Review

uBITX V6 QRP transceiver

Those in the amateur radio community who enjoy low-cost transceiver kits may have heard of the uBITX series transceivers created by maker-guru Ashhar Farhan, VU2ESE. His kits are well known for being affordable, functional, and completely open-source. Late in 2019, he announced his latest QRP transceiver design: the uBITX V6 general coverage transceiver.

The uBITX transceivers are quite well suited to experienced radio users who love to experiment and modify their radio equipment. But they are equally well suited to those just getting started in amateur radio who want an affordable, hands-on way to learn about electronics and to experience the HF bands.

The uBITX Version 6 (V6) comes in two configurations:

- a Full Kit that includes the transceiver, hand mic, power cable, and black powder-coated metal chassis for $200;
- A Basic Kit that includes everything, save the chassis, for $150

It's worth noting that the uBITX V6 is not actually a 'kit' in the traditional sense of the term. All of the boards are fully-populated and assembled by a women's cooperative in India (thus providing a fair living wage to women and their families). So the main work on this 'kit' is complete. But perhaps this is what makes the uBITX such an excellent project for a beginner: while there are no surface-mounted components on the circuit boards to solder nor fussy toroids to wind, the simple process of connecting the boards together yourself provides you with the opportunity to view the receiver/transmitter sections, the display, and the processor, thus building an understanding of the uBITX V6's inner workings.

Another fact that may surprise some readers: the uBITX is *not* a software defined radio (SDR), rather, it is an *analogue receiver.* The receiver has a roofing filter at 45MHz and an 11.059MHz QER SSB filter with 8 crystals. The uBITX shouldn't overload too easily, but if you live near a blowtorch medium wave broadcaster, the manufacturer recommends adding a high-pass filter with a cut-off around 1.6MHz.

According to the manufacturer, the uBITX transmitter produces 10 watts, but my measurements indicate that output is around five to seven watts on most bands. The amplification section relies on inexpensive IRF510s, so if you were to accidentally blow a final at some point, replacements are cheap and rather simple to replace.

The 'brains' of the uBITX V6 is an Arduino Nano board. Anyone with Arduino experience will find uploading, altering and replacing the uBITX's free and open-source software a breeze. Those who've never really dived into the Arduino world will find, as I did, that the learning curve is pretty modest, and certainly worth the investment of time.

The uBITX transceivers are well suited to experienced radio users who love to experiment.

After learning about the uBITX V6, I ordered one immediately; I was promptly shipped a radio from the very first production run.

Assembly

The entire uBITX V6 kit was packed in a box and shipped with the contents and chassis very well protected.

While all of the components and assembly screws were packed in individual bags, there was little in the way of documentation or instructions, only a contents/parts list. I consulted HFSignals.com, the home of uBITX, and found easy, straightforward assembly instructions (www.hfsignals.com/index.php/ubitx-v6-assembling-the-full-kit/)

My young daughters are studying for their amateur radio licenses and were curious what was involved with assembly, so I handed the process over to them. They had no problems whatsoever in assembling the entire transceiver themselves – I only helped by plugging in one board and, to be frank, they could have easily done this as well.

Again, the process was not difficult, but we did scratch our heads a few times trying to determine which length of screw to use for each assembly stage (hint: save the shortest screws for attaching the TFT display to the front panel). I heard from other uBITX V6 early adopters that some screws were missing in their kits. Our kit, on the other hand, had almost *two* of everything. The only missing part was a nut that fits around the BNC connector; I quickly unearthed one in a miscellaneous parts drawer, so that was really not a problem, either.

There turned out to be only one component that required soldering: the coaxial DC power plug. I had to read the schematic to determine that it is positive-tip polarity, and then we secured it.

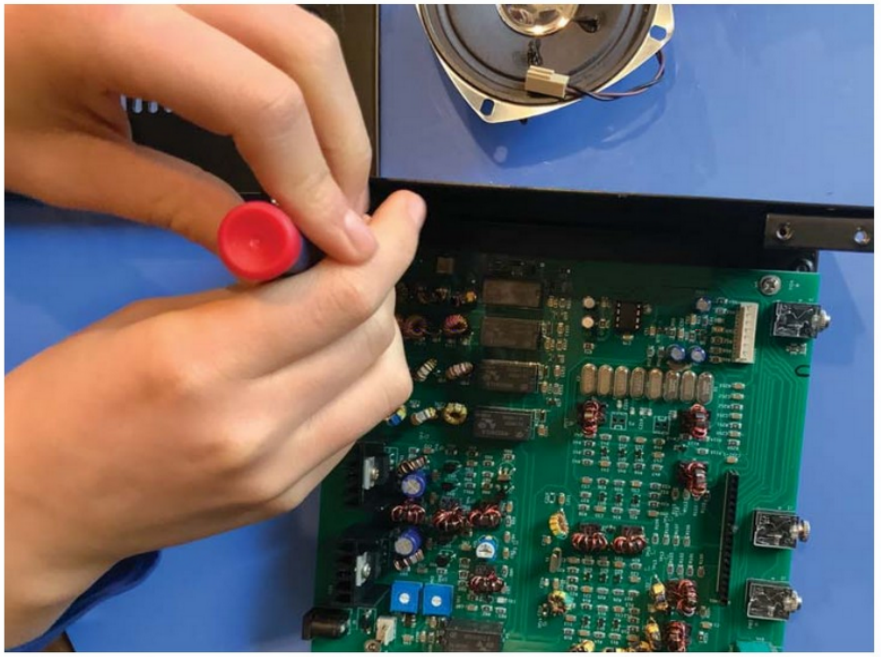
Powering it up

After assembling the radio, it was then simply a matter of applying power and turning it on with the volume control.

The first time I turned on the uBITX V6 transceiver, the rig was running on the initial software release (v6) pre-loaded at the factory. It's worth noting that within one month, uBITX released a *much improved version* of the software (v6.1) that included several minor fixes and dramatically increased the colour screen refresh-response time. The initial version took at least twice as long to redraw the screen as in version 6.1. The following observations all pertain to v6.1.

Speaking of the colour screen, the user experience couldn't be more straightforward: the display is laid out with the 'A' and 'B' VFO frequencies at the top of the screen. To use it, you simply touch the VFO you want to use. Below that, there are buttons for selecting RIT, USB, LSB, CW and Split. There are also dedicated buttons for the 80, 40, 30, 20, 17, 15 and 10m bands. The uBITX is not limited to those

The uBITX V6 kit was packed in a box and shipped with the contents and chassis very well protected.



My young daughters had no problems assembling the entire transceiver.

bands, however; in fact, it will also transmit on the 160m & 12m bands and more.

Immediately following the band buttons, there are dedicated buttons for controlling CW words per minute rate, CW tone, and a direct frequency entry button that, when pushed, brings up a frequency keypad.

Although my first experience tuning the uBITX was quite pleasant, I had read in advance (via the uBITX email discussion group) that I should *follow a frequency and BFO calibration procedure for optimal performance*. Because this process isn't well documented, I'll first share the steps for calibrating the frequency, and then the steps for calibrating the SSB/BFO.

To calibrate the frequency:

Tune to a known, stable reference frequency. I used WWV on 15.000MHz.

Press and hold the tuning encoder for several seconds to enter the settings screen.

The first item pre-selected in the Setup menu is 'Set Freq…' Press the encoder knob once to enter tuning mode, then zero-beat the signal. To save this, simply press the encoder once more.

On the main menu, turn the encoder until 'Exit' is highlighted, then press it once again.

Now the radio should display the correct frequency for your reference signal.

To calibrate the BFO:

Find an SSB signal. I chose a chat between two strong operators on the 40m band.

With a laptop or PC with a microphone nearby, go to the uBITX calibration page (www.hfsignals.com/index.php/bfo-tuning-aid/) and allow it to access your microphone.

Press and hold the tuning encoder on the uBITX for several seconds to enter the settings screen.

Next, turn the encoder to highlight the 'Set BFO…' and select by pressing the encoder.

Bring the PC microphone close (within an inch or so) of the uBITX's speaker. Tune the encoder until the noise graph on the web page is centered approximately between the two red lines of 300Hz and 3kHz, and the audio sounds natural. (If you don't see an audio graph on the uBITX page, it's likely that either the microphone isn't engaged on your PC/laptop/tablet/phone, or the script isn't running on the page.)

Once you're pleased with the audio, simply press the encoder to save the setting, then exit the settings menu.

Once the SSB had been calibrated, I found the audio quite pleasant.

On the air

Only moments after assembling and calibrating my uBITX V6, I plugged in my large horizontal skyloop antenna and had it on the air.

The rig only has three modes – Upper Sideband, Lower Sideband, and CW – obviously, the most useful modes for a portable QRP radio.

My first contacts were made using single sideband and the supplied Baofeng hand mic. There is no mic gain control, leaving the user to raise or lower his/her voice as needed – but that first day on the air, I easily made a handful of contacts on 40 and 20 metres. My signal reports weren't incredibly strong, but then again, I was only transmitting at perhaps 5-7 watts. I did receive good audio reports, fortunately, but I do believe the radio would benefit from some sort of compression function to make the audio a little 'punchier'.

There's one great feature on the uBITX I must mention. With microphone plugged in, if you press the 'CW mode' button on the display, then key down the 'mic PTT', the uBITX will transmit either a constant carrier or a string of 'dits' or 'dahs', depending on the keyer mode. I found this is incredibly useful when I wish to tune via my external ATU on a new band.

CW operators will be happy that the uBITX V6 supports straight key, plus Iambic A and B operation, via the built-in keyer. In addition, the sidetone's tone can be adjusted as well as the keyer speed. The sidetone volume is tied to the AF gain. Most importantly, the CW delay can be adjusted as well from 100 – 1000 milliseconds. While not full break-in QSK, when set to 100 milliseconds, T/R recovery is very quick. In short: the uBITX V6 has a surprising amount of adjustments available to the CW operator, providing quite a lot of control.

Thomas Witherspoon, M0CYI
thomas@swling.com