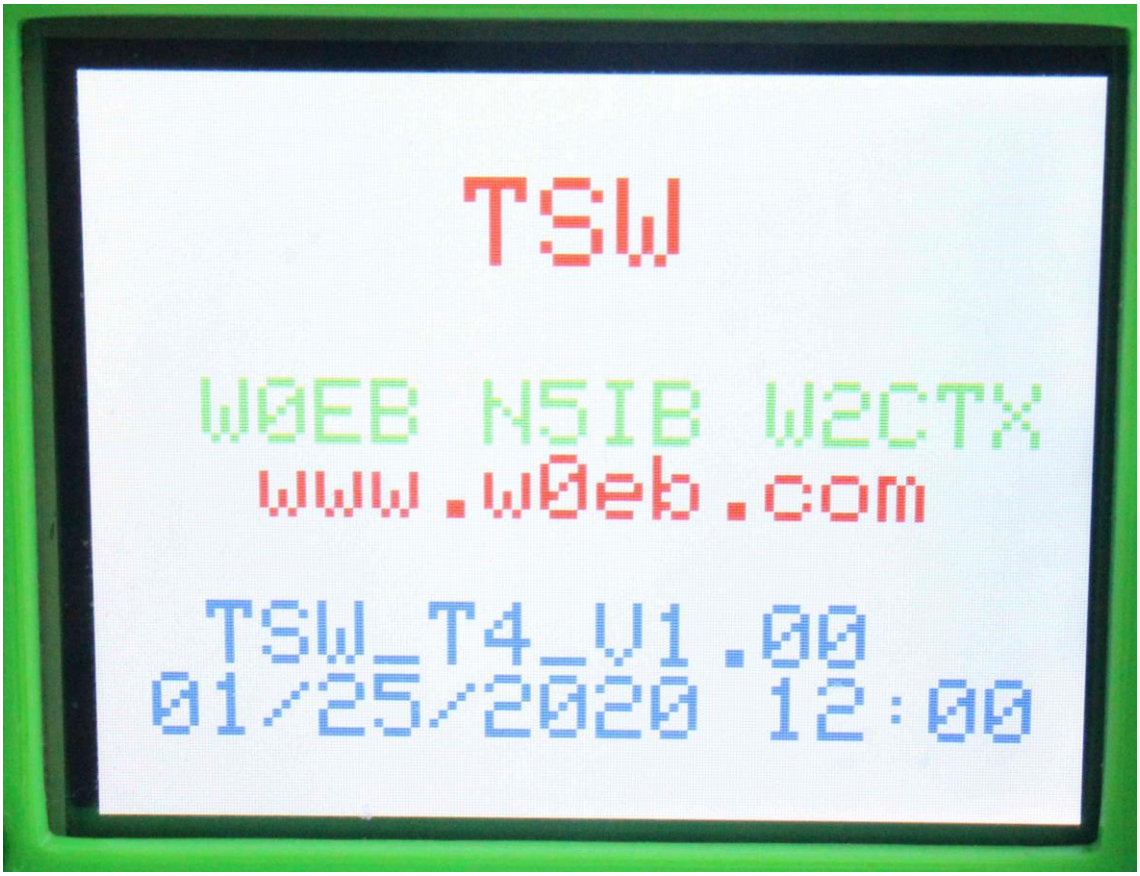
TSW Software V1.00 for the Teensy 4.0 MPU used with the μBITX transceiver from HF Signals (India)

From the Triumvirate Skonk Worx, www.w0eb.com

Manual V1.00 – last update 01/24/2020 @ 19:00 UTC



TSW_T4_V1.00 brief Splash Screen

Teensy 4 Adapter mounted on a V6 uBITX running TSW's V1.00 software.

Introduction:

There are many advantages to using a Teensy 4.0 in place of the original NANO. The T4 is MUCH faster – 600 MHz processor vs. the NANO's 16 Mhz, T4 has 1024k+ of RAM, much larger Flash memory, larger stack area for storing variables & program constants to name just a few. Its clock oscillators operate on different frequencies which get rid of a bunch of "birdies" that plague the NANO MPU Raduino's.

The major operational changes TSW made to the original software were to improve the CW keyer by using separate dot and dash wires, using the "Timer Interrupts" to allow other things to happen in the software while the key paddles are not pressed instead of having to constantly poll the inputs to see if a key press occurred. You ONLY have to add another Key Paddle jack which will connect to pins on the Teensy4.0 to NANO adapter to be able to use the Iambic Keyer. Hand key operation via the Microphone Jack's PTT connection remains the same and NO mods to the uBITX OR Raduino are needed for this.

The new paddle jack connects directly to the A8 and A9 input pads on the end of the adapter board by the Teensy's USB connector. A8 gets wired to the "TIP" connection of the new jack and becomes the Paddle DOT connection. A9 gets wired to the "Ring" connection and becomes the Paddle DASH connection. The jack's ground connection connects to the ground pin on that header.

We also greatly improved the calibration routines for both the Master Oscillator and BFO to make them much simpler and more intuitive to use, something that has been needed for the μBITX practically from day one.

NOTE: For CW, the hand key will still operate normally as long as you plug it into the microphone connector because, just like the Factory V6 uBITX (and Raduino), the PTT line is used for hand key operation, independent of the Iambic Keyer. The hand key MUST connect between the “RING” (PTT) and “SLEEVE” (Ground) connections of a 3.5mm stereo plug in order to be able to plug it into the Microphone Jack and use it for CW, but NO mods have to be made to the uBITX or Raduino itself. You can also connect an external keyer in the same manner if you like.

TSW’s keyer routine is “interrupt” driven and we’ve given the Hand Key priority in the interrupt hierarchy so if you key via Hand Key (or mic PTT button), until the uBITX returns to receive, the Paddles will be locked out.

How to wire the hand key properly is listed in the highlighted paragraph above. Please pay close attention to this and it will work great.

All Menu operations are now done by touching the buttons on the color touch screen either with your finger or a blunt plastic stylus.

Example showing rig in CW LSB (normal CW) mode. Keyer speed @ 25 wpm.

Using the above screen as a guide, a touch on CW will switch to SSB. A touch on LSB will switch the rig to Upper Sideband mode when in SSB and if CW was selected, (normal CW receive is on the lower sideband) it will select the other side of the filter and can help remove or reduce adjacent frequency interference from another station. When in CW mode, the Keyer speed will be displayed to the right of the CW/SSB button. Touch on the speed indicator will highlight it and allow you to set the speed with the encoder knob. Touch again and the new speed value will be saved.

Touching RIT will turn on the Receiver Incremental Tuning (RIT) and the transmit frequency will be locked. TX frequency will appear under the VFO A button. If RIT is selected, the “SPL” (split) function is locked out as only the receiver is being tuned in this mode anyway.

Example of RIT display

Touching SPL will put the rig into SPLIT frequency operation where VFO A will be receive and VFO B will be the transmit frequency. This will also lock

out the RIT function for as long as split is being used. To change your transmit frequency, click on the VFO B frequency. It will change to indicate “R” and be the highlighted window. Since the law in most countries requires you to listen on your transmit frequency, B will now be the receive frequency until you select VFO A again after setting your TX frequency using VFO B (you automatically listen on B while you are tuning it and the receiver switches back to A when you click on the VFO A frequency. The selected VFO is highlighted with a white box around it.

SPL example

The buttons labeled 80, 40, 30, 20, 17, 15 and 10 when touched will switch the rig to a frequency within that band and a yellow underscore line will indicate the selected band (initially the QRP CW calling frequencies are defaulted). You can set your own frequency on each band by first selecting that band and either tuning to it with the encoder or using the Frq button to select the desired frequency on that band. Waiting 5 seconds before

changing it again or changing to another band will save that frequency in the selected band button.

Example of the display after the Frq button and the digits 7,0,2,4 have been touched.

Using the Frq button is really simple. For example to set 7024.00 as the desired 40 meter frequency, touch Frq which will bring up the above menu. On this menu you would touch 7, 0, 2 and 4 in that order and then touch the on screen OK button. This will save that frequency to the 40 meter band button. Works the same way for all bands and if you are trying to do a quick QSY within a band, the Frq button will most likely be quicker than tuning with the encoder, especially for a long QSY to SSB from CW or vice versa. The other buttons on the Frq display are <- which is the digit erase button to wipe out a wrong number press one at a time. The Can button is “CANCEL” and will drop out of this menu back to the main screen without making any changes.

The Set menu

Set menu: In this menu you have 6 selections and an Exit selection. The first 4 selections operate on this screen for “S” meter on/off, Sidetone Frequency, CW delay (between TX/RX) and paddle swap for the keyer. The other two, CAL Freq and CAL BFO each take you to a special sub-menu for calibrating either the Master Oscillator (CAL Freq) or the BFO (CAL BFO) Exit will take you back to the main screen either immediately or after using one of the selections. There are more explicit instructions for calibrating the uBITX as used with the TSW software in Appendix B: of this manual.

The menu items may be selected either by touching them with your stylus or finger, or rotating the encoder knob to highlight the desired item and then pressing the encoder button to select it. This method can be a bit tricky due to the possibility of accidental encoder movement jumping to a different item than desired, so the best way is to just touch the item on the screen to select it.

S meter: The default is “Off” but you can turn it on by touching on the item. This has not been fully tested yet, but if you have an AGC (automatic gain control) board installed in your μBITX that has an “S-meter” output, as long as the maximum DC supplied by that output is LESS than 3.3 volts, you can connect it to the A7 input on the Raduino which has been listed as a “Spare” in the factory documentation. Assuming the AGC is operating correctly and the “S-meter” voltage varies between zero and about 3 volts, the S-meter should give you a relative reading. Right now it isn’t really calibrated and if the AGC output voltage is adjustable the user can use the adjustment to calibrate the S meter to S9 with a 50 microvolt input signal AT THE ANTENNA of the transceiver.

“S” meter display example.

Tone: This item works is for setting the sidetone frequency to your comfortable listening frequency. Default is 800 Hz. Touch the item and a black box will appear with the tone highlighted in white and the sidetone sounding in the speaker/headphones. You can set it using the encoder knob to any desired tone frequency from 100 Hz on the low end to well over 1000 Hz on the high end. Touch on the item again after setting the frequency to save the new value.

Set Menu included again to aid readability of instructions.

CW Delay: Sets the number of milliseconds after the last paddle or hand key press before the uBITX switches from Transmit back to Receive. Menu item works exactly the same as the Tone selection.

Paddle, “Normal” or “Reverse”: This gives you the option of reversing your dot/dash paddle levers (Reverse is for left handed persons or those who have their paddles wired backward from the conventional TIP=DOT, RING=DASH setup.) Click on the item and the on-screen value is swapped to its opposite (Normal or Reverse).

CAL Freq is used to calibrate the Master Oscillator frequency. See Appendix B to this manual for detailed instructions on using this menu item.

CAL BFO is for setting the BFO frequency to the right point on the crystal filter's passband. See Appendix B to this manual for detailed instructions on using this menu item.

When finished with any item in the Set menu and not intending to use another, touch the red Exit bar and you will be back at the main screen again.

During testing, it was noted that the annoying "Tuning Clicks" and many of the internal "Birdies" that had been associated with using a NANO for the main processor were either gone completely or reduced to a level where any atmospheric noise in the receiver covered them to the point they were not noticeable. This is one of the really nice advantages of using the Teensy 4.0 as the MPU instead of the factory supplied NANO.

Appendix A:

How to install the TSW µBITX Software onto the Teensy 4.0

By Jim Smith, KK0U, used with permission.

Overview

The Triumvirate Skonk Worx (TSW) has provided an enhanced operating sketch for the µBITX V6 radio project. This enhanced sketch takes advantage of the PJRC Teensy 4.0 board (available for purchase here: (https://www.pjrc.com/store/teensy40.html), which supports a 600 MHz processor, 1024k RAM and 2048 bytes of Flash memory – well beyond the capabilities of the Arduino NANO board the µBITX is based on. This guide will show you how to compile the software and upload that software to the Teensy 4.0. For instructions on how to purchase and build the TSW Teensy 4.0 to NANO plug-in adapter board, see the TSW website http://www.w0eb.com .

Installing the Arduino IDE

First the Arduino Integrated Development Environment (IDE) must be installed on your computer. The latest version (as of this writing) is Ver. 1.8.10. The software is available here: (https://www.arduino.cc/en/main/software) and versions for Linux, MacOSX and Windows are available. Follow the installation instructions on the Arduino website (https://www.arduino.cc/en/guide/HomePage) for your particular OS.

NOTE: Install the Arduino IDE. Before doing anything else, go into the Libraries directory for the basic IDE and DELETE the LiquidCrystal directory completely.

Installing Teensyduino

Next, go to the PJRC website and download the Teensyduino installer found here: (https://www.pjrc.com/teensy/td_download.html). Again, versions for Linux, MacOSX and Windows are available. Save the file to a directory on your hard drive and follow the directions on the PJRC website to run the installer – when it asks you for the directory to install it in, point the installer to the directory in which you installed your Arduino IDE and Teensyduino will be installed and integrated into the Arduino IDE's environment.

NOTE: Do NOT delete the LiquidCrystal directory under "Hardware – Teensy – Libraries" though as this is the one that is needed to compile the software.

Getting the needed AdaFruitILI9341 libraries

Now run the Arduino IDE by clicking on it. From the initial panel, click on the "sketch" dropdown menu, click on "Include Libraries" (alternatively, you can press CTRL-SHIFT-I on your keyboard which will bring up the "Manage Libraries" window). In the "Filter Your Search" section, type "ILI9341" and wait for the IDE to present library options. Hover over "AdaFruit ILI9341" with your mouse and click on "Install". The program will ask you some additional dependencies, click on "Install all". Once this completes, you are ready to download the latest TSW Sketch.

Downloading, compiling and installing the TSW Sketch

Go to the TSW website http://www.w0eb.com/TSW_uBITX_VERSION6_NEW_SOFTWARE/ directory and download the latest (by date/time) version of the TSW software for the Teensy 4.0. Usually there will be only 1 zip file in that directory but at times there may be earlier versions left for experimentation. You should normally download the latest version. Save this file to a separate directory on your hard drive and unpack (unzip) it. You will then have a directory containing all the program files for compile. Copy or move this unzipped directory into the same directory where your Arduino IDE files are.

Next plug your Teensy 4.0 board into a USB port on your computer, using an appropriate cable. Note that many cell phone charger cables are ***not*** wired for data transfer, only power. Be sure to use a USB cable that can transfer data.

With a brand new Teensy, the factory will have loaded a test program that causes the orange (or red) LED to blink once a second. This blinking will start as soon as the Teensy 4.0 is powered up by the +5 volts supplied through the USB programming cable. This is normal. If it doesn't blink, you may have a bad board or a bad cable. If the blinking LED's color is green or blue you most likely have a counterfeit board – you should have bought it direct from PJRC as the counterfeit ones may or may not work – "Caveat Emptor" (let the buyer beware).

Now, run the Arduino IDE. Once the main compiler window comes up, click on the "Tools" dropdown, hover over the "Board" line which will bring up another drop down giving a huge list of different boards that can be used with the IDE. You should see a faint "Teensyduino" at the top of this list and directly under that "Teensy 4.0". Click on Teensy 4.0. That will set the proper board for the IDE's compiler. Click on the "Tools" menu again set USB to "Serial" if it isn't already, Set CPU Speed to 600 MHz, Set "Optimize" to "Faster" Hover your mouse over "Port" and with your Teensy plugged into the USB cable select the serial COM Port that is indicated for the Teensy 4.

Now you have the IDE environment set properly for the Teensy 4.0 board, click on "File", "Open" and find your TSW sketch file in the list of file directories that are shown. Double click

on that directory which will bring up another list of actual files. Find the one that has the file extension ".ino" and double click on that one. The compiler window will now open with that .ino file loaded. Click on the "UPLOAD" Icon which is a right pointing arrow inside a green circle. This will compile the software and run the Teensyduino loader once it's finished compiling. If everything is okay, the compiled program will be installed in the Teensy. If the Teensyduino loader asks you, you may have to press the tiny button on the Teensy 4 at which time it will load the program onto the Teensy – this may or may not be necessary as the loader should be set for "Auto" upload as it comes from the PJRC download of Teensyduino.

Appendix B:

Calibration Procedure for TSW_T4_V1.00 and later firmware.

Starting with TSW_T4_V1.00, the Triumvirate Skonk Works programmer, W2CTX has included in the firmware's "SET" menu a reasonably simple, easy to use set of calibration routines for calibrating the uBITX Version 6 Raduino when used with either a TSW Teensy4 to NANO adapter on a factory V6 Raduino, a TSW V6 Raduino Clone or our forthcoming stand alone Teensy4 Raduino Clone (projected availability late February/early March 2020).

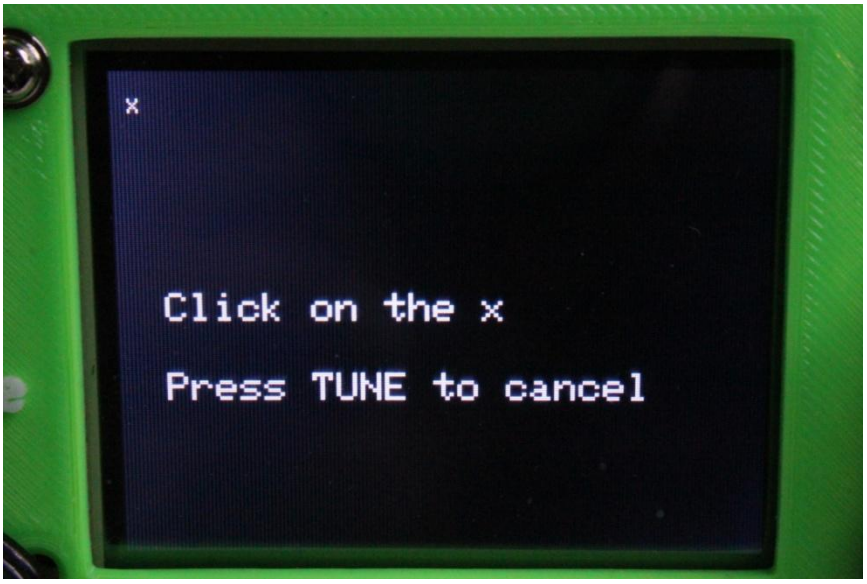
This manual is intended to provide the user with an easy to follow set of instructions to use these routines in achieving accurate "master oscillator" and "BFO" calibration for the HF Signals uBITX Version 6 transceiver that is using a TSW Teensy 4 to NANO adapter in an HF Signals V6 Raduino, in A TSW V6 Raduino Clone with the adapter or an a Raduino Clone that has been designed specifically to use only the Teensy 4.0 MPU and still plug into the uBITX Version 6 main board.

Also included are the instructions for calibrating the "Touch Screen" which should be done initially on a new radio or when the Teensy 4.0 is first used as the MPU to insure the entire on screen "Touch" buttons work properly. This calibration should be stable and should not need to be frequently updated.

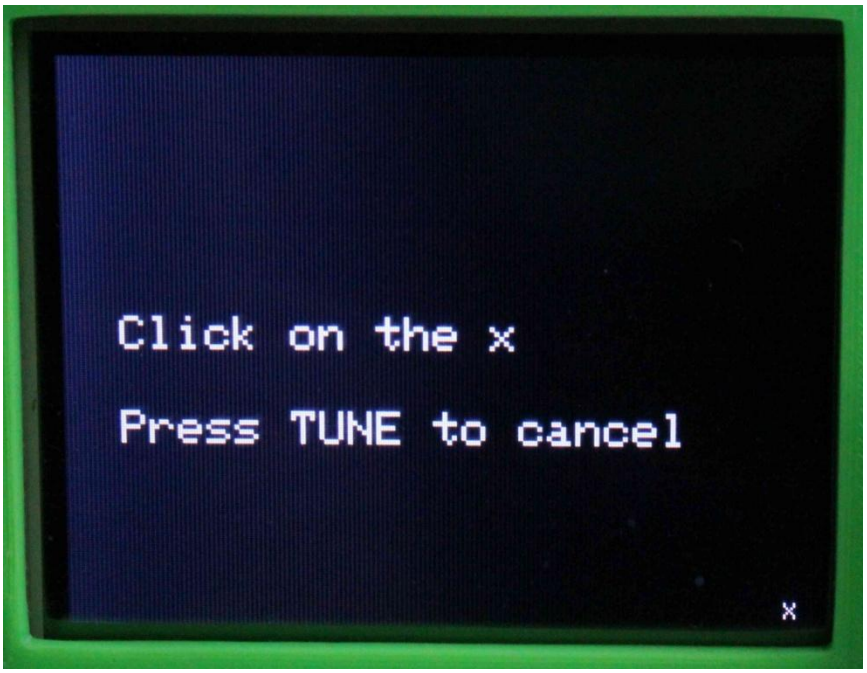
Calibrating the "Touch" screen:

The first item that needs calibrating on the uBITX when a new Raduino or MPU (be it a new NANO or Teensy 4.0) has been installed is the "Touch Screen" so the software knows the coordinates of all the "Buttons" on the screen.

This is accomplished by holding down the encoder button and then while still holding the button, power up the uBITX. After the initial startup of the firmware, keep holding the button until the Touch calibration screen appears. It will look similar to the following:



Example: Step 1, touch the “X” in the upper left



Step 4, touch the “X” in the lower right to save and exit

Follow the directions on the screen – touch your stylus (or finger) to the “x” in the upper left. When you release it, the x will move to the upper right. Touch it again and it will move to the lower left. Touch it again and it goes to the lower right. Once you touch the x in the lower right of the screen, it will calculate the coordinates for every pixel on the screen and save that value so the program will

be able to decipher which button has been pressed during subsequent operation of the uBITX and bring up the following “Main” operating screen.

To get to the Master Oscillator and BFO CAL routines, first touch the “Set” button the uBITX main operating screen shown in the photo below.

uBITX main screen shown on W0EB’s test set.

Touching the “Set” button will switch the display to a set of menu selection items as shown in the next picture.

There are 6 items on this menu screen plus an EXIT button which will bring you back to the uBITX main operating screen. For now, we are only going to explain how to first calibrate the Raduino's BFO and then the Master Oscillator. The other menu items are explained in the main body of the manual for the version being used.

The best sequence to use (After first calibrating the "Touch" screen) for calibrating your uBITX with the Teensy 4 for the first time is to set the BFO frequency first.

This needs to be done properly to insure that the USB/LSB selection is correct. The uBITX Version 5 and later use an 11.059 MHz IF frequency and the crystal filter is actually a USB (Upper Sideband) filter but due to the conversion scheme used with the first mixer the rig thinks it is a LSB (Lower Sideband) filter.

A bit confusing, but since it is actually designed as a USB filter, we need to set the BFO to the right point on the skirt of that filter to help suppress the carrier for SSB and to ensure that USB and LSB can be properly selected with the buttons on the main screen. Our default BFO frequency winds up being 11.055.0 MHz but this will only get you close. The following procedure will help you fine tune this to the point you should have no trouble selecting USB or LSB and having the uBITX receive on the proper sideband in SSB mode and CW operation.

BFO Calibration Procedure:

Calibrate BFO screen

To calibrate the BFO, touch the “CAL BFO” button on the SET menu shown on page 3 above and you will enter the screen shown above. If you entered this screen accidentally, touch the “USER EXIT” button and it will take you back to the SET menu without changing anything. If this is the first time you are using the screen the frequency in the window should be as shown.

The calibration procedure for the BFO is a bit more tricky than for the Master Oscillator, but isn’t really difficult. Best to do this one with the antenna connected and the frequency set to one with no signal, the volume control turned up about halfway, or at least so you can hear the background noise fairly well.

Tune the encoder knob (the frequency will display backward from the tuning knob’s direction of rotation) for the loudest background noise first. Write this frequency on a piece of paper. Keep tuning in the same direction until the noise just starts to diminish. Write this frequency on the paper. Now, tune in the opposite direction, through the peak of maximum noise (it may be pretty subtle

so listen carefully) and continue until it just starts to diminish again and also write this frequency down.

Look at the 3 frequencies you have written. One of the two diminished noise frequencies should be up around 11.057-11.059 and the other one should be down around 11.053-11.055 or so. (Due to individual crystal and other component differences, this will vary from uBITX radio to radio.) You want to set the BFO to the LOWER of those two frequencies. This places the BFO oscillator on the lower edge of the filter's steep skirt and all the upper sideband energy should be in the middle area of the filter. Do NOT set it to the higher of the frequencies or your USB/LSB buttons will be reversed and cause you a lot of confusion when trying to tune in an SSB signal.

Once you are satisfied you have the right frequency showing on the screen, press the "USER SAVE" button. This will save your now calibrated BFO frequency to the Teensy 4.0's EEPROM and drop back to the SET menu.

A word about the other "on screen" options; there are several touch buttons and they are "FACTORY RESET" this recalls the TSW default frequency of 11.055.0 and should only be used if absolutely necessary. "USER EXIT" gets you out of this screen without changing anything. "USER SAVE" is used to save your selected frequency and exit the screen. Finally, if you accidentally change something but didn't save it, you can use the "USER RESTORE" button to get back to your last saved BFO frequency.

This completes your BFO calibration.

Master Oscillator Calibration:

With the uBITX V6 transceiver powered up, tune the radio to the 10000.00 MHz frequency of a standard time/frequency station OR if one cannot be heard where the user is located, an accurate signal generator on ANY known frequency. 10000.00 MHz will be used in the example given here but any accurately known frequency may be used. Just tune the uBITX to that frequency prior to entering the calibration routine. Set the mode to SSB.

If this is the first time calibrating your uBITX, you should now touch the item labeled CAL Freq on SET menu and it will bring you to the following screen:

Master Oscillator calibration screen.

Referring to this “Calibrate Frequency” screen, you have several options.

If you have a 10000.00 MHz signal tuned in and can hear a tone, turn the encoder knob until the tone decreases in frequency and finally goes to zero. Keep tuning and if the tone appears again, tune in the opposite direction until the tone again goes to zero. If you are using a time/frequency standard station you should now hear the time ticks or beeps clearly. If you are using a signal generator or some

other 10000.00 frequency standard oscillator you should hear only background hiss and no tone, not even a very low frequency one for best accuracy.

Once you have achieved this, you can either press the encoder button (not recommended because it is very easy to accidentally rotate it slightly, throwing the calibration off by a few Hz or more), or better, being careful not to move the encoder knob, touch the blue “USER SAVE “ button on the screen. This will save the calibration value in the Teensy’s EEPROM and return to the SET Menu screen.

You can now touch the “Exit” button and return to the main operating screen with your master oscillator calibrated for normal operation.

After the first time calibrating your uBITX, if you need to touch up or recalibrate either the Master Oscillator or the BFO, it doesn’t matter which sequence you do them in as they will both be correct.

Credits:

Author/photographer, Jim Sheldon, W0EB Email: w0eb@cox.net

TSW Master Programmer, Ron Pfeiffer, W2CTX

Appendix A author, Jim Smith, KK0U

(KK0U generously submitted the material for us to review and after making a few minor revisions, we asked and received his permission to use the information so we have included it as an appendix to this manual).

TSW Website: http://www.w0eb.com

Manual version 1.3 dated January 25, 2020

Credits:

Jim Sheldon,

TSW Project Coordinator and Webmaster

www.w0eb.com

Ron Pfeiffer, W2CTX

TSW Master Programmer

Jim Giammanco, N5IB

TSW Engineer/PC Layout Specialist

Jim Smith, KK0U author of Appendix A to this manual.

Last update 01/24/2020 @ 19:00 UTC