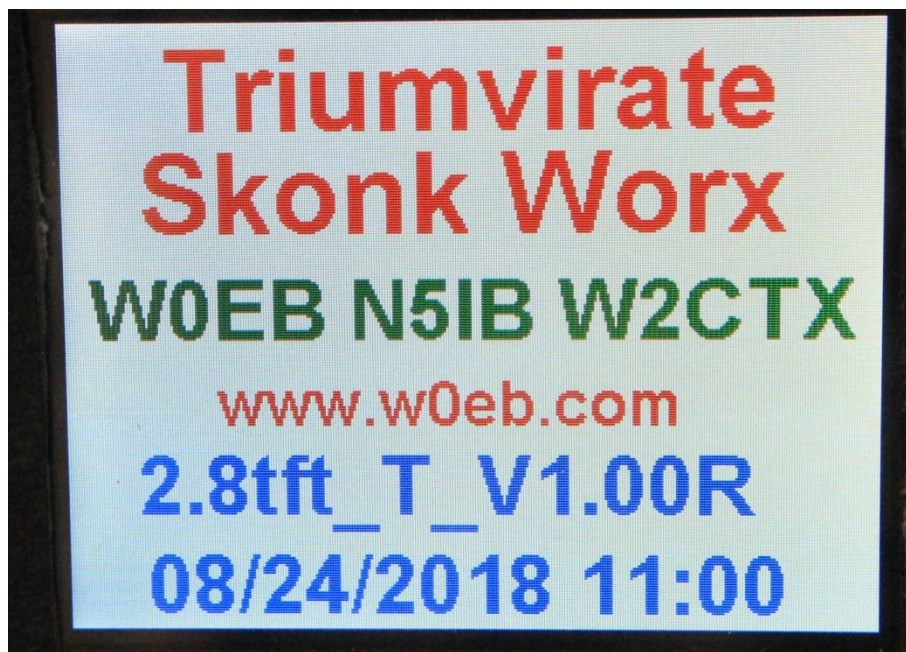


Photo of the 2.8 inch Color Touch Display Prototype's "Splash Screen"

INTRO:

This software operating documentation will be somewhat different than our other offerings. Since changing to the Teensy 3.6 MPU from the Arduino Nano, we will assume the user has the ability to download the files and program the MPU using the Arduino IDE with the “TeensyDuino” add-ons installed. We will no longer include those instructions in these documents and the published manual’s version# will not necessarily reflect the version # of the software it refers to.

New revisions of this document will be issued only when sufficient changes to the normal operation of the software defined herein have been made to warrant such changes. Minor revisions to the software will be documented in a “change log” README file that will be included in the ZIP archive for a particular version and will be available as soon as practical in the Documentation download directory of the files link at www.w0eb.com.

Every effort will be made to ensure that website has the latest versions of our software available for download within a reasonable time after release.

Hardware considerations and items necessary for operation:

There is a power jumper on the backside of the Teensy 3.6 as shown on the card that came with your Teensy. It says this should be cut to separate VIN (Teensy power) from VUSB (power supplied through the USB port) if you are using external power. You MUST cut this jumper before first powering up your BITEnsio card!!!! This will keep BITEnsio power to the Teensy from conflicting with power from the USB port and possibly destroying your Teensy!

W0EB/W2CTX/N5IB “BITEnsio” Controller board with a Teensy 3.6 MPU containing at least a 1 gigabyte microSD card (16 gig max) formatted FAT32

ILI9341 based 2.8” Color TFT touch screen display (available from eBay, Amazon and other places. (search for 2.8" SPI TFT LCD Display ILI9341)

Fabricated Shielded cable to match the connector on the display and the header connections on the BITEnsio. Details on this cable and all necessary jumper connections are contained in the BITEnsio construction manual available for

download on the <http://www.w0eb.com> website in the “Files/Documentation” section.

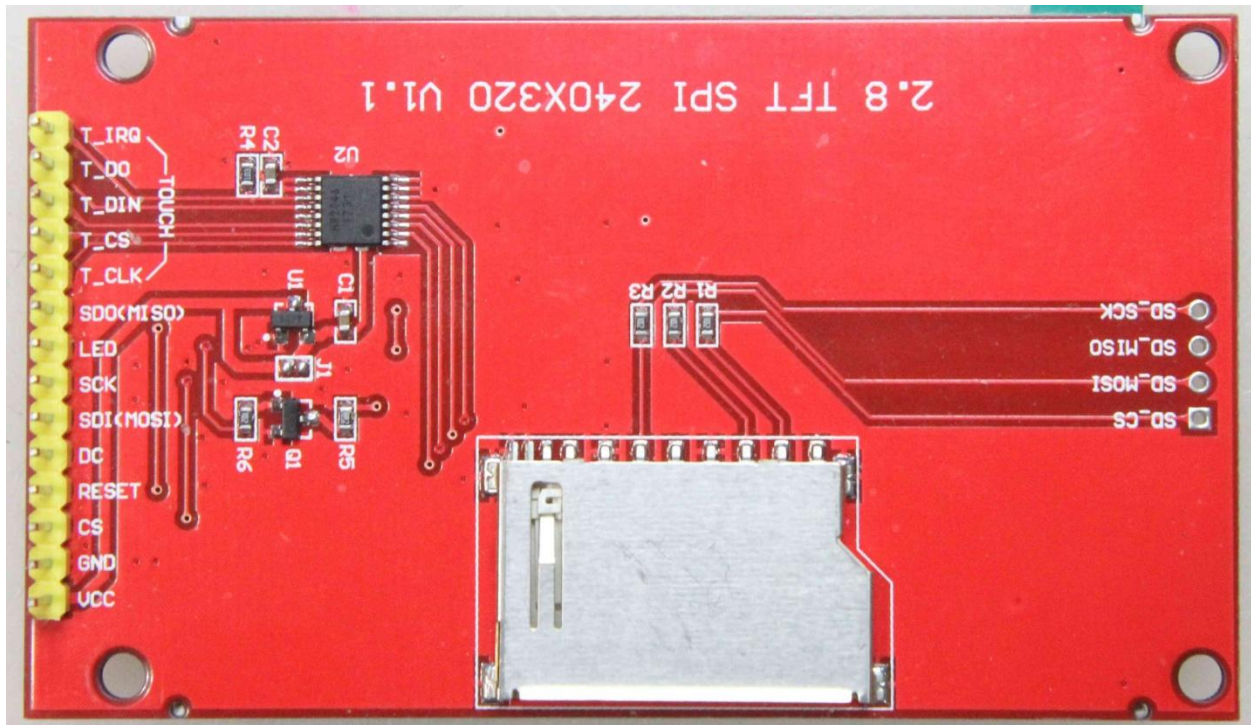


Photo of the rear of the display showing the 14 pin connection header.

The original tuning encoder supplied with the uBITX kit will be the only encoder used with this software. The encoder's built in function button will also be used (this switch may be paralleled by a discrete push button if desired for convenience).

All the other push button functions will be implemented as “Touch” buttons on the Color TFT display.

The display may be small but using the supplied stylus (or some other rounded point object made from wood or plastic so as not to damage the display) is very easy. Using this smaller display allows the uBITX to be built into a fairly compact case and still have many, many functions that would otherwise require extra encoders, pushbuttons and toggle switches.

The two VFO readouts (VFO A and VFO B) are displayed about the center of the screen. The active VFO is colored Orange and the inactive one is colored Light Grey. On the Active display, you can easily select the tuning rate by tapping on the digit you want tuned by the encoder. The digit to be actively tuned will be indicated by a green colored “Underscore” cursor and this cursor can be moved to any digit (except the far left “Tens of MHz” digit) as the rate with the MHz digit selected is more than fast enough to cover anything needed. You can also tune the selected digit with the green UP or DOWN pointing arrows to the right just by clicking on one or the other. Holding either arrow for a short period will switch to “repeat” mode and continue tuning the digit in the selected direction until the arrow is no longer being touched.



(The green “Tuning Arrows” and the Band UP/Down arrows aren’t shown in the following photos because they were added to the software after this manual was completed.) The frequency of the active VFO is saved to memory approximately 4 seconds after it has been changed during tuning if

no further change is detected. This keeps unnecessary memory writes to a minimum. If the frequency has remained unchanged for at least 4 seconds and the uBITX is turned off, that frequency will be the one set when the radio is again turned on.

There are various selector buttons that can be touched to activate their functions and they are (starting with the rectangular box in the upper left):

Memory Bank Selector (CW shown). The Banks are CW, SSB, DIG and USR. These represent 4 banks of 10 channels each which have been pre-loaded with the QRP calling frequencies for each band the uBITX can currently be operated on.

These banks are accessed by first touching the displayed band (left hand item) which is currently set to the CW QRP bank and this will activate a “drop down” list of the 4 banks as shown in the photo below. Tapping again outside the box will close the drop down box and leave the bank unchanged.



Each of these banks contains a list of ten sets of frequency and mode channels each of which can be selected by touching within the lettering for that channel from the left side (about the center of each line) through approximately the first third of the line.

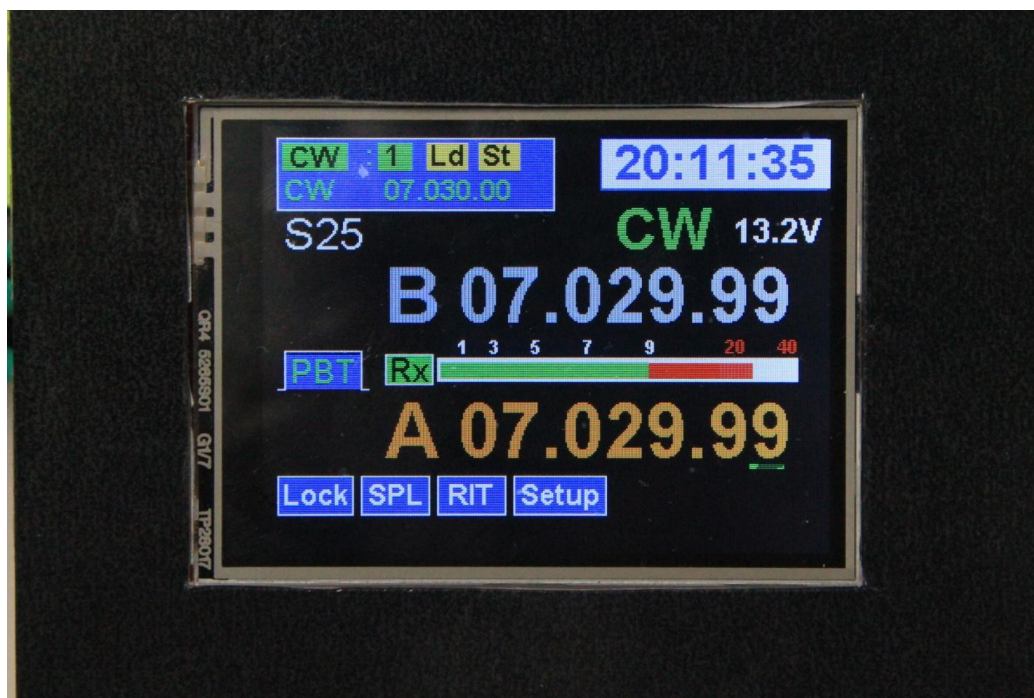
A quick tap seems to work much better than actually pressing on the area.

The drop down channel list for the CW bank is shown in the photo below and the lists for the SSB, DIG and USB banks are similar.



The “Band” up/down arrows will cycle through the channels of the selected bank (CW, SSB, DIG or USB).

Once you have selected the channel, its frequency and mode will appear on the bottom line of the display rectangle in the upper left. To load it into the “active” VFO, simply tap or lightly press the Ld button shown below. Be careful not to press the St (Store) button or whatever frequency and mode is actually in the active VFO will be stored and overwrite the frequency and mode for the channel you selected previously. Although the default banks are set at compile time, any one of them can be overwritten. The only thing that cannot be changed (at least in this version of the software) is the bank’s name (CW, SSB, DIG or USB). We recommend leaving the first 3 banks at the default values and only changing the USB (user) bank though this will be a matter of personal preference.



Immediately below the box in the upper left is the Letter S with the number 25 displayed. This indicates the current Keyer speed in words per minute.

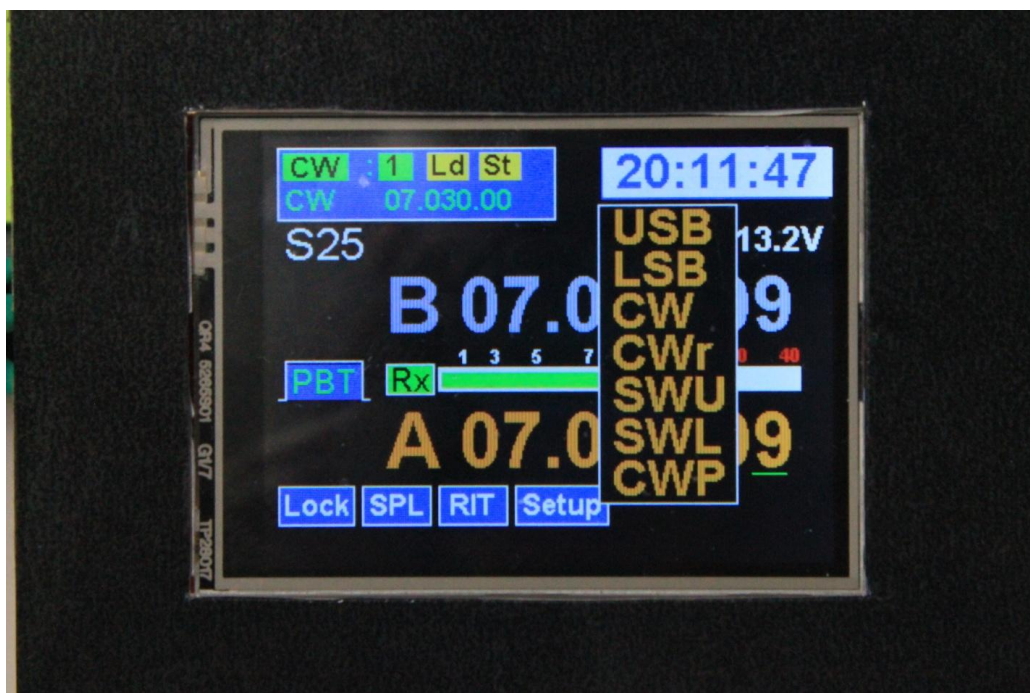
To change the speed, tap somewhere on the number portion of the speed display and they will then turn the color Orange indicating they may now be adjusted by using the encoder knob. The speed range can be varied from 5 to 50 words per minute and once you have the number set to the speed you want, simply tap or press lightly in the number area to accept the speed change and resume normal operation.

The White box with blue numbers is the display for the Real Time Clock. The time may be set either by the W2CTX remote control program, (the RCP and its instruction manual may be downloaded from the website at (http://www.w0eb.com/uBITX_RemoteControlFiles/W2CTX_RCP.pdf) or manually by using an entry in the setup menu (more detail on this later).

Directly below the clock display are the “mode” selector and the input voltage monitor. The voltage monitor may be turned on/off and calibrated using two different routines in one of the setup menus (discussed later).

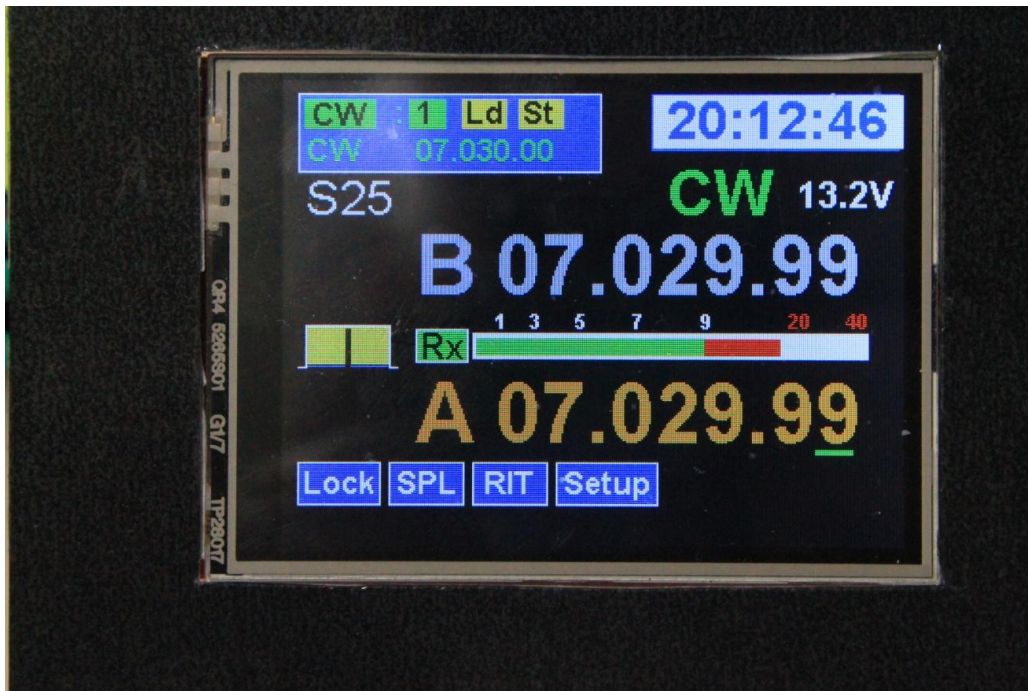
Setting the mode is pretty much identical to the operation of the memory “bank” selection shown previously.

Simply tap or press lightly within the confines of the Green mode indicator letters and this will bring up a drop down menu showing all the available modes and the desired one may be selected by tapping on the one you want. Once the mode is changed the drop down menu will close and your selected mode will be displayed in 2 or 3 large green letters. Simple and convenient. The available modes are USB, LSB, CW, CWr (reverse sideband), SWU (SW listener USB), SWL (SW listener LSB) and CWP (code practice). SWU, SWL and CWP all inhibit the transmitter for safety.



Selecting any mode will close the drop down and to leave it unchanged tap anywhere on the screen (outside of the drop down box).

The PBT (Pass band tuning) is a fairly new feature for the BITeensio software. It allows for tracking the IF and BFO frequencies across the passband of the 12 MHz crystal filter. Clicking on PBT turns the feature on and the box color will change to Yellow with a black line centered in it. This indicates that PBT is “on” and the black line is a cursor showing where in relation to filter “center” the IF is tuned. For adjusting the PBT, simply press and drag your stylus left or right along the “S-meter” display window. This won’t affect the S-meter but it will do double duty as a “slider” for the PBT. It tunes pretty slowly so it may take more than 1 “press & slide” to get it where you want it. The Black bar cursor in the PBT window changes to a Red bar when it’s moved off of center and that bar moves left or right to show the new PBT position. To turn PBT off, simply tap anywhere in the yellow window and it will reset the IF to normal and turn PBT off.



Finally, there are 4 “buttons” along the bottom of the screen labeled “Lock, SPL, RIT and Setup”. Lock when pressed turns RED and locks the frequency encoder so bumping it won’t accidentally change the frequency.

SPL turns Split on or off and it is a toggle. When SPL is on, the button turns Green with Black letters, the two VFO’s labeling changes from A and B to R and T with the top display the TX frequency and the bottom one the Receiver frequency. To make setting up your split easier, when SPL is turned on the TX VFO is made equal to the RX one. Next, select the T VFO, set it to the offset you want and reselect the R VFO. You can now fine tune the receive to the DX station without moving the transmitter. To move the transmitter, simply select the T VFO and adjust it, reselect R and you are set to go.



Tapping the SPL button again turns off the function.

RIT, (receiver incremental tuning) when tapped turns RIT on, the block changes to Green with Black lettering and the RIT +/- frequency display appears to the far right of the “setup” button. Using the tuning encoder you can adjust the receivers frequency +/- 5.0 KHz without affecting the transmit frequency which remains set to the “active” VFO frequency.



Tapping RIT again turns the function off and returns the receive frequency to that displayed on the active VFO display.

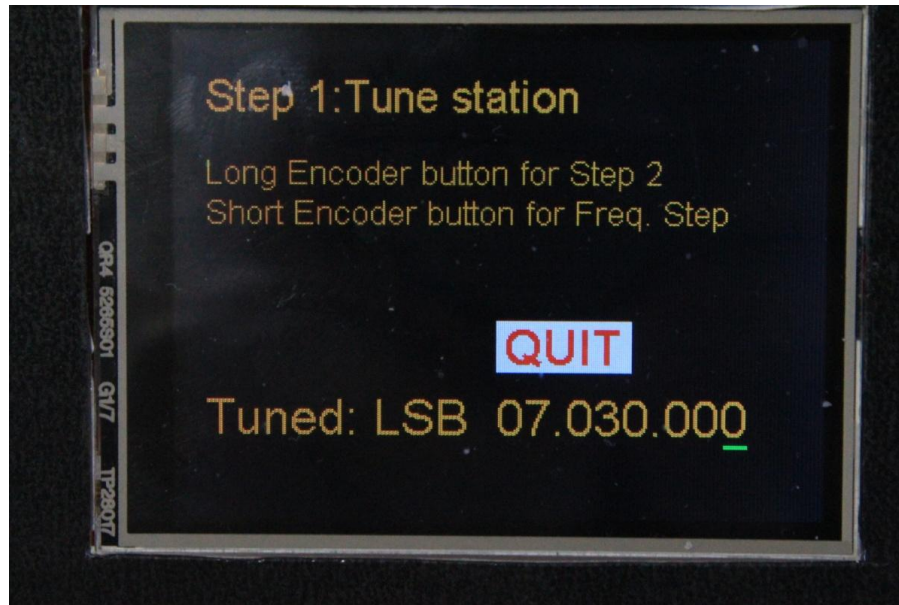
The Setup button actually brings up a pop-up window containing 4 selectable items. As shown in the next photograph they are, from bottom to top, Calibrate Radio, Configuration 1, Configuration 2 and CW options. These are separate pop-up menus that contain various setup functions for the software.



For instance, Calibrate Radio brings up a small pop-up that gives you 3 options. CALIBRATE (calibrate the master oscillator frequency), BFO Adjust (set the BFO to the right sideband and proper slope on the filter) and Exit which returns you to the main screen.



CALIBRATE takes you into the calibration “Step 1: Tune Station window where you will select the known accurate frequency (either a station such as WWV or other Time/Frequency standard, Broadcast station or calibrated signal generator). There is a “QUIT” button so you can back out of this without having to go through the calibration procedure if you accidentally enter the routine.



As it says on the screen, tune to the station – frequency display is at the bottom of the screen. Short presses on the encoder’s function button will move the green cursor under the desired digit for quicker tuning. Once you have the station tuned in properly, press and hold the function button until the Step 2: Best Sound window is displayed.



Using the frequency encoder, adjust the signal for zero beat or best sounding voice if listening to WWV or a broadcast station and when you

are satisfied, a short press on the encoder's function button will set the calibration and exit back to the previous main menu where you can then either exit or proceed to set the BFO frequency in which case you will enter the following screen(s).

Setting the BFO is very similar to setting the master oscillator except you will be adjusting the frequency directly with the encoder. You can set the tuning speed just like in the previous routine with a short press of the encoder function button. When you are satisfied with your setting a long press of the encoder function button sets the frequency and stores it. You can come back to either the master oscillator or BFO calibration menus at any time and work on them individually to refine your settings. This way you can tweak one without having to mess with the other and vice versa. As with the master oscillator calibration page, there is a big "QUIT" button to allow you to exit without having to change anything.



Next is the "Configuration 1" menu which contains some "Toggle" entries. Volts Display [ON] (or [OFF]) is self explanatory. Calibrate Voltmeter allows

you to set your displayed voltage to match what your multimeter tells you is the input voltage to your uBITX (especially useful if running on battery). S-meter Display [ON] (or [OFF], Set Clock(display [O N] or [OFF] and finally Set Time which allows you to manually set the time (less accurate than using the time setting in the Remote Control Program but useful when a computer isn't handy). Details on the Remote Control Program (RCP) are contained in another document and information on how to obtain it and the instruction manual for it are given earlier in this document.



The Configuration 2 menu which has entries to allow for reloading the default memory tables (re-sets the CW, SSB and DIG(ital) channels to the original QRP calling frequencies for each mode. The USR bank is not changed and will retain any frequencies & modes you have set for the 10 user defined channels. It also has a toggling OSC Load [XTAL xpf] entry where you can set the crystal's load PF to one of 3 – 6pF, 8pF or 10pF. The current BITeansio master crystal has a 10pF load and should be set to

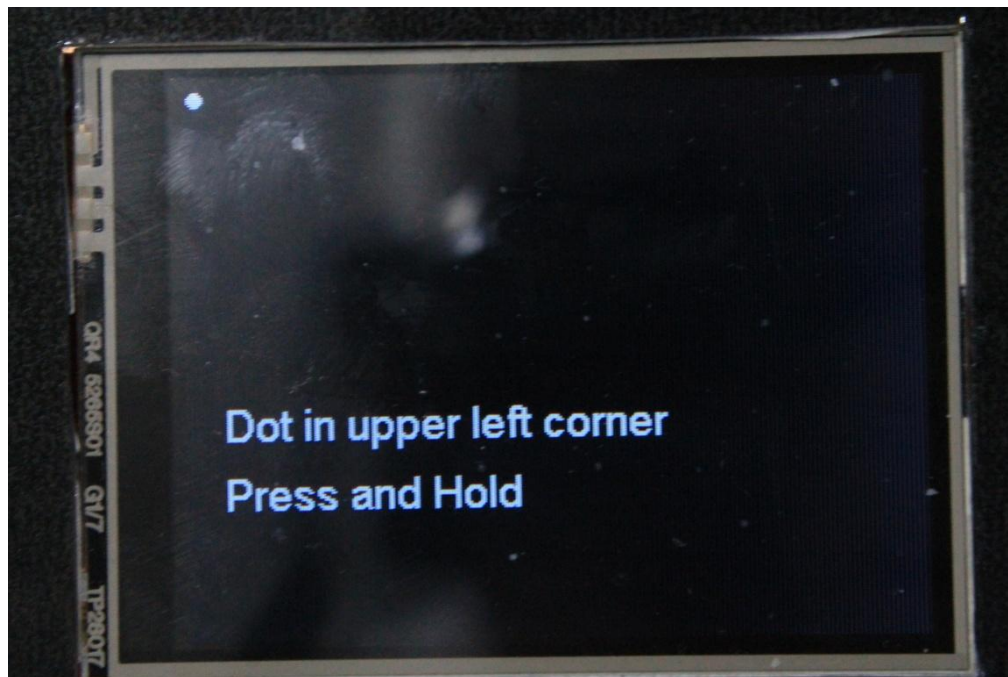
that value in this menu item. Next is the CAT Baud Rate (default is 38400 but it can be set to the standard baud rates from 1200 baud to 115,200 baud. It should be set to 38400 for use with the W2CTX remote control program. and the final entry is Screen Cal which allows you to calibrate the touch screen's X and Y coordinates so that the touch button areas are accurate.



The screen calibration routine brings up a screen that starts out with a 3 pixel dot in the upper left corner. You lightly press and hold the stylus point on that dot until it tells you "RELEASE NOW" at which time it will move to the lower right corner and the press and hold process needs to be repeated until RELEASE NOW is again displayed. This will then go to a screen that will display the actual minimum and maximum X and Y coordinates which have been averaged from ten samples taken while you were holding the stylus point on each of the two dots. These values are saved on the Teensy 3.6's SD card which acts as the EEPROM for the Teensy and is non-volatile storage. Once calibrated, it should not need to be done again

unless the resistive touch screen ages to the point where the touch points become badly inaccurate. If for some reason the coordinates become so badly corrupted that the “Setup” menu cannot be selected, you can also start the screen calibration routine by holding the encoder’s function button down when you turn the uBITX power on. Press and hold the button, turn the uBITX on and keep the button pressed until after the “Splash Screen” switches to the main screen and “Screen Cal” will be entered. This will allow re-setting the master coordinates for the program and re-save them on the SD card.

On initial use with a blank SD card, the Screen Cal routine will be entered immediately after the “Splash Screen” clears as the touch panel needs to be calibrated before initial use.





That pretty much covers the operation of the 2.8 inch color touch screen display that can be added to any uBITX radio that contains a “Triumvirate Skonk Works” BITEensio card.

The displays can be obtained inexpensively from Amazon.com or from various vendors on eBay.

The user will have to make their own interconnect cable and the wiring diagram for this is located in Appendix B of this document.

No BOM (bill of materials) is listed as it will be up to the user to determine what connectors, etc they need to make the cable.

The Red & Black power leads should be a twisted pair, the Green and Yellow wires (MOSI and MISO signals) should be a twisted pair and the other 4 wires (clocks, chip select, etc) should be twisted together as a 4 wire bundle. This will minimize noise pickup/radiation from the cable and so far the beta testers are not reporting any problems with this arrangement.

Appendix A: Keyboard.

The Teensy (3.6 only) MPU has the capability of implementing a USB “Host” mode and if you look at your Teensy 3.6 from the top side, just to the right of the USB “Micro B” input jack, there are a series of white outlined 5 holes. Carefully solder a 5 pin header into these holes with the long side of the pins sticking out the top side of the Teensy. You will have to order the 5 pin USB Host cable from PJRC.com (very inexpensive) and connect it to this header to be able to use the KB function we’ve added to this firmware.

The red wire (power to the KB) MUST be connected to the pin that has the white square around it on the Teensy.

The USB connector on the end not connected to the Teensy should be mounted to the uBITX front or side panel to allow a Keyboard to be plugged in.

Do NOT use a lighted keyboard as it will draw too much current from the BITEensio card and could cause damage to the Teensy AND the BITEensio card

The keyboard may be used without the uBITX being plugged into a computer and so far we have tested several 2.4GHz (not Bluetooth) wireless keyboards and they work well. The choice of wired or wireless is strictly up to the user and you may or may not get the same results we did so experiment to make sure your keyboard will work.

We have implemented a fairly comprehensive command list which covers most if not all of the available menu functions used in normal operation.

COMMAND LIST:

Currently there are two types of commands:

The following take effect immediately with NO Parameter entry –
(Command immediately toggles the item)

<CTRL>c Puts the system into Keyboard CW keyer mode - <ESC> exits. There is a 64 character type-ahead buffer and the input is displayed on the bottom line of the TFT color display. When the end of the line is reached, the characters will continue to scroll to the left and exit the screen to the left. During the time the KB CW Keyer is active, once you start typing, the uBITX goes into CW Transmit mode and stays in TX until 4 seconds after you stop typing or <ENTER> is pressed. If you don't exit KB CW mode with the <ENTER> key, typing another character puts it right back into TX again and will stay there for 4 seconds after the last character is typed or <ENTER> is used to exit the KB Keyer mode.

If you need to stop the keyer in the middle of a transmission, the <ESC> key will terminate sending and keyer mode immediately. To exit the keyer when you are done using it the normal exit mode is to press the (<ENTER>) key.

<CTRL>d Band Down cycles through memory table bands from 0 to 9 and back to 0 for the selected table (CW, SSB, DIG & USB)

<CTRL>m change mode (toggles through CW CWr, USB, LSB, SWU, SWL CWP and back to CW) Mode change takes place immediately.

<CTRL>i	toggle iambic (mode A or B)
<CTRL>l	toggle frequency encoder LOCK
<CTRL>p	toggle paddle (N or R)
<CTRL>u	Band UP cycles through memory table bands from 9 to 0 and back to 9 for the selected table (CW, SSB, DIG & USB)
<CTRL>v	toggle VFO (rotates through A-B)
<CTRL>r	toggle RIT (RIT-ON or RIT-OFF)
<CTRL>s	toggle SPLIT (SPLIT-ON or SPLIT-OFF (When SPLIT is ON, then <CTRL>v selects T or R VFO for frequency entry)

The following commands require the <CTRL> command and then a parameter entry followed by <enter> to change value or <esc> to cancel the command.

<CTRL>fxxxxxxx - set active vfo frequency

No leading zeros are needed but the trailing ones are. For example, to set the active VFO to 7.040.000 MHz would take: <CTRL>f (brings up blank 3rd line on display), type 7040000 and then press <enter>.

<CTRL>wxx set CW speed 5 =< xx <= 50

(Examples - <CTRL>w, blanks the line, type 05 <enter> sets 5 WPM, <CTRL>w then 25 <enter>, sets 25 WPM)

<CTRL>txxx set CW tone 100=< xxx <= 990

(<CTRL>t blanks the line, type 3 digits from 100 to 990 and then <enter> sets the entered value)

With Parameter entry an "Enter" completes the input.

With Parameter entry an "Esc" terminates command with no change.

With Parameter entry the value is tested against a range and a display tells user if outside range.

The error message will be something like "Bad Input" and the bad command is cleared so you can start over.

The keyboard can be used as a fully functional Keyboard CW keyer for transmit or code practice (CWP). However, we are not implementing a CW decoder and the reason we are not including a decoder is to encourage CW operators to learn to copy in their heads or at least on paper and to NOT send faster than they are capable of copying. CW decoders are extremely unreliable under noisy/QRM conditions and the possibility for decode errors causing errors in decoding call signs or other critical QSO data is too great. The best Morse Code decoder is still the human brain.

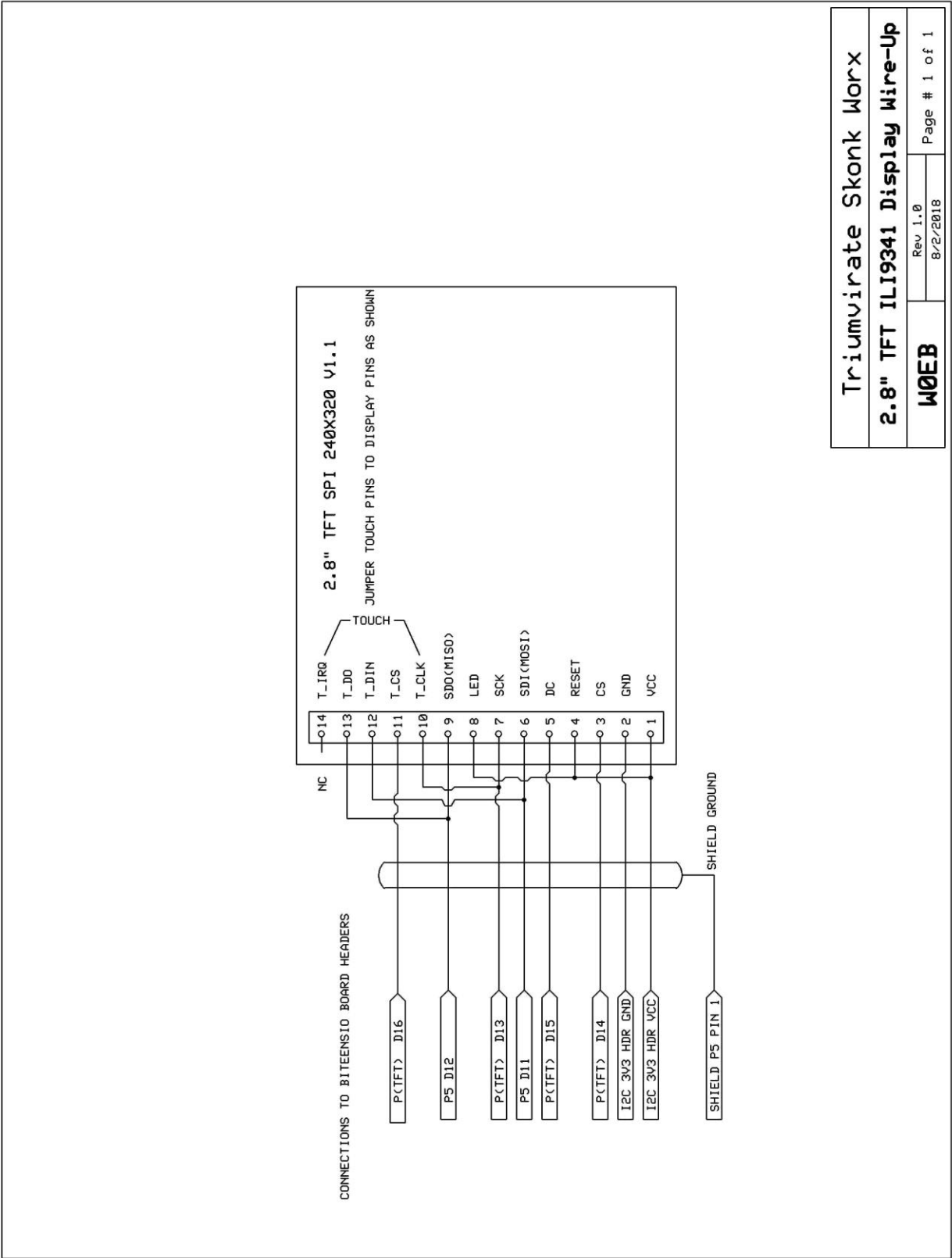
In addition to the normal letters and numbers , the full character map including punctuation has been implemented is as follows:

(There are several other key assignments that are treated just like letters and are buffered. These transmitted characters are sent as the two letters run together.)

= is mapped to BT
< is mapped to AR
> is mapped to SK
/ is mapped to DN
; is mapped to KR
@ is mapped to AC
! is mapped to CM
(is mapped to KN
\$ is mapped to VU
) is mapped to KK

- “ is mapped to AF
- is mapped to DU
- ‘ is mapped to WG
- _ is mapped to UK
- : is mapped to OS
- + is mapped to AR

Appendix B: Wiring diagram for the Display to BITEensio connecting cable.



Credits:

Authors and editors:

Ron Pfeiffer, W2CTX

Jim Sheldon, W0EB

Photography by Jim Sheldon, W0EB

Images copyright © Aug. 2018 by Jim Sheldon, all rights reserved.