Software and Calibration

Jim Kortge K8IQY and Mike Maiorana KU4QO 10/30/2021

The following instructions cover building and installing the radio's software, followed by the calibration procedure.

If you don't already have it, install the Arduino IDE from https://www.arduino.cc/en/software Follow the instructions to install the IDE. As of this writing, the latest version is 1.8.16

Download the code zip file from GitHub at https://github.com/ku4qo/ss-40-Digital-Transceiver. Click on the "Code" button, then "Download ZIP". The zip file received includes all the software, along with all the other project data. Unzip the file and store it somewhere on your computer. Create a folder named "ss-40-V4" in the folder that contains the Arduino projects (on my computer in Documents\Arduino) and copy the 7 .ino files from the Zip file to that folder. If you double-click on the ss-40-V4.ino file it should launch the Arduino IDE and open the project (you should see 7 tabs at the top of the program window).

Next add the required libraries. Click on "Tools" then "Library Manager". Once the library window pops up, search for "LiquidCrystal_I2C". Install the matching library (by Mako Schwartz, version 1.1.2). Then browse to https://github.com/pavelmc/Si5351mcu to download the Si5351 library zip file. Install it in the Arduino IDE by clicking "sketch" then "include library" then "Add .ZIP Library". After installing you should see it listed under "sketch" then "include library" then "Contributed Libraries" as Si5351mcu.

Once all the libraries are installed, compile the project and make sure there are no errors. As of software version 07282021 the sketch uses 44% of program memory and 31% of RAM, leaving plenty for future enhancements.

Install the built sketch to your Arduino Nano via the USB port, or the debug port if you don't want to run the Arduino bootloader.

For calibration I suggest you only connect the receiver and the controller, leaving the transmit and attenuator boards unconnected for now. When you power it up you should see a brief splash screen with the project name and code level, then you should see the normal display with the frequency and modes.

The calibration routines consist of three parts, calibrating the Si5351, centering the IF filter and setting the BFO side tone frequency. The calibration is saved in the Arduino Nano's EEPROM and is loaded at boot time. The calibration function is started by holding down the Enter button and turning on the receiver. Hold Enter down until the system boots past the splash screen to the

normal receiver display, then release the Enter button. You should see the initial calibration screen. Tap the Enter button to continue to calibrate the Si5351. The clk0 output of the Si5351 will be set to approximately 10MHz (should be within a few kHz of 10MHz before calibration). Use a frequency counter to measure, or a calibrated receiver to zero-beat the clk0 clock. Adjust the frequency using the encoder to exactly 10.000000MHz (or as near as possible). You could also zero-beat with WWV to adjust the 10MHz clock. Use the Up and Down buttons to change the step size of the encoder (notice the carat "^" moves to point to the digit that the encoder affects). Once you are happy with the 10MHz calibration, tap the Enter button again. The display will show the calibration offset for the Si5351 which will be saved in EEPROM. Tap Enter one more time to continue to the next step.

This step finds the center of the 11MHz IF filter and saves that value in EEPROM. This data is used for correctly setting the VFO and side band selection in software. Tap Enter again to start the calibration. The display will show a frequency near 11MHz which is its initial guess at the IF filter center frequency. During this step, the system also turns on the clk1 output (normally used as the transmit clock) to produce a signal for the receiver. Depending on how close the IF frequency is and the setting of your BFO trim cap on the receiver, you may hear a tone in the speaker. Turn up the audio volume and try to detect it. Since the BFO is not adjusted by the computer, moving the IF frequency also changes the side tone pitch. If the side tone is still not audible, try adjusting the BFO trim cap to see if you can find it.

As you adjust the IF frequency the received side tone will change in pitch and volume. You want to measure the level of the side tone as you sweep across the IF filter. An RMS volt meter can be useful for this. Find the upper and lower -3db points, then adjust the IF frequency to the middle of that. Alternatively, if you have already characterized your particular IF filter, simply adjust the display to that frequency. My beta receiver has an IF center frequency of 11.000150 MHz.

Once you have the IF centered, now adjust the BFO trim cap to the desired side tone that you like.

This completes the calibration. Tap the Enter key again and the radio will resume normal receive operation using the updated calibration information. An easy way to test the calibration is to tune in a broadcast of a known frequency (I use ARRL code practice as it is quite loud here in the evenings). I adjust the receive frequency to exactly 7.0475MHz and ensure my side tone is where I like it (600Hz for me). Then I swap side bands with a long press of the Up button. When you release the Up button the side band should change (USB to LSB). The side tone should stay pretty close to what it was on the first side band. If it is within 10Hz or so, your calibration is good.