Red Pitaya 2-Port VNA / HF SDR Transceiver

Gregg Daugherty - WB6YAZ

gregg-daugherty@comcast.net

3221 Nowata Ave, Point Pleasant, NJ 08742

7/21/17

Having recently retired, I was looking for a good project to get back into ham radio. I came across the Red Pitaya, ([www.redpitaya.com](http://www.redpitaya.com)) a dual high speed ADC / DAC / Xilinx SoC board that comes supplied with an interesting suite of standard tools (1-50 MHz oscilloscope, spectrum analyzer and waveform generator).

An SDR transceiver FPGA design (<http://pavel-demin.github.io/red-pitaya-notes/>) has also developed which is compatible with PowerSDR mRX PS, a windows SDR application based on the FlexRadio Systems PowerSDR program.

I was mainly interested in the Red Pitaya as the basis for an HF 2-port vector network analyzer (VNA). This capability would allow one to test two port devices such as filters and amplifiers as well as antennas. The Red Pitaya also comes with a SCPI server which allows for easy programming via high level languages such as Matlab and Python.

My thought was to develop a PWB that would support both the 2-port VNA and the HF SDR transceiver function. Figure 1 is a representation of the Red Pitaya. The board is available in two versions, either 10 bit or 14 bit 125 MSPS ADCs. The 10 bit version is a bit less expensive. Both versions also provide 16 bits of digital I/O which can be used for controlling switches, etc.

Figure 2 is a block diagram of the dual use PWB. I had several dual SPST relays lying around which I used for the RF switching functions. The board allows for switching the inputs to the Red Pitaya ADCs so that they can be utilized both for the VNA function and for the dual receive functionality provided by the PowerSDR application.

Figure 3 shows the PWB schematic and layout. DAC2 is used for the VNA function and DAC1 is used for the transmit portion of the SDR transceiver. The maximum DAC output level is 2V p-p, therefore an amplifier is needed to achieve sufficient transmit power. I modified my IC-751 RF output stage to accept the DAC1 output and thus amplify the transmit level to 100W. The SDR implementation provides PTT in / out via the digital I/O. There are also several other functions available for controlling additional HPSDR hardware such as the Alex module (<https://openhpsdr.org/>).

The 2-port VNA is comprised of two MiniCircuits dual directional couplers and three of the RF switches. DAC2 is used as the source for the VNA. The ADC1 input is used as the reference for ports 1 and 2, ADC2 for the reflected signals. Relays were used as the RF switches as the absorptive RFIC switches available do not provide a good 50-ohm termination for the unused ports at low frequency.

I used Matlab for the VNA programming language as the SCPI server provides easy access to the Red Pitaya imbedded functions (<http://redpitaya.readthedocs.io/en/latest/appsFeatures/remoteControl/remoteControl.html>). I will post the Matlab files to my github account (https://github.com/greggdaug). The software is still a work in progress and provides endless hours of entertainment during retirement. A more elegant (and less expensive) approach would be to use Python, this is also on my retirement learning list. For those interested in FPGA design, the Red Pitaya documentation (as well as Pavel Demins webpage) provides several examples. The Red Pitaya runs Linux directly and there are many examples for imbedded C code as well.

Figure 4 provides pictures of the hardware enclosure. Fans were included to cool the Xilinx SoC and amplifiers on the PWB. Inexpensive SMA connectors and cables can be found on EBay. Figure 5 is a 2-port VNA measurement example. The DUT is a MiniCircuits SLP-30 low pass filter. 12-term error correction has not yet been implemented as I am still wading through the math and Matlab implementation.

Figure 6 shows the PowerSDR program receiving a 5MHz WWV signal. Figure 7 is a screen capture showing the Red Pitaya web interface display.

Reference:

Network Analyzer Error Correction and Calibration Methods (Doug Rytting – Agilent Technologies, now Keysight)

Many thanks to Keysight for usage of their Advanced Design System EDA software package.