

W0EB/W2CTX Firmware for the Micro BITX
80-10 meter Transceiver
for NON I2C Release V2.02R



Software running on W0EBns Test Platform.

Instructions for uBITX Version 2.02R software written by Ron Pfeiffer, W2CTX and Jim Sheldon, W0EB, including the minor hardware changes needed to make it work properly.

Every effort was made by the authors to ensure the software is trouble free and safe to use. However, NO WARRANTY is provided or implied and it is up to the person(s) using this software and this manual to read and understand the following before proceeding. Email support is available from the authors. See credits section at the end of this document.

NEW IN THIS VERSION, THE CW KEYSER IS NOW COMPLETELY INTERRUPT DRIVEN FOR EVEN BETTER CW PERFORMANCE! ALSO SEE APPENDIX A, AFTER THE CREDITS PAGE FOR MORE INFO!

INTRO:

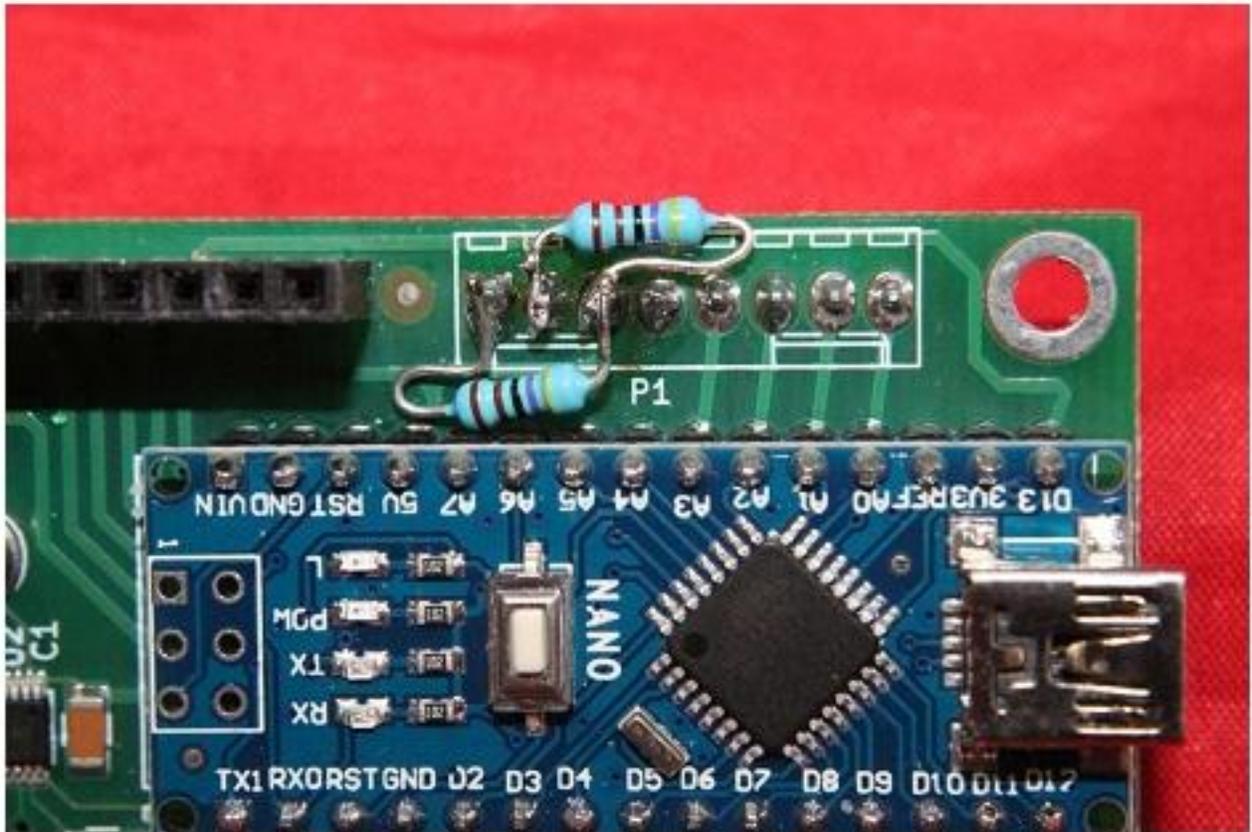
uBITX - Necessary information for programming and utilizing this software:

Three things need to happen for the user to be able to load and operate this software with the Micro BITX transceiver or as it has been called, the uBITX.

First, we assume the user knows how to connect to the Arduino "Nano" on the "Raduino" card that controls the transceiver and has installed the Arduino "IDE" to allow for program compiling and uploading software to the card. There is an excellent tutorial available via a link on the HF Signals website. You must become familiar with this part before any reprogramming of the uBITX can take place.

Second, you will need to unzip the ubitxV2_02R.zip file into your Arduino directory. This file contains ALL the necessary files to allow the program to be compiled and uploaded to the uBITX by the Arduino IDE program.

When you wired up your uBITX in the first place, you were told you MUST add a 4.7K resistor between the A6 (Blue wire) input and +5 volts (Green wire). With this version of our software it is STILL necessary. It IS okay to install this resistor directly on the Raduino card's terminal pads (reverse side of the "Digital" connector), just be careful not to short the +5 volt pin to anything. (Size 0805 or 1206 Surface mount 4.7 K resistors fit pretty well between the two pins but it's perfectly okay to wire an axial lead resistor to the proper pins as well.)



The above picture shows two resistors soldered in place on the Raduino Board. Only one is needed, the one near the top of the card. (The one on the bottom was for a different version of the software that used the A7 line for paddle dash.)

The key jack also needs to be rewired slightly - the Blue wire needs to be connected to Pin1 (tip) of the key jack and a wire from Pin2 (ring) of the key jack needs to be connected to the PTT terminal on the microphone jack. Depending on a selection in the new menu (CW:Key, CW:Pdl or SSB), the A3 analog input to the Nano (Orange wire on the DIGITAL plug) which is used for microphone PTT (Push to Talk) now does triple duty. It is PTT (in SSB mode) CW Hand Key or external keyer (In CW:Key mode) and Paddle DASH (in CW:Pdl mode). The tip connection of the existing "Key" jack becomes the dot paddle input and the ring connection is now the dash paddle to the keyer as well as hand key/external keyer input.

It might be a good idea to install another key jack for the Hand Key/External keyer which should have its TIP pin connected to the RING pin of the existing microphone jack or to the PTT connection on the microphone jack if you are using one different than that supplied with the kit.

Alternatively, you could make up a short adapter cable to allow the hand key to plug into the microphone jack. This cable should have a standard 1/8" stereo plug on the radio end and a standard 1/8" stereo JACK on the key end. If you have used something other than the microphone jack supplied with the kit, this alternate cable would need to use a matching plug with the key connection hooked to whichever pin you used for PTT in order to work properly.

To make this cable work, connect the two shield connections together, leave the RING connection open on the JACK end and connect the tip of the JACK to the RING (or PTT connection) on the PLUG to the radio.

A very big advantage of using the A3 connection for PTT, Hand Key and Paddle Dash is you may now use the microphone PTT button as an EMERGENCY hand key if caught out in the field without a key and need to use CW to get a message across under conditions where CW can be heard but signals are too weak for a legible SSB contact.

Special contact conditioning is no longer needed for either the paddle contacts or a hand key/bug's contacts and the A7 input is now back to being a spare (or for those wishing to implement metering and requiring an analog input).

Another minor hardware change the user might want to make is to provide a second function pushbutton separate but in parallel with the button on the

encoder. When selecting menu items with the button switch on the encoder it is easy to accidentally turn the encoder slightly when pressing this. That can have the effect of selecting something you didn't really want and cause you to have to go back and do some operations more than once. With a separate switch, pushing the button will be done with your hand away from the knob on the encoder and make life a bit easier. This is not absolutely necessary and tins up to the user to decide whether or not he or she wants to add this modification.

That covers the hardware changes necessary for making the keyer work properly and the convenience modification the user may want to make.

uBITX Operational Modes:

We have created 3 operational modes in an attempt to optimize the basic HF SSB/CW operations. The modes are selected during power-up via the encoder (or separate) push button.

The three modes are: Normal, EEPROM UPDATE and ALIGNMENT. (ALIGNMENT is the original "Factory Alignment" programming).

All mode selections occur ONLY at power-up. They are mutually exclusive and when you are finished using the EEPROM UPDATE and ALIGNMENT modes you must cycle the power (Turn the uBITX off and then back on again).

SELECTING MODES:

Normal Mode: General radio operations. Power up your uBITX and do nothing. The normal radio features will be defaulted. This is the mode you will use to operate the uBITX on the air.

EEPROM UPDATE Mode: This mode will accept EEPROM reads/writes via Ian Lee's (KD8CEC) Windows EEPROM Manager program. (In the future there will be a Linux EEPROM manager).

EEPROM memory locations for our sketch may be different than Ian Lee's.

In order to use Ian's EEPROM Manager:

1. Power up your uBITX.
2. When the Version banner is displayed, push the encoder button.
3. Keep the encoder button pushed until "EEPROM" is displayed.
4. When "EEPROM" is displayed release the encoder button. You may now use Ian's program (follow his directions) to read/write the EEPROM memory locations. (Not for the faint hearted, but it will allow you to change values in the various locations.)

To exit this function close down Ian's program and depress the encoder button until "EXIT" appears on the top line of the uBITX display. Tap the button to Exit. Power cycle the uBITX and it will come up in "Normal" mode with the changes you made applied.

ALIGNMENT Mode: Will present the menu for performing a factory alignment (Ashhar Farhan's original factory alignment code has not been changed other than to add a "Tune" function to allow tuning in WWV or some other standard frequency, so follow the original uBITX instructions for setting Calibration and BFO.)

1. Power up your uBITX.
2. When the Version banner is displayed push the encoder button.
3. Keep the encoder button pressed until "Alignment" is displayed.

4. Once "ALIGNMENT" is displayed, release the encoder button and "Exit" will appear. Turn the encoder knob to the right to select the desired alignment function, either Tune, Calibration or Set BFO. After these functions are complete, turn the encoder knob to the left until "Exit" appears and press the button to exit. Even though the uBITX will appear to come up in "Normal" mode on exit, you will still need to power cycle the radio to make sure the alignment changes are applied.

Turning the knob to the right after "Exit" appears will get you to "Tune": short press the encoder button to go into tune. Use encoder to select the frequency you wish to use for calibration. Use the same short press of the encoder button to change the cursor under the digits for frequency increments. A Longer press will exit the "Tune" portion of the alignment menu, leaving the uBITX tuned to that frequency and you may then select the calibration or BFO adjustment procedures as normal. After adjusting the Calibration or BFO you press PTT (NOT the encoder/function button) to save the item. Pressing PTT exits the Calibration or Set BFO procedure back to the Alignment Menu. Turn the encoder knob to the left until "Exit" appears and press the encoder button.

"Power Cycle
Radio"

Will appear. Turn the uBITX off and then back on to make sure the changes you just made are saved to EEPROM.

NORMAL MODE FEATURES:

Many people expressed the desire for a different method of changing bands and tuning speed as the original method was sometimes difficult to control.

We removed the accelerated tuning feature as the original speed-up timing caused the accelerated speed change to happen far too quickly for many people and it was hard to set an exact frequency if you happened to slip. The frequency would sometimes jump by 2-300 KHz and then you had to crank forever to get back to where you were. Instead, we created a function where a short press (tap) of the encoder button would change the position of an underscore cursor and allow you to select the digit of the frequency you wished to change. This allows for more precise quick setting of the frequency.



The cursor under the digit can be moved by tapping the encoder button.

MENUS:

Now for the menu I mentioned near the beginning of this manual.

To enter the Menu, power up your uBITX and wait for it to boot up into the default which is "Normal" mode.

Press the encoder button until "Menu" is displayed. Immediately release the encoder button and a display full of changeable items will display.



Initial Menu Display

The picture shows the display just after entering Menu mode. The cursor is under the C in CW. A "tap" of the encoder button will change the mode from CW to SSB and exit the menu.



The CW:Key (or Pdl), Sidetone Frequency and Keyer Speed items won't show in SSB mode as, of course they are not used for SSB.

Entering the menu again, the cursor will appear under the S in SSB. A tap of the encoder button will change the mode back to CW and again exit the menu.

Anytime you enter the menu, the underscore cursor will appear under the item that will be changed by a tap of the encoder button. The cursor can be moved either right or left by turning the encoder knob. Once you have the cursor under the item you wish to change, a tap of the button will carry out the change.

When changing the sidetone frequency, once the item is selected, the current tone frequency will be displayed, the current tone will be sounded in the speaker/earphones and turning the encoder to the right or left will increase the tone frequency in 10 Hz increments. Once you have the tone you like, tap the encoder button. The tone value will be saved and the menu exited.

Keyer speed works the same way - put the cursor under the current speed value and select it with a short press of the button. Turning the encoder knob right or left will increase or decrease the keyer speed in 1 WPM increments. Tap the button again to save and exit.

RIT is enabled by entering the menu and rotating the encoder knob until the cursor is under the R on the bottom line of the display. A tap of the button turns on the RIT and you get the message "rit Enable" on the top line of the screen for about a second and then you see the following display:



Even though no cursor is shown, turning the encoder knob changes the RIT increment on the top line of the display. Turning the knob to the right will increase the receiver frequency in 100Hz increments above that shown on the bottom line of the display and the + sign will appear in-between the R and the first KHz digit. Turning the knob to the left will decrease the frequency and if the frequency is below that displayed on the bottom line, a minus sign will appear between the R and first digit. When RIT is on, the transmit frequency remains fixed on the frequency shown on the bottom line of the display. RIT is limited to plus or minus 5 KHz from the main displayed frequency.

To turn RIT off, re-enter the menu, make sure the cursor is under the R and tap the encoder button. "rit Disable" will appear briefly and normal transceiver operation will resume.

USB or LSB may be selected by putting the cursor under the U (or L) of USB or LSB (whichever is being used at the time) and tapping the encoder button. The mode will toggle and exit the menu.

CW is normally received using upper sideband by current convention but this can be changed by using the USB/LSB change function in the menu as well.

The two VFO's A and B can be toggled by placing the cursor under the letter A (or B if that's the one selected) next to the first digit of the frequency on the bottom line of the display and performing a short press of the encoder button. The VFO assignment is toggled and the menu exited. To reverse the change, enter the menu, and set the cursor under the VFO's letter and tap the button again.

Changing bands can be accomplished in a couple of ways. The easiest doesn't even require entering the menu. Simply place the cursor under the MHz digit on the display and turning the encoder right or left until the desired MHz frequency of the desired band is reached. Tap the encoder button to move the cursor to the appropriate digits and set the desired frequency within the band.

The second way is to enter the menu, move the cursor under the MHz digit of the frequency and short press the button. This will bring up the band select menu and you can use the encoder to select from the 80, 40, 30, 20, 15, 17, 12 or 10 meter bands. Tapping the encoder button will enter that

band and set the frequency to the QRP calling frequency (dependent on CW or SSB modes) for that band/mode and exit the menu. This menu item does not have an exit function other than selecting a band, so if you get here by mistake you will have to select one of the 8 available bands in order to exit.

SPLIT mode. This is a little different to use than entering the menu and selecting it. To start with, tune to the band and frequency of interest. Now, press and hold the encoder button a little longer until "SPLIT ON" appears.



Immediately release the button and you will see the following for a second.



Roughly a second later the display will change to:



VFO A is used for the receive VFO and VFO B is used for transmit. Initially when SPLIT is turned on, they are set to the same frequency. (My finger slipped on the encoder and the R frequency is at 7.025.10 instead of .00.)

Right now as the display shows, we are in CW mode with the transmit frequency and receive frequency showing. The uppercase R in front of the receive frequency indicates that any encoder knob movement will change the receive frequency only and currently it's set to 7.025.10 MHz. Adjust the Rx frequency for best reception on the station you are trying to work.

Now, enter the menu. With SPLIT on, things work a bit different. The cursor is placed automatically under the R on the bottom line. A quick tap changes it to r and moves the cursor to the right most digit of the transmit frequency on the upper line of the display. The t is now changed to a T and this indicates the encoder will now change only the transmit frequency. Set this to 1 or more KHz up (or down if indicated by the DX station) and enter the menu again. The cursor will be under the "r" and if you do another tap of the encoder button, it will change to R so you can tune the receiver if necessary. The transmit frequency will be displayed on the top line and since it isn't being tuned right now the letter will be a lowercase t. You now have SPLIT mode set up.

To turn SPLIT off, press and hold the encoder button until "SPLIT OFF" appears. Immediately release the button and "split Disable" will appear on the top line of the display followed a second later by a return to normal transceiver function.

The last feature of the encoder button is something we recently added as a convenience feature so you could save your operating frequency before shutdown if you wanted to come back to it later instead of having the uBITX go to the default frequency on power up. It's called "Save State" and it's activated by a very long press of the encoder button (approximately 5 seconds) and "Save State" will appear on the display. Releasing the button saves the frequency as the "default" in EEPROM so the next time the uBITX is turned on it will come up on that frequency. All other items such as mode, VFO, tone, speed and USB/LSB are already saved in EEPROM when they are changed we just save the frequency in the Save State function.

Questions, comments (positive/negative, we're pretty thick skinned) or complaints should be directed to the authors at our respective email addresses listed below:

CREDITS:

The authors of this program and manual are Ron Pfeiffer, W2CTX and Jim Sheldon, W0EB. Hardware modifications/Photos by Jim Sheldon, W0EB.

We have tried to ensure that the program functions properly in a standard Micro BITX transceiver after the required hardware modifications (moving the keyer paddle dash line to the same wire (orange) as the PTT and also connecting (suggested) a separate Hand Key/external keyer jack to the same wire. This works because we don't use PTT in CW and we don't use the keyer in SSB. Also, the Key/Paddles are selected in the menu so this wire can do triple duty.

DISCLAIMER:

The authors are releasing this software into the Open Source community for the Micro BITX transceivers. If you chose to use it, you are responsible for its use in your own Micro BITX transceiver. We, the authors are offering NO Warranty as to the program's suitability and you must act accordingly if you use it. You are responsible for determining the laws of your country governing Amateur Radio operating frequencies and modes and operating in strict accordance with those rules. The authors assume absolutely NO responsibility for the users' actions and may not be held responsible for the actions of any/all third parties using this software.

The authors' contact email addresses are provided below for bug reporting and feature requests.

Jim Sheldon, W0EB - email address w0eb@cox.net

Ron Pfeiffer, W2CTX - email address w2ctx@yahoo.com

APPENDIX A:

PC Control "Command" Reference for the uBITX Transceiver (CAT)

(Added to uBITX software by W2CTX/W0EB in Version 2.01R of our software on Feb 15, 2018)

Since at least one other programmer has implemented CAT or PC Control software for the Micro BITX transceiver (uBITX) and is using the Yaesu FT-817™ command protocol, in order to keep from duplicating his efforts and possibly causing command conflicts, we have chosen to use a small subset of the Kenwood™ TS-590S™ transceiver's PC control commands in our version along with one or more unique uBITX commands (more may come later).

Listed here are the commands we use and how they affect the operation of the uBITX transceiver.

CAT has been implemented in both the I2C and NON-I2C versions of our software and this is the current version of the "Command Reference" document.

Only the Kenwood™ commands pertinent to the uBITX are shown in the manual along with any non-Kenwood items. In this version there is one command totally unique to the uBITX. When and if they are required, more unique uBITX commands may be implemented and included in future updates to this document.

Jim - W0EB

External control parameters: First off you need your Raduino Nano's USB port connected to a valid USB port on your computer or other device you use to control the uBITX. You will have to know which "COM" port your computer (or other USB terminal device) uses to communicate with the NANO. This will most likely be the COM Port used by the Arduino's IDE to upload programs to the NANO in the first place.

COM Port parameters for this version of CAT are 38400 N-8-1 (baud rate 38400, No Parity, 8 data bits and 1 stop bit). You can also use 2 stop bits if you like, either will work.

Current list of available commands: (usage and required parameters will be explained individually later in this manual:

FA;	Reads and sets the frequency of VFO A
FB;	Reads and sets the frequency of VFO B
FT;	(Turns on SPLIT frequency mode)
KS;	Reads and sets the internal CW Keyer speed
MD;	Reads and sets the operating mode (CW SSB/USB SSB/LSB)
RX;	Returns the uBITX to RECEIVE (after a TX; function)
TX;	Puts the uBITX in Transmit (SSB/USB SSB/LSB only)
uST;	Reads and sets the CW side tone freq. (unique to the uBITX)
VV;	Sets VFO's A and B to the uBITX current display frequency

At the time this was written, these were the only commands that were used in the software. As they become practical, more may be added as well as more that will be unique to the uBITX.

Detailed description of the commands & parameters:

All commands MUST end with a semi-colon which tells the software to EXECUTE the command.

FA; Returns the frequency of VFO A.

For example, the command FA00007025000; Sets VFO A to 7.025.00 MHz the frequency must be exactly 11 digits long between the FA and the semi-colon at the end of the command. Even though the frequency is set and displayed to only 2 digits to the right of the last decimal point, the third digit must be a zero. Also all empty digits must be zeros.

Example of a frequency 10 MHz or higher - FA00014200000; sets the uBITX to 14.200.00 MHz (VFO A frequency)

FB00007025000; sets VFO B to 7.025.00 The same parameters are used as for VFO A and the commands are identical except it's FB instead of FA.

FT; Two parameters are used here. FT0; turns SPLIT OFF

FT1; First sets VFO B to the same frequency as VFO A then turns SPLIT mode on. (FT0;) turns it off again.

KS; Reads and sets the internal CW Keyer speed (05-50 wpm)
for example, KS05; would set the keyer speed to 5 wpm. KS25; sets it to 25 wpm and so on.

MD; Reads and sets the transmit / receive mode.

MD1; Sets SSB mode and Lower Side Band (LSB)

MD2; Sets SSB mode and Upper Side Band (USB)

MD3; Sets CW mode and USB

MD; by itself returns the current uBITX mode setting.

RX; No parameters are used with this command and it returns the

uBITX to the receive from a TX0; command.

TX; Only the 0 (SSB mode transmit) parameter is used.

For example, TX0; puts the uBITX in transmit in SSB mode just like using the PTT button on the microphone. This and the RX; command is mainly for use with external programs such as FLDIGI and the like that need to be able to put the uBITX into transmit mode and return it to the Receive state under their control.

uST: Reads and sets the CW side tone frequency. This command is unique to the uBITX so we preface it with a lowercase u and it MUST begin with a lowercase u. Uppercase U will not work.

This command is not found in the Kenwood command list.

VV; No parameters are used - simply sets VFO A and B to the current uBITX display frequency. (Those are two uppercase V's, not an uppercase "W". I clarify this as a quick glance might seem that it was a W rather than two V's side by side.)