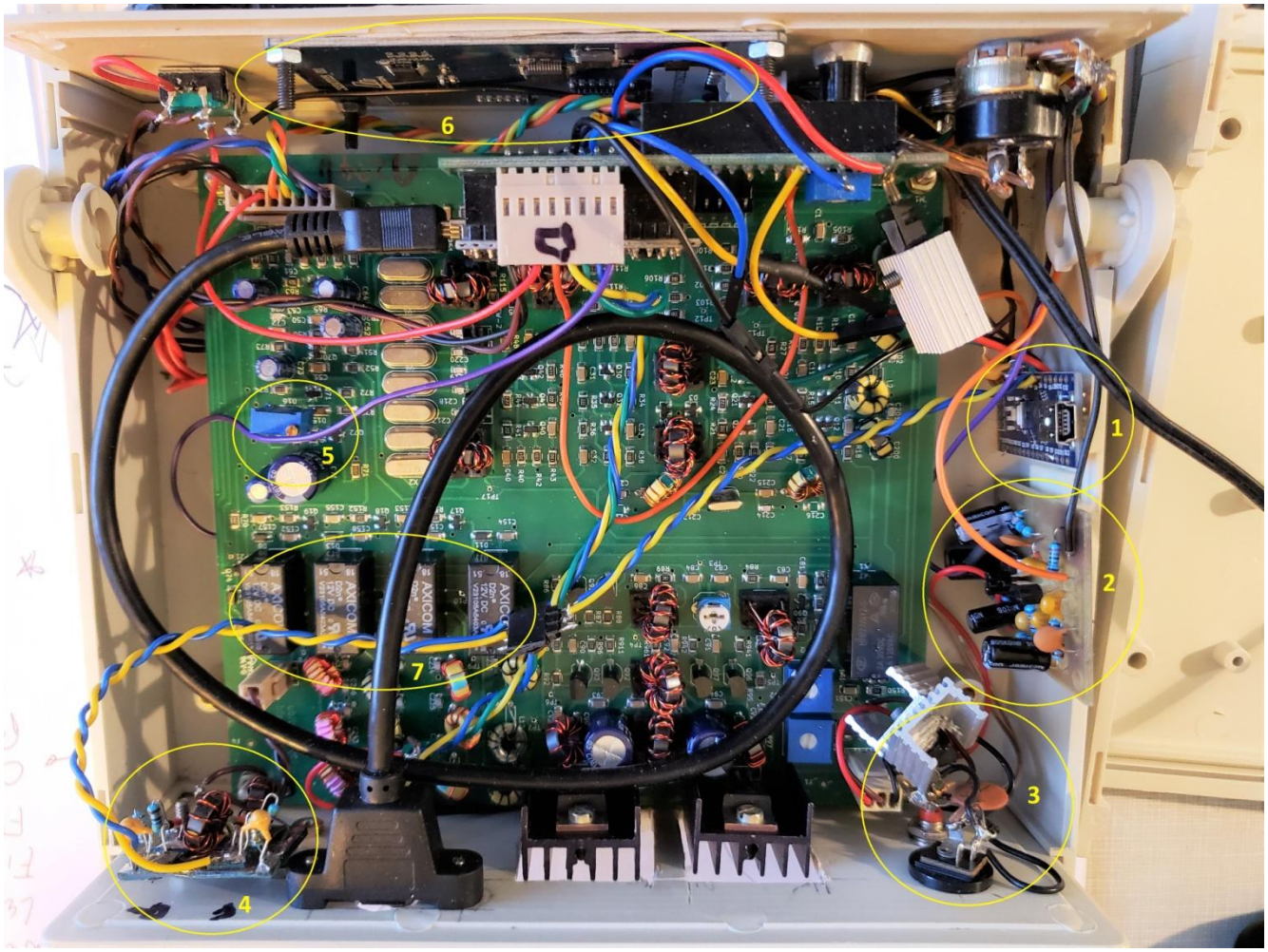
µBitX radio mods

In 2019 I had the fun of building a µBitX rig and added a color touch-screen. I've used that rig many times from home and portable at the beach, a Maker Faire, or at a hamfest using either hamsticks or a small portable loop during the past year. During that period I kept identifying further modifications I wanted for the radio, and over the year they have been getting included. Below is a picture of the inside of my rig with the various mods circled and described below:



1. This is an outboard Arduino Nano board that links to the main Raduino board and adds several features including: A DSP S-Meter (replacing a simple audio amp circuit), a CW decoder, SWR logic, and an audio spectrum display. This (like the incorporation of the touch screen) comes from the mind of KD8CEC and you can see it at http://www.hamskey.com/2018/07/standalone-signal-analyzer-i2c-type.html#more

2. An audio amp and AGC module came from a QST article in the July, 2019, issue on page 45, thanks to K1BQT. If you are a member you can look it up on the ARRL website. His article discusses replacing the stock audio amp circuit with this one, but I have a V4 µBitX, which uses discreet transistors in the audio section and has notoriously low audio output, so I used it as a preamp stage instead. The result is that I brought the audio up substantially for weaker signals and on higher bands, while controlling strong signals to a reasonable level. I fed this circuit with the audio normally going to the high side of the µBitX volume control, and used a 100K resistor from the output of this circuit back to the high side of the volume pot. I also increased the AGC delay substantially by changing a 2.2uf cap (C6 in his design) to 10uf to make for more

comfortable listening. However, because switching transients are still present ahead of the final audio amp, I needed to run 4 BA432 diodes in series across that 10uf cap to limit the maximum voltage to just under 3v and prevent momentary blanking of the audio when going from transmit to receive.

3. This is a simple 12v regulator using a 7812 chip to protect the main board (the PA circuit bypasses the regulator) and lower the load on the 5v regulator, especially when running higher voltages as in a car with the engine running. I found when running at 14v and transmitting continuously for over a minute this chip would overheat and shut down, so it has a couple of piggybacked heatsinks from my junkbox added to keep it happy.

4. I built a Stockton SWR bridge circuit, thanks to an ND6T design seen at http://www.nd6t.com/bitx/RF%20Monitor.htm. This generates the FWD and REV voltages sent to the Arduino described in no. 1, above, to calculate the SWR from the formula VSWR= (Vfwd+Vrev)/(Vfwd-Vrev) and display the result graphically when transmitting using the S-meter scale.

5. The use of discreet transistors in the V4 µBitX audio section in a push/pull circuit had some issues with crossover distortion (later versions used an LM386 chip, instead). Raj, VU2ZAP, identified a fix by adding a small pot to set the bias, which is shown here. His description of the fix is on the BitX20 user group, a great place for support of all the BitX radios, and his message is at this location: https://groups.io/g/BITX20/message/52181?p=,,,20,0,0,0::relevance,,crossover,20,2,0,22188783

6. This is the Nextion color touch screen originally proposed by Ian, KD8CEC on his website: http://www.hamskey.com/2018/06/how-to-use-nextion-lcd-1-start-nextion.html. You have to search around his site, but he describes how to modify the µBitX to use this display, how to program the display, and has his KD8CEC software for 2 and 4 line LCD displays as well as this color display available. His software adds many features to the µBitX, even using the original 2-line LCD as supplied with the V3, 4, and 5 kits. The new V6 µBitX has its own color touch screen that isn't compatible with this software, but some folks have already retrofitted the Nextion to the new version of the radio as the main board is electrically the same as a V5.

7. The V3 and V4 µBitX boards had a problem with spurious transmissions because of the conversion scheme used and crosstalk in the relays used to switch the various bandpass filters in and out of the circuit. These issues were solved in the later V5 and V6 boards, but almost by accident a person on the BitX20 user group discovered that the relays supplied were really designed for DC, not RF service, and by substituting this model of Axicom relay (designed for RF) the crosstalk was virtually eliminated. These are the substituted relays. I found replacing these as well as replacing ICs (I burned up a few LM386 chips experimenting before getting no. 2, above, to work) is almost trivial using a desoldering iron that was recommended to me on another forum. This one is cheap, but removes virtually all the solder from the pins and the part almost falls out of the board: https://www.amazon.com/Science-Purchase-Desoldering-Iron/dp/B00CUKTH2A/ref=sr_1_7?keywords=desoldering+iron&qid=1584390074&sr=8-7

The result of these mods does not change the external appearance of my µBitX, except for what is displayed on the screen, but they have made it a more capable and fun radio.

73, =Vic=
WA4THR