A low effort .1 – 2.5 GHz dBm power meter

Chuck MacCluer W8MQW

w8mqw@arrl.net

08-14-2019

A simple dBm meter

In the event that you missed Andy Nehan's clever RadCom article on a simple UHF AD8307-based power meter [1], I present here an even simpler AD8313-based meter for SHF. See Figure 1.

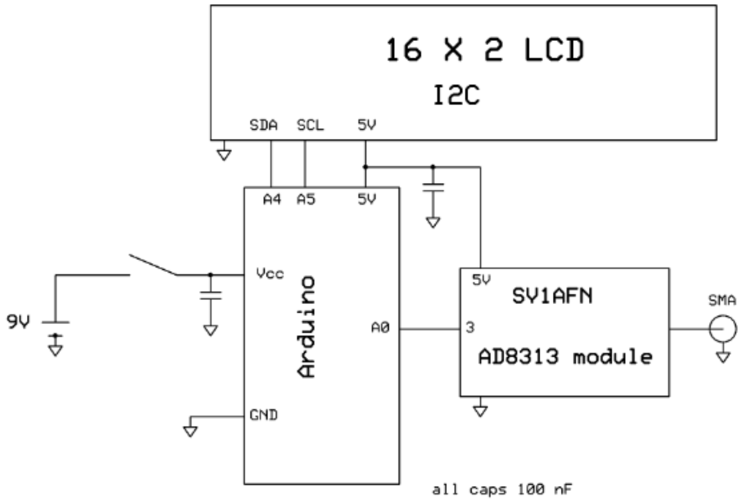

Figure 1. An Arduino reads the DC output voltage of the AD8313 logarithmic detector, then translates to dBm to be displayed on the serial LCD module.

I have also built this meter using lower cost AD8313 modules available on Ebay, but SV1AFN's high quality module is more stable. The dynamic range of this meter is from -60 dBm up to about 10 dBm. The modules report out a voltage of about 0.5 volts at -60 dBm, increasing linearly (with dBm) to 2.5 volts at 10 dBm. The Arduino does the A-to-D conversion, translates voltage to dBm, then displays the result through serial I2C communication on a I2C 16X2 LCD.

[1] Andy Nehan, G4HUE, "RF meter for low power using an Arduino and LCD," RSGB RadCom, December 2018, pp. 75—77.

Construction

This is a very low-effort project when you use a transparent acrylic enclosure (from say *The Container Store)* since you merely glue the LCD panel to the inside front wall with four tiny dabs of Loctite super glue. Drilling the acrylic for the SMA and on-off switch is a matter of seconds. The 9-V battery and Arduino Nano are secured with hook-and-loop strips, and all connections are made with 4-inch breadboard jumper wires. See Figures 2 and 3.

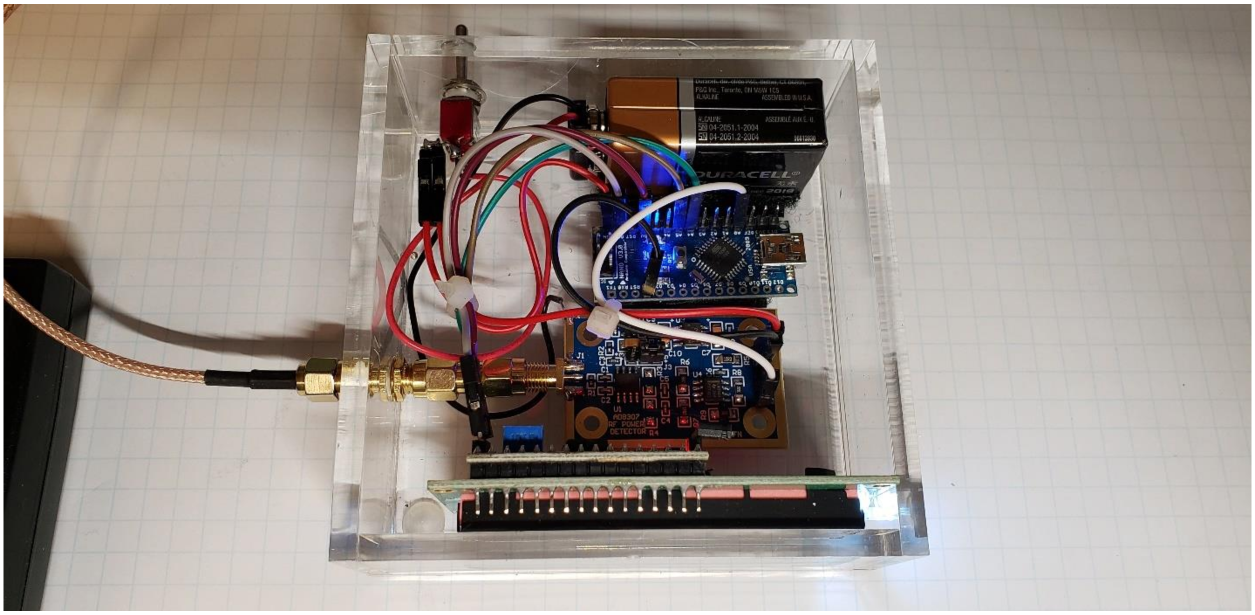

Figure 2. The LCD is spot glued to the inside front panel of the transparent acrylic enclosure while the Arduino and battery are secured with hook-and-loop strips. Wiring is quickly done by pushing on breadboard jumpers.

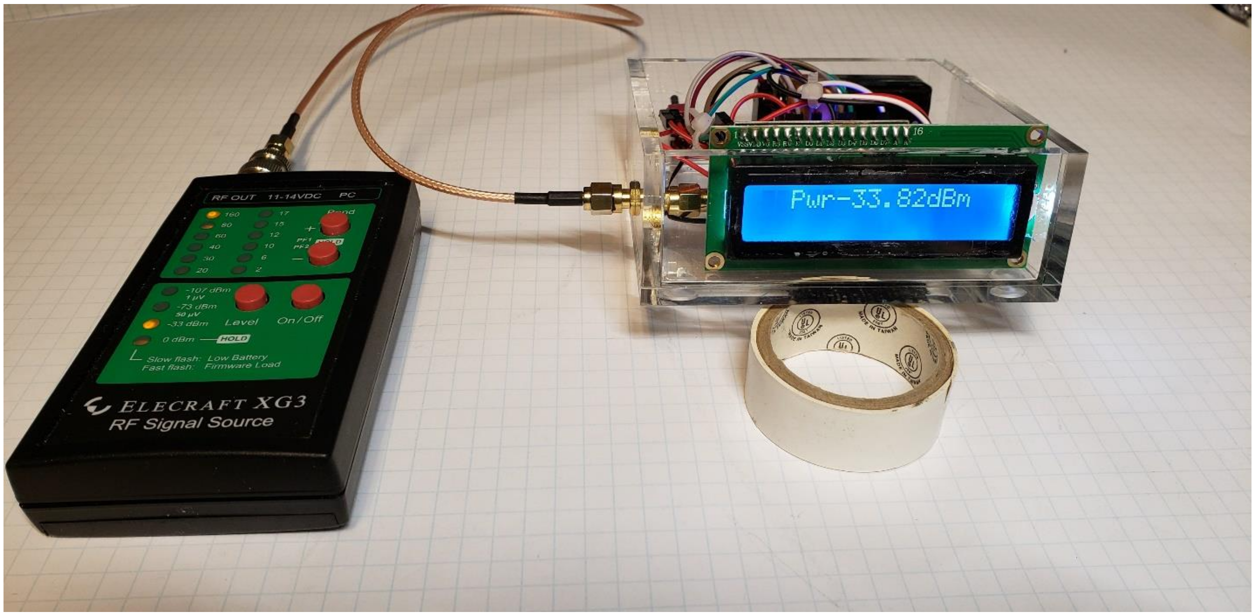

Figure 3. The power meter in operation.

Arduino sketch

I will be delighted to supply the code for this power meter in two flavors: The first is a one-pass A-to-D capture of the AD8313 output, the second employs averaging. The averaging seems necessary for the low-cost Ebay AD8313 modules, while the single pass appears sufficient for the SV1AFN module. Request by email at w8mqw@arrl.net.

You will have to calibrate the meter via the software for your particular module. You can experimentally tweak the values of the function *scaledb* found in the last two lines of the sketch until you are within acceptable accuracy. Or, more systematically, measure the output DC voltage of the AD8313 module at a high known power level (like 0 dBm), insert attenuation, then measure output voltage again at a low level (say -40 dBm), then apply the point slope formula for a line to obtain the exact slope and intercept for the translation from voltage to dBm.

Several lines of the sketch are devoted to aural readout, which you may ignore or delete. I needed the audio feedback once while adjusting an EME dish feed for minimal return loss and illumination. The audio quality of the Arduino’s pulse-width modulation scheme is ghastly.