SDR Cube Transceiver

OPERATOR'S MANUAL

A portable, standalone SDR transceiver for voice and CW using the Softrock RxTx 6.3 for the RF front end, and embedded Digital Signal Processing for the HF modem.

Optimized interface provides for digital modes with the NUE-PSK Digital Modem.

Version 0.8 (DRAFT)

CONTENTS

1 Overview

The SDR Cube is a totally self-contained, embedded SDR transceiver for CW and SSB operation using a Softrock for the RF front end and a pc board implementation of an HF modem. A personal computer (PC) is not needed for using the SDR Cube, as all DSP processing is accomplished by an embedded processor on the three internal pc boards. The SDR Cube is designed to fit into an optional 4" x 4" x 4" pre-cut black powder-coated aluminum enclosure containing all controls, a blue graphic display showing the transceiver settings and an exciting 8 kHz-wide band scope of spectrum signals, and the popular Softrock RXTX v6.3 board. The SDR Cube may also be used with any of the 11,000 other Softrocks in the field today! Sold as a kit or as a fully assembled & tested transceiver, the Cube will be available starting in November 2010. We presented on the SDR Cube at TAPR's annual "Digital Communications Conference" in Portland, Oregon, and the DCC Proceedings containing our paper are available from TAPR. Many details are already presented at the product website (www.sdr-cube.com), and active email discussion occurs in the SDR-Cube Group in Yahoo Groups.

Figure 1: The SDR Cube Transceiver is dimensioned at 4" x 4" x 4", containing a full user interface, DSP processing, I/O connectors on the rear panel, and 27 in^3 shielded space internally for Softrock.

Figure 2: The rear of the SDR Cube contains the usual complement of I/O connectors, including a specialized one for optimized use of the NUE-PSK modem for digital modem support.

MAJOR FEATURES:

- Standalone SDR transceiver ... no PC, portable, compact
- Self-contained single band ... based on the I/Q RF front end
- Softrock-compatible ... designed to interface with SR v6.3 RXTX, etc.
- Low Power ... 90ma (Cube), plus 100 ma (Softrock Rx) or 300 ma (Software Tx)
- Add-on RF Amp & Attenuator ... good control of incoming RF, optimize some SR features.
- Quadrature Sampling Clocking options ... DDS, Si570, or I2C to target Softrock
- Built-in Keyer ... 1-50 wpm, Iambic A, B, or straight key
- Popular HF modes ... SSB, CW, AM, Digital (with special interface to NUE-PSK for digital modes)
- Special interface to NUE-PSK Modem ... digital interface provides best quality
- Graphic LCD Display ... Provides clear indications of the many status and options
- Bandscope ... provides +/- 4 kHz spectrum visibility for Rx, signal monitor for Tx
- Audio filtering ... low corner 200Hz, high corners 700, 1500, 2400 or 3600Hz
- Audio Output ... Headphones or amplified speaker, Binaural Audio
- Beeper ... User interface clicks, code practice oscillator, and more

- Frequency agile ... Fast/Med/Slow tune, dual VFOs with 20 memories, RIT/XIT
- Menus ... Calibration, all settings, system gain, sidetone frequency, etc.
- Software Upgradeable ... Bootloader enables user to load new software versions
- Open Source ... Take the source code and add your own features

2 Background

Software Defined Radio (SDR) in the Amateur Radio community has been making great strides in recent years. From the innovative and ground-breaking products of Gerald Youngblood, KSDR of Flex Radio some four years ago, to the most recent state-of-the-art designs in the HPSDR group, SDR technology has really now come of age. This current state-of-the-art has also been greatly enabled by the tireless work of Tony Parks, KB9YIG, the father of the Softrock designs, who has empowered thousands of hams worldwide with his series of inexpensive radios that work with a PC sound cards.

Each of these radios in their most basic form consists of electronics that sample the incoming RF after it has been converted down to baseband and send the results to a PC for digital conversion. The PC then performs the complex demodulation computations so we operators can understand the SSB, AM or digital mode communications coming in. And of course the reverse happens for transmit, whereby the PC presents electronic signals to the front end electronics for mixing and ultimate transmission.

However throughout all the excitement of PC-based software defined radio, there has also been a quieter background quest for a form of SDR that is not tethered to a PC. While the PC offers seemingly unlimited processing power, gorgeous user interfaces and lots of memory, the PC is still an expensive and cumbersome accessory to take to the field when the need arises for portable operations. Classic problems are encountered with regard to the ability to see the PC display in bright sunlight, and powering the PC and the power-hungry SDR front-end electronics is tough with limited-capacity batteries. In general, lugging around an expensive and delicate laptop is not something one wants to regularly do. Also, many hams just do not care to operate a ham radio with a mouse controlling knobs shown on a on a PC screen.

However most would still agree that the *performance* of such a PC-based radio is of great value, given the flexibility offered with SDR’s innate ability to handle virtually any operating mode – SSB, AM, CW, and digital modes – with just a new software file or program loaded into the radio.

Numerous experimenters have been seeking to develop a portable SDR transceiver to address the shortcomings of using a PC in the field, while retaining enough of the benefits of SDR in general so as to have an inexpensive-yet-powerful transceiver. Embedded SDR projects are in progress with in the High Performance SDR Group (HPSDR) and yet another is being described in an article series in QRP Quarterly Magazine. Many of us experimenters are standing on the shoulders of previous pioneers in the field of DSP for Amateur Radio: Rob Frohne KL7NA, Lyle Johnson KK7P, Rick Campbell KK7B and others. The handbooks from the ARRL and RSGB, as well as the seminal work of “Experimental Methods for RF Design” by Hayward/Campbell/Larkin, are replete with the basic building blocks that we designers of today are employing in our SDR implementations.

This paper chronicles and describes the path that our team has been taking, resulting in what we call the “SDR Cube”.

3 Brief History of the Design

The genesis of this embedded DSP approach to a Software Designed radio has its root in Finland, at the shack of OH2NLT (co-developer of the SDR Cube). Over the course of four years, Juha Niinikoski had been experimenting with many of basic building blocks to show how a “cheap DSP” solution can be used to create a ham radio. In fact, using that very name for his progression of early designs with Microchip’s dsPIC processors, Juha was able to demonstrate a minimalist working radio based on a dsPIC30F processor to control the phasing of two quadrature-related sampled data streams.

The CheapDSP results were very encouraging and early in 2009, George Heron N2APB (the other co-developer of the SDR Cube) learned of Juha’s work and it became something that had to be duplicated in his own workshop in Maryland. Heron had always been focused on providing portable radio solutions, as evidenced by his recent co-design of the NUE-PSK Digital Modem, which also used the dsPIC processor.

In the course of replicating the OH2NLT CheapDSP controller, George and Juha began collaborating on an evolved and more complete SDR implementation, this time using the more capable dsPIC33F processor at the heart of the design. Using this Microchip controller provided greater processing power, more input/output pins for an integrated and capable user interface, and a way to inherit some of the features and drivers that made the NUE-PSK modem so popular – namely, the spectrum display and FFT processing, the drivers for the rotary encoder, and field programming and bootloading.

Together, N2APB and OH2NLT embarked on an development campaign, working "transoceanic" and across many time zones to ultimately create the encompassing SDR Cube Transceiver.

4 SDR Cube Overview

The SDR Cube is a single-band QRP transceiver consisting of an embedded DSP controller coupled with the Softrock RxTx v6.3. Sized at 4" x 4" x 4", the appropriately-named Cube also contains a full complement of built-in user interface: graphic LCD for spectrum display, typical controls for frequency, mode and signal management, and I/O connectors for connection to the outside world. The Cube design is optimized to internally accommodate the popular Softrock RXTX v6.3 electronics. Different from other experimenter "single board" solutions, the Cube was designed from the start to serve as a full transceiver with minimal component hassle imposed on the builder.

4.1 Software

The design is based on the classic "phasing method" of SSB generation, whereby a 90-degree delay is imposed on the source audio signal coming from the microphone, or from a NUE-PSK Digital Modem. The resultant signals are applied to a pair of mixers – in our case these are provided on the Softrock board – with 0 and 90-degree LO signals. The output of the mixer is amplified on the Softrock and presented to the output BNC connector on the rear of the Cube.

The receive path works in a similar manner, but in reverse. The received signal is mixed with LO signals on the Softrock card that are at 0- and 90-degrees, producing the I and Q quadrature signals. These component signals are digitized by the Cube hardware and the dsPIC controller imposes a 90-degree phase shift between them by passing them through two balanced and matched FIR filters, one advancing the signal 45 degrees and the other delaying the signal by 45 degrees. Maintaining accurate phase delays and corresponding amplitudes have a direct impact on the quality of the demodulated signal. The demodulation occurs at the summing junction, shown in Figure 2 below; when the dsPIC performs the computations for the desired mode, the receive audio is able to be heard in the headphones.

Note that either USB or LSB may be demodulated merely by reversing the I and Q signals where indicated in Figure 2.

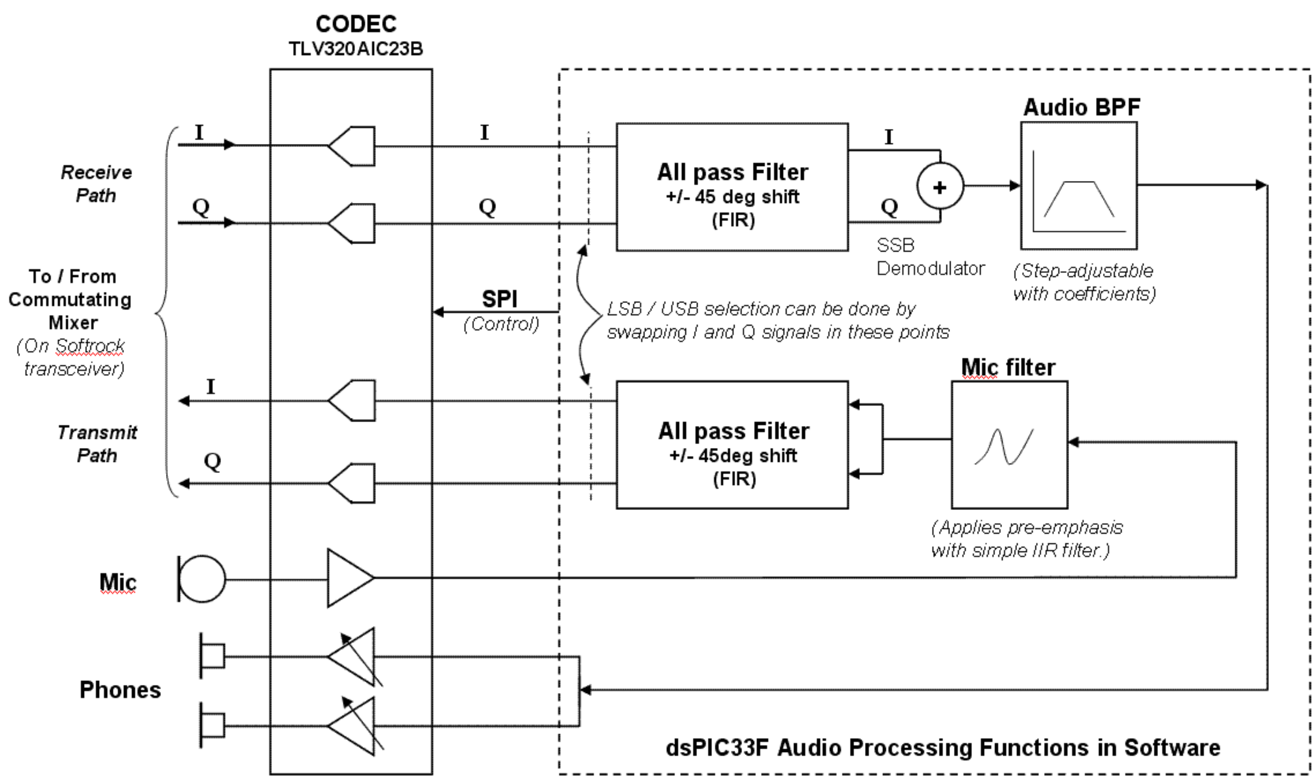

Figure 2: Software Processing Block Diagram

4.2 Hardware

The Microchip dsPIC33F at the heart of the design, taking inputs from many user controls, displaying text and graphics to a 128 x 64 pixel graphic LCD, and implementing the audio data paths through the codec and over to the Softrock board on the right.

The dsPIC33F is a 40 MIPS controller (40 million instructions per second), which comparatively is only moderately capable with today's technology. But unlike some of the more powerful embedded DSP controllers from Texas Instruments or Freescale, the dsPIC package is relatively straightforward for homebrewers to attach on pc boards. Even with the 100-pin package of the dsPICmodel we use is able to be attached in the conventional "flood solder and wick off" technique. Microchip offers a free, very capable and intuitive development environment called MPLAB, enabling experimenters to focus on the design and not the tool. For these reasons, the dsPIC controller remains popular in the experimenter community.

The dsPIC offers enough capabilities for designers to provide some demanding solutions: a 16-bit fixed point architecture, yielding 96 dB dynamic range in its computations. It has 40-bit accumulators and hardware support for division operations, barrel shifters, multipliers and a large array of 16-bit working registers – all of which provide ample DSP power for the operations performed in such an embedded SDR. Admittedly, the dsPIC controllers are getting "long in the tooth", however microchip is committed to continuing the product line, perhaps due to continuing applications like those here in the ham radio community.

For the important A/D and D/A conversion block we used the Texas Instruments TLV320AIC23B high performance stereo codec. In our first implementation, 8 kHz sampling is used, as anything much more than this is difficult for the dsPIC to handle, and a higher sampling rate has a two-fold performance cost: (1) Filtering must be performed more often (at the pace of the sample rate); and (2) More filter taps, resulting in more computation time, is required at a higher sample rate, which is a subtle and often-overlooked consideration in design. In later versions of Cube software we will be implementing additional processing hardware to accommodate increased computational power; for example providing a wider spectrum band scope and digital modem functionality. So keeping the computing overhead lower is just as important as achieving adequate quality. The integrated headphone amplifier, programmable gain microphone amplifier, and SPI control bus were important selection criteria for this codec.

Optimal gain distribution in the system is not necessary known yet, and AGC is an important (and difficult) feature to implement. Thus we have started getting our hands around receive path gain by implementing a controllable RF Attenuator as a plug-in for the Softrock board in place of its existing BFP board. In this way we can have several levels of signal attenuation to assist in establishing optimal signals throughout the system.

Clock generation is optionally provided for on the DSP board. In fact, because of the difficulty that homebrewers have in obtaining the ideal clock generator (Si570), we have also provided for use of the classical DDS signal generator.

Special connection to the NUE-PSK Digital Modem is provided in order to optimally interoperate with the modem to provide digital mode capability for the SDR Cube.

The display system is a step up as compared to conventional minimalist SDR implementation that uses a character-based LCD. As illustrated in the Cube screen graphics we are able to provide a multitude of graphical representations of system status. Additionally, and most exciting is the ability planned for the Cube to display a spectrum band scope function that is a graphical representation of the 2.5 kHz swath of spectrum being processed. Admittedly, this close-in view of the spectrum is not as sexy as the 25 kHz spread offered by commercial rigs; but using only 8 kHz sampling presents limitations, and various tradeoffs are necessary in embedded solutions. This is an area we intend on improving in future iterations of the design – either by improving the processing software or by substituting a more powerful DSP in place of the dsPIC.

5 Design Goals & Features

It was important for us to understand the natural constraints offered in embedded systems with limited resources and computing power. The designer's goal is always to provide "just enough" without over-taxing the available resources; or conversely without creating a radio with more bells and whistles than are actually warranted.

5.1 Portable, standalone, QRP-level, all-mode SDR transceiver for Amateur Radio

a. Portable: 12-volt battery operated, easily transportable, hand-held form factor for convenient field use *(EmmCom, Field Day, Trail usage, etc.)*

b. Standalone: Design uses embedded microcontroller for all signal processing - no PC or Laptop required. *(Decouples the product from PC complexities, cost & usage concerns.)*

c. Band Coverage: 160-10 meters (1.8-30 MHz). *(Base design provides for 20m BPF/Output modules, with other plug-ins accommodated.)*

d. Low Power: Low current draw from power source, approx. < 500 ma. *(Maintains battery life in field use.)*

e. QRP: RF transmissions < 5 Watts, typical. *(QRP output levels are achievable in small form factor. Can later add options for power amp.)*

f. Modes: Voice, CW, and select Digital modes. Digital modes achieved by interoperation with NUE-PSK Digital Modem. *(These modes cover the wide range of anticipated user needs: bench/field use, casual use, EmComm use, etc.)*

5.2 Built-in Transceiver "RF front end"

a. QSE/QSD-based quadrature signals provided to HF modem for all-mode modulation/demodulation of signals. *(Easiest , least expensive and most convenient architecture for implementing SDR.)*

b. Softrock RxTx 6.3 transceiver assumed in base design of enclosure. *(Best performing and most compact Softrock transceiver. 50 kits already stocked for this SDR use.*

c. Other Softrock models or other QSD-based transceivers able to be plugged into core signal processing of the HF modem. *(Leverage the > 10,000 Softrocks already in the field, lowers the user's cost of ownership by optionally selling the built-in RxTx)*

5.3 HF Modem

a. Microchip dsPIC33FJ used as the primary embedded signal processor performing the HF modem functions *(Software architecture available, easy to port in critical functions from digital modem: display, USB, bootloader, keyboard)*

b. TLV320AIC23B codec used as the multi-channel, gain-controlled analog-digital-analog signal conversion *(Driver available, sufficient bits & sampling)*

5.4 Form factor

a. Size: Small, driven by graphic LCD and controls, also including Softrock. *(Facilitates convenient and integrated portable operation.)*

b. Material: Aluminum. *(Needed for RFI shielding)*

5.5 Software Field Upgradeability

a. Integrated bootloader (*Allows user to download improved software versions from the website and load into SDR.)*

6 Operator Controls – The Front panel

The user interface of the SDR Cube was carefully designed to provide the maximum amount of control of the transceiver, while using a minimum-yet-convenient number of physical controls. The main focus of the interface is on the graphic display (for real-time status and tuning) and the tuning dial for frequency agility.

In order to provide additional functions for the five pushbutton-capable controls, we adopted the technique of accessing the main function with a tap of the control, and the secondary function with a long press of the control. In this way, for example, the user is able to tap the Rate pushbutton to select the desired tuning rate as displayed in the display, or long-press the rate pushbutton to put the SDR Cube into tune mode (continuous carrier transmitted). The main function label is shown first, and the sub-function is shown below the main and in parentheses.

6.1 Display

The attractive, blue-screen graphic LCD serves many status display functions. The display has a backlight that may be turned on/off via the Configuration Menu.

Bandscope – The top portion of the display is the band scope area, where an 8 kHz swath of spectrum is displayed, centered at zero in the middle of the display. When the frequency dial is moved, the spectrum energy (i.e., signals, noise, etc.) is moved up or down along the spectrum relative to the center zero frequency point, which is numerically displayed as the kilohertz frequency below the spectrum. In essence, the signal indications in the spectrum display are the audio signals after demodulation. So a CW tone of 800 Hz will appear as a small, narrow "pyramid" to the right of center. Similarly, the multi audio frequency voice energy of an SSB signal on upper sideband would appear to the right of the zero point, as shown in the photo above. When LSB is the selected mode, the signals appear to the left of the zero point. Marks are located along the top of the band scope, marking the 1,000 Hz points for easier signal identification.

Frequency Display – The current operating frequency, in kHz, is shown in large digits on the display.

The Tuning Rate – The tuning Rate is displayed to the right of the frequency, and indicates 'fst', 'med' or 'slo' corresponding to the three tuning rates able to be selected by the Rate pushbutton. See the explanation below for use of the Rate control.

6.2 Frequency / (Mem Select)

The center large Frequency dial is the way that operating frequency is changed, as well as option selection for various sub-menus. It is a smooth rotary encoder.

When a long press is made on this control, the twenty different **Memories** are able to be accessed. Two memories, or VFO settings (A and B) are actually associated with each memory bank. When the control is turned while being pushed in (i.e., push-hold-turn) any of the memories in the Cube may be selected as the main operating memory, as shown at the VFO position in the center-right of the display. It may be convenient to designate certain groups of memories as being set with frequencies for a certain band, thus enabling quick change to different bands or band segments. Once the desired memory number is seen in the display, releasing the control sets the main operating frequency to that memory setting in either VFO A or VFO B associated with that memory.

6.3 Rate / (Tune)

The **Rate** control is used to adjust the step rate of the frequency when the dial is changed. When in fst (fast) mode, the frequency changes in 100 kHz steps when the dial is moved. When in 'med' (medium) mode, the frequency changes in 1 kHz steps when the dial is moved. When in 'slo' (slow) mode, the frequency changes in 10 Hz steps when the dial is moved. Thus by tapping the Rate control to select the desired tuning rate or granularity, one is able to quickly move about the band to find and tune in signals. For example, transitioning to the (upper) voice portion of the band from the (lower) CW portion, one would tap the Rate control to get 'fst' displayed and then change the dial. Then in order to zero in on a desired signal seen in the band scope, the Rate control is tapped to display 'med' and then moving the dial results in smaller 1 kHz steps which is often good for moving a selected signal into the passband of the rig. Then finally in order to fine tune the signal to get the SSB voice at the right pitch, one would tap the Rate control to select 'slo' thus allowing very fine movement of the signal in the rig's passband. After just a little bit of use, we found that our "test ops" quickly got the hang of using the Rate control to effectively and quickly move about the band.

The **Tune** control is accessed by a long press of the control button. When activated, the transceiver begins transmitting a continuous tone. This is useful for tuning other equipment connected to the SDR Cube, such as an ATU or an amplifier. Taping the control or the CW paddle will terminate the Tune mode and drop the Cube back into receive mode.

6.4 Menu / (Lock)

The **Menu** function is accessed with a tap, and the Configuration Menu is brought up on the display for subsequent selection. See the Configuration Menu section for details.

When one long presses this control, the **Lock** function is activated. In this state, most user interface controls are locked in order to prevent unwanted state changes. Useful when engaged in a QSO and you don't wish to inadvertently change frequency by bumping the Frequency dial. The lock mode is terminated by another long press of the control.

6.5 Mode / (Save)

The **Mode** of the SDR Cube is selected by tapping the Mode control. Currently supported modes are CW, USB and LSB.

When a long press is made on this control, all user settings (memories, VFOs, and current operating frequency) are **Saved** to nonvolatile EEPROM memory. When the Cube is again powered up, these settings are pulled from EEPROM and put in place on the transceiver.

6.6 VFO / (RIT)

When the **VFO** control is tapped, the VFO setting is toggled between A and B. Each VFO is able to hold a different frequency, and this control can provide convenient "split" operation. To set up the A and B VFOs, select the A VFO and move the Frequency dial to the desired point. Then tap the VFO control to select the B VFO and again move the Frequency dial to another desired point. Press-hold the Mode/(Save) control to save the specific momory's A and B VFOs for use in the future.

In order to lock in one of the VFOs, either A or B, press-hold the control to designate the VFO as the stationary memory. Then "**RIT ON**" is displayed and that VFO is held stationary while the other is able to move. This capability is convenient in the case of being in QSO with a station that is drifting. You want to keep your transmitted frequency unchanging, yet be able to follow the frequency of your drifting mate.

6.7 AF Gain

This conventional potentiometer control provides audio output level control from a maximum of 6 dB down to a minimum of -73 dB, as shown in the display at the "AF" location.. When turned completely counter-clockwise, the audio is fully muted, and "Mute" is displayed.

6.8 Keyer

This conventional potentiometer control provides for varying the speed of the built-in electronic keyer. The range of the control is from ___ to ____ wpm.

6.9 Filter

Adjustment of this conventional potentiometer control provides a varying audio filter to the audio output of the transceiver, thus aiding in the reception of SSB and CW in crowded band conditions. Common filter widths provided are: 2.6 kHz, 1.5 kHz, 500 Hz. *(Values to be corrected.)* The corresponding filter graphic is shown in the bottom portion of the display for quickly identifying the filter currently in place.

6.10 RF Gain

The RF Gain is adjusted by means of a conventional potentiometer control. Two functions are wrapped into this control: changing the state of the two cascaded attenuator pads on the RxAmp board of the Softrock, and changing the Codec gain settings for the receive path. Thus a wide range of attenuation may be achieved with a single control.

6.11 Beeper

This miniature piezo speaker, located behind a hole in the front panel serves multiple purposes: "key click" for changes to menu settings, side tone for use as a code practice oscillator, and as a beeper to indicate possible error conditions occurring in the SDR Cube.

7 Interconnections – The Rear Panel

7.1 +12V

12V DC power input. 2.1mm jack. Power requirements: SDR Cube (80ma), Softrock RXTX 6.3 receiver-only (+120 ma), RXTX transmit (+100ma).

7.2 On-Off

Toggle switch turns the power on and off.

7.3 NUE-PSK

Optimized digital connection for the NUE-PSK Digital Modem. 8-pin miniDIN jack.

7.4 Phones

Jack for standard 1/8" stereo plug headphones or external amplified speakers.

7.5 Paddle

Jack for standard 1/8" stereo plug on paddle or key.

7.6 Mic

Jack for standard 1/8" stereo plug for microphone (with PTT)

7.7 Aux

Jack for standard 1/8" stereo plug cable providing RS-232 signals to PC serial port for Bootloading feature

7.8 Antenna

BNC jack for antenna feedline

7.9 External Softrock

15-pin D-style connector for connecting the Cub signals to an external Softrock. (Note: Not pictured in the photo of the prototype enclosure above.)

8 Configuration Menu

Many infrequently-used or one-time configured transceiver parameters are accessed by taping the Menu pushbutton on the front panel. When tapped, the following list of items is displayed in a circular list manner, showing the individual settings currently in place for each. In order to change a given setting, the main dial is turned until that setting is displayed with the cursor (>) next to it, and then may be changed by pressing the dial. At that point the parameters allowable settings are displayed in a list that is again controlled by the dial. Once the desired setting is displayed for the parameter, it may be selected by pressing the dial pushbutton. Once the desired settings have been changed to one's satisfaction, the user presses the Menu pushbutton to exit and the settings are saved to nonvolatile EEPROM memory.

8.1 CW Sidetone

Allows the audio pitch of the CW sidetone to be changed. (Only the sidetone in the piezo speaker in the front panel is affected.)

8.2 CW RF Pitch

Changes the offset of the CW frequency generated during transmit. This is the offset from zero beat that the receiving operator would hear for the SDR Cube's transmitted signal.

8.3 Binaural RX

Changes the delay imposed between the left and right audio output channels, thus simulating binaural receive function. This feature enhances many operators' listening experience and enables improved selectivity in the brain while listening to a crowded band of CW signals, and provides a more rich and enjoyable listening experience for SSB signals.

8.4 Terminal UART Baud

Presents various baud rate settings for use with special PC terminal functions and applications

8.5 PSK UART Baud

Presents various high speed baud settings for the high speed digital link to the external NUE-PSK Digital Modem.

8.6 Keyer Type

Allow selection of Iambic Mode A, Iambic Mode B, Dot Priority, or Straight Key modes.

8.7 Display Backlight Timer

Allows setting of a timer for turning off the display backlight. When the timer count expires, the backlight is automatically turned off, but any "touch" of a front panel control retriggers the timer and turns the backlight on again for that period of time.

8.8 Beep Length

Adjusts the length of the sounded beep, or click, from the piezo speaker mounted in the front panel.

8.9 FIR Gain I

A setting done one time upon initial SDR Cube calibration.

8.10 FIR Gain Q

A setting done one time upon initial SDR Cube calibration.

8.11 I/Q X-Gain

A setting done one time upon initial SDR Cube calibration.

8.12 Codec MAG

Provides variable gain levels in the Codec receive path.

8.13 FFT Window Type

Allows selection of different FFT processing algorithms for display of the band scope spectrum display: Rectangular, Blackman or Hanning.

8.14 DDS Reference Oscillator

A setting done one time upon initial SDR Cube calibration.

8.15 LO Type

Provides for the selection of either onboard clock options (DDS or Si570) or control of an off-board Si570 clock via the I2C control bus.

8.16 LO Multiplier

Provides for 1x, 2x or 4x clock generation, depending on the needs of the specific Softrock being used.

9 RF Front End – Softrock