This session explores equipment we can build to make RF measurements. It is based on the article by Ken Pollock WB3JOB titled "RF Measurements Using Homemade Equipment" which he sent to me via email.

## Contents



## BILL OF MATERIALS:

- HiLetgo ESP-WROOM-32 ESP32 ESP-32S Development Board, available at Amazon for $11. You can use a different microcontroller but you will need to adjust your program and wiring.

<< Upload button

<< Reset button

- Computer to host software and power the ESP32
  - Windows, Mac, Linux, including Raspberry Pi, are supported
- USB cable to connect them together: MicroUSB to USB-A
- Optional: 10 uF electrolytic capacitor
- From Amazon for $11: DEYUE 3 Set Standard Jumper Wires Plus 3 Set of Solderless Prototype Breadboard 830 tie Points Breadboard | 3 Set of M/F, M/M, F/F - Each 40pin Electronic Jumpers Wire

https://www.amazon.com/Standard-Jumper-Solderless-Prototype-Breadboard/dp/B07H7V1X7Y/

- Breadboard full size
- Jumper wires
  - Male-male if all parts are on breadboard
  - Male-female if some parts are not on breadboard
  - Female-female if all parts are not on breadboard

- ALTERNATE for neater breadboards: Use Breadboard Jumper Wires, such as, https://www.amazon.com/BOJACK-Solderless-Flexible-Breadboard-Connecting/dp/B08YRGVYPV/ref=sr_1_6 these are a little more expensive as there are more wires needed but the resulting breadboard is much neater. You will want to get the above kit as well since the breadboards are still needed and the jumper wires are essentially no additional cost.
- From Amazon for $17: Display 320x240 pixel TFT on a carrier board using the ILI9341 driver and an SPI interface. I used "HiLetgo ILI9341 2.8" SPI TFT LCD Display Touch Panel 240X320 with PCB 5V/3.3V STM32" https://www.amazon.com/HiLetgo-240X320-Resolution-Display-ILI9341/dp/B073R7BH1B/ Also available from sources like EBay and Banggood.
- The following parts for Noakes Field Strength Meter can be sourced from Jameco.com whose part numbers are shown here.
  - 2 – 1N34a germanium diodes (2220533)
    - This is the most critical part and is essential
  - 4 - .01 uF disk capacitors
  - 1 – 68K resistor
  - 1 – 100 uH inductor (2312479)
    - I had ordered some 1 mH inductors from Jameco (2339327) so I used one of them which seemed to work fine.
  - 1 – antenna jack, can be BNC or SMA or ? (BNC is 71590)
- You will use an antenna for the band you are trying to measure, with an associated plug. I used antennas from RTL-SDR multipurpose dipole antenna kit available from Amazon.com ( https://www.amazon.com/dp/B075445JDF/ref=as_li_ss_tl ) for $18.
- The following parts for Pollock digital measurement meter can also be sourced from Jameco.com
  - 3 – 1M ohms resistors
  - 1 – LM324 quad Op Amp (826671)
  - 1 – 10K variable resistor, linear taper (2272207)
  - 1 – 0.01 uF disk capacitor
- Some of the values are not critical. For example, I had 100K resistors, and 1M resistors, and they seemed to work fine.

## INTRODUCTION:

In his article, Ken Pollock, WB3JOB, describes his interest in various RF measurements. His paper is attached to this document. Although I am sure that his equipment works for him, I didn't really understand the added value of all of his probes. Some of his devices can be best replaced by an oscilloscope with good probes. The wavemeter is, according to Ken, useful in checking if a transmitter is producing harmonics. For that capability, you can use the TinySA or the Digilent AD2. I will focus on an RF Field Strength Meter.

## FIELD STRENGTH METER DESIGN:

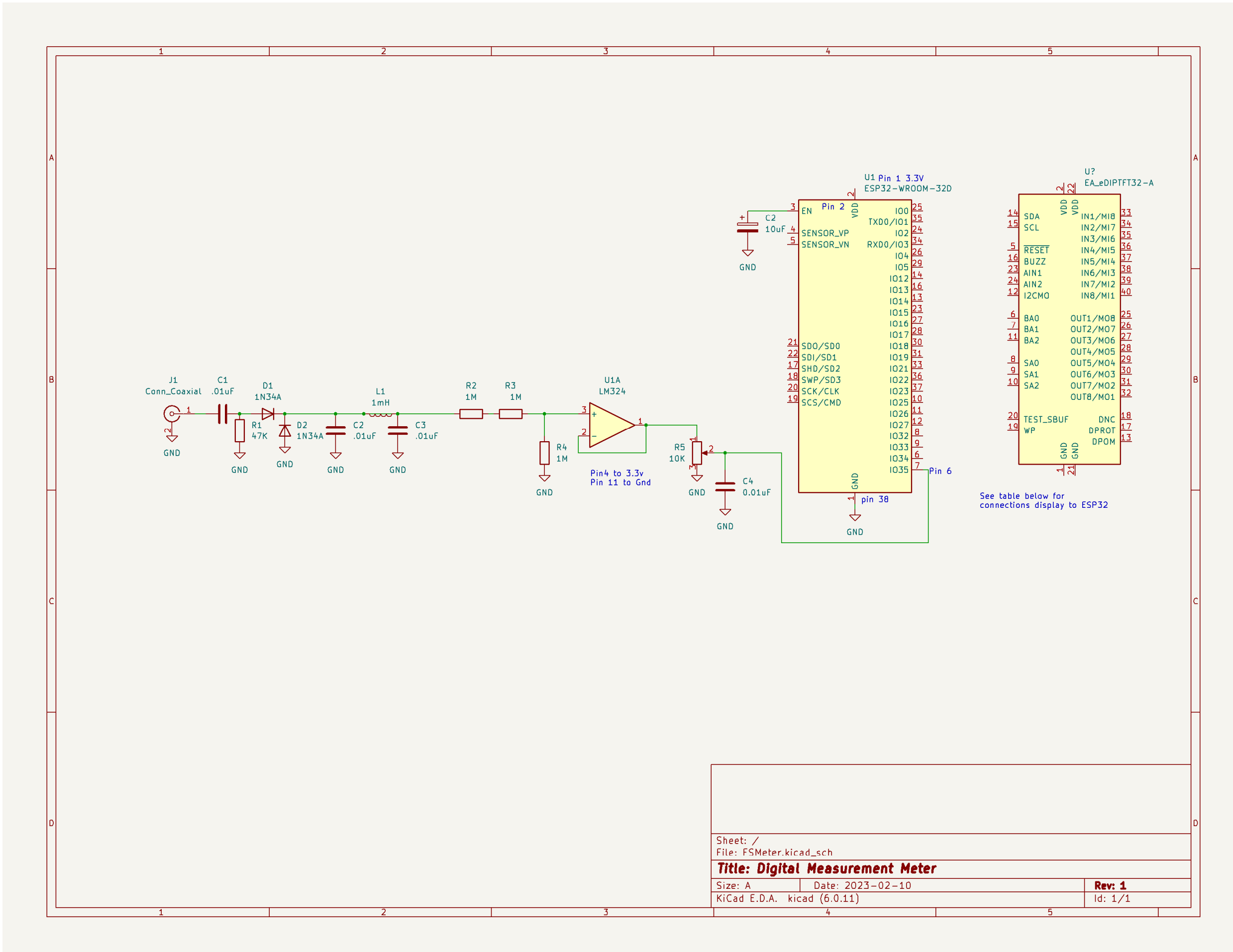
I examined the design from Ken's paper. I also got out my ARRL Handbook 2016 and read Chapter 25, Test Equipment and Measurements. Under Antenna Measurements, I read "A field strength meter is a device, generally with a built-in antenna, that picks up the radiated signal off the air and indicates the level on a meter or display. … They are useful for tuning an antenna, antenna tuner, or transmitter for maximum signal as well as for comparing different antennas. A field strength meter is a simple one-evening construction project." I think "simple one-evening construction project" is optimistic.

The chapter goes on to discuss some of the difficulties in getting consistent and useful accuracy.

A field strength meter's role is to provide a relative signal strength of a near field RF signal radiated from a transmitting antenna. It will reveal how well or poorly your antenna is radiating. If you measure the relative value over time, you can get an idea how well your antenna is radiating, or not radiating.

We will adapt the digital meter that Ken Pollock designed as our display device in keeping with the workshop use of Arduino microcontrollers. I would note that using an analog meter would make the instrument battery free.

Here is the schematic diagram for my RF Field Strength Meter.




| Display Pin | Display Label | Connects To ESP32: | Function |
|---|---|---|---|
| 1 | Vcc | "5V" | Power |
| 2 | Gnd | "Gnd" | Ground |
| 3 | CS | GPIO 5 | Chip Select |
| 4 | RST | "3.3V" | Display Reset |
| 5 | DC | GPIO 21 | Data/Cmd Line |
| 6 | MOSI | GPIO 23 | SPI Data |
| 7 | SCK | GPIO 18 | SPI Clock |
| 8 | LED | "5V" | LED Backlight Power |
| 9 | MISO | n/c | SPI Data |

D1 and D2 act as both rectifier and voltage doubler. It uses the 1N34A which is a germanium diode because it has a low forward voltage and fast switching speed.

The next section of the circuit looks to me like a low pass filter. Why is it here? I don't know but I tried to eliminate it and the circuit did not work as it produced no output.

Ken's design uses a 16x2 LCD display which is easy to program, but we already are familiar with another display. I modified the circuit and his code to use the 320x240 pixel TFT from our clock demonstration.

The circuit uses an Op Amp, or Operational Amplifier. The name comes from its original use as an amplifier in an analog computer to perform mathematical operations. The Op Amp is quite versatile and is a good device to get to know. I studied it in the ARRL Handbook. Op Amps today typically come in packages that include two or four Op Amps in one integrated circuit package.

One disadvantage of the Op Amp is that if you have a signal that swings both positive and negative, then you must supply both positive and negative voltages to it. Ken solves this problem by using one section of the Op Amp as a way to generate these voltages from a 9 volt battery. I found that I could use a single positive voltage from the ESP32 which provided an output and I knew I would not be overloading the analog input in the ESP32.

In this case, the circuit uses the Op Amp as a voltage follower, or unity gain amplifier. It provides a high impedance input and a low impedance output. There are lots of other uses for an Op Amp, which could be material for a future workshop.

The output of the Op Amp feeds to an analog input of the ESP32 which can then display the voltage. The analog input is connected to an ADC or Analog to Digital Converter inside the ESP32.

You can use a different probe as the input to the display unit if you want to measure RF voltage in a circuit, so this work has other uses on your bench.

I built this circuit on a breadboard and connected it to the clock circuit from last session, and loaded a different sketch onto the ESP32. See “sketch_RFMeter.ino” below.

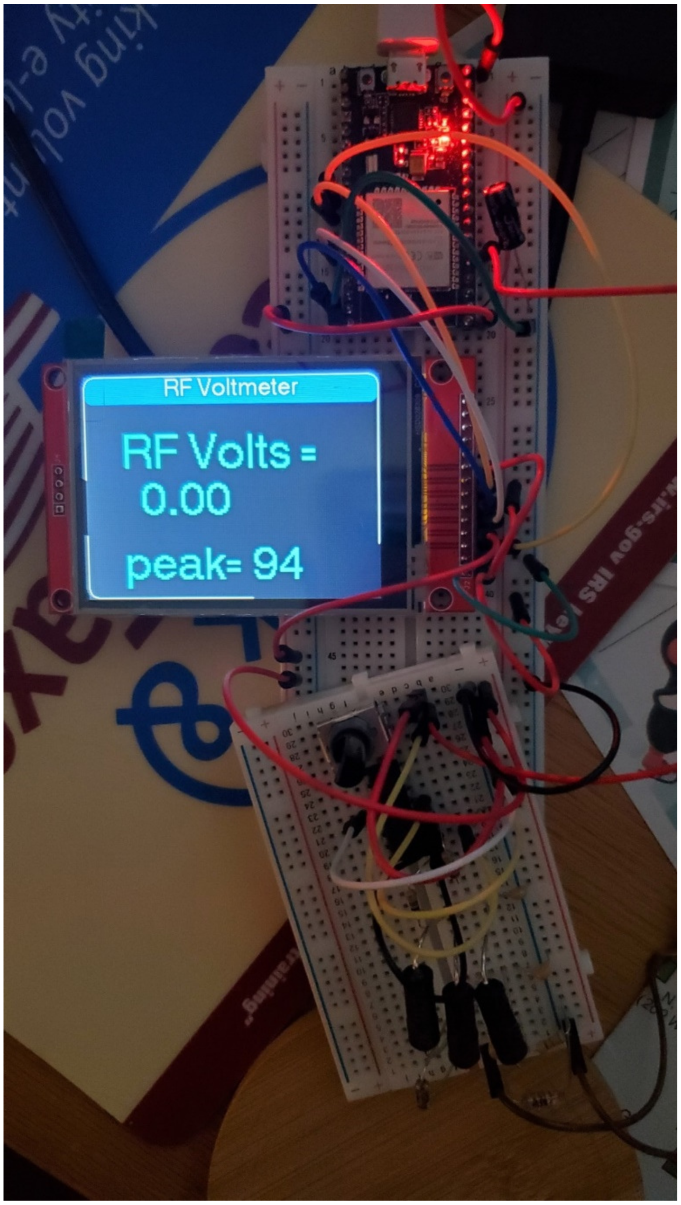
This schematic drawing is for the Hiletgo ESP32 Dev board. If you use a different type, the pins may be different.

I used the potentiometer to adjust the display value.

I tested this circuit using my FT1-XDR HT on 2 Meters at varying distances and power levels, and using my IC-7100 on 40 Meters running JS8CALL and detecting the strength of the heartbeat transmission. The HF antenna is a dipole about 30 feet from where I was measuring.

The display shows a value which indicated relative strength of the RF signal received at the antenna. It also shows the peak value received during the last 10 seconds.

Here is a photo of the breadboard version of the project.




## Sketch_RFMeter.ino

```
/* 20 volt RMS RF Voltmeter using ESP32 and TFT LCD display
based on WB3JOB article in QST 2/2023
modified to work with TFT display
AA6BD
*/
#include <TFT_eSPI.h> // Hardware-specific library
#include <SPI.h>
```

```
#define LABEL_FGCOLOR   TFT_YELLOW                 // color of label text
#define LABEL_BGCOLOR     TFT_BLUE                 // color of label background
#define TITLE             "RF Voltmeter"

int analogPin = 35;                 // Analog input pin
float voltInd = 0.0;                // display voltage
float voltPeak = 0.0;               // peak display voltage
int delayPeak = 0;                  // delay time since last zero reading
int delayReset = 20;                // when to reset delay timer
float vIn = 0.0;                    // Variable when analog inut is read
float vScale = 0.1;                 // scale factor for voltage in
float vLast = 0.0;                  // last value of voltInd
float peakLast = 0.0;               // last value of voltPeak
TFT_eSPI tft = TFT_eSPI();          // Invoke custom library
const int f = 7;                    // screen font

void setup() {
  // put your setup code here, to run once:
  //Serial.begin(115200);
  tft.init();
  tft.setRotation(1);
  tft.fillScreen(TFT_BLACK);
  tft.fillRoundRect(0,0,319,30,10,LABEL_BGCOLOR);  // title bar
  tft.drawRoundRect(0,0,319,239,10,TFT_WHITE);     // draw edge screen
  tft.setTextColor(LABEL_FGCOLOR,LABEL_BGCOLOR);   // set label colors
  tft.drawCentreString(TITLE,160,2,4);             // show sketch title on screen
  tft.setTextColor(TFT_CYAN,TFT_BLACK);      // set to cyan
  tft.setTextSize(2);
  pinMode(analogPin,INPUT);          // Set pin as analog input
}

void loop() {
  // put your main code here, to run repeatedly:
  voltInd = analogRead(analogPin)*vScale; // read digital representation of
voltage
  tft.drawString("RF Volts = ", 40, 60, 4);
  if (voltInd != vLast) {                     //  if voltInd changed
    vLast = voltInd;
    tft.fillRect(0,110,240,60,TFT_BLACK);  // clear field
  }
  tft.drawFloat(voltInd, 2, 60, 110, 4);
  if (voltPeak < voltInd) {                     // save if new peak
    voltPeak = voltInd;
  }
  tft.drawString("peak=", 40, 180, 4);
```

```
  if (voltPeak > peakLast) {                 // if voltPeak changed
    peakLast = voltPeak;
    tft.fillRect(180,180,240,60,TFT_BLACK);   // clear field
  }
  tft.drawFloat(voltPeak, 0, 180, 180, 4);
  if (delayPeak < delayReset) {
    delayPeak++;                             // increment ow long peak has been
displayed
    if (delayPeak >= delayReset) {           // if peak displayed long enough
      delayPeak = 0;                         // reset
      voltPeak = 0.0;
      peakLast = 0.0;
      tft.fillRect(180,180,240,60,TFT_BLACK);   // clear field
    }
  }
  delay(500);
}
```

## SOFTWARE IMPROVEMENTS

I began thinking about “sketch_RFMeter.ino” and have made some improvements. The original version uses straight TFT objects and you may notice some flickering, so I found that there is a better way to program this, using “sprites.” A sprite is a video object that can be sent to the screen as one object, so there is no flickering.

I learned that a sprite has some limitations. In particular, it cannot be the full size of the screen, 320 x 240 pixels, due to memory limits in the ESP32 microprocessor. I ended up with a sprite size of 316 x 190.

To see how to construct any function using the TFT_eSPI library, go to the Arduino > libraries > TFT_eSPI folder and examine the file: TFT_eSPI.h

First, I modified the original sketch to use a sprite:

```
/* sketch_RFMeter_sprite.ino
20 volt RMS RF Voltmeter using ESP32 and TFT LCD display
based on WB3JOB article in QST 2/2023
modified to work with TFT display
and modified to use sprite
AA6BD
*/
#include <TFT_eSPI.h>                  // graphic library
#include <SPI.h>

#define LABEL_FGCOLOR   TFT_YELLOW                // color of label text
```

```
#define LABEL_BGCOLOR     TFT_BLUE              // color of label background
#define TITLE             "RF Voltmeter"

int analogPin = 35;              // Analog input pin
float voltInd = 0.0;             // display voltage
float voltPeak = 0.0;            // peak display voltage
int delayPeak = 0;               // delay time since last zero reading
int delayReset = 20;             // when to reset delay timer
float vIn = 0.0;                 // Variable when analog inut is read
float vScale = 0.1;              // scale factor for voltage in
float vLast = 0.0;               // last value of voltInd
float peakLast = 0.0;            // last value of voltPeak

TFT_eSPI tft = TFT_eSPI();       // Invoke custom library
TFT_eSprite sprite= TFT_eSprite(&tft);  // also add sprite functions

void setup() {
  // put your setup code here, to run once:
  //Serial.begin(115200);
  tft.init();
  tft.setRotation(1);
  tft.fillScreen(TFT_BLACK);
  tft.fillRoundRect(0,0,319,30,10,LABEL_BGCOLOR);  // title bar
  tft.drawRoundRect(0,0,319,239,10,TFT_WHITE);     // draw edge screen
  tft.setTextColor(LABEL_FGCOLOR,LABEL_BGCOLOR);   // set label colors
  tft.drawCentreString(TITLE,160,2,4);             // show sketch title on screen

  sprite.createSprite(316,190);                    //size is limited by ESP32
memory
  pinMode(analogPin,INPUT);         // Set pin as analog input
}

void loop() {
  // put your main code here, to run repeatedly:
  sprite.setTextDatum(0);
  sprite.setTextColor(TFT_CYAN,TFT_BLACK);      // set to cyan
  sprite.setTextSize(2);
  voltInd = analogRead(analogPin)*vScale; // read digital representation of
voltage
  sprite.drawString("RF 'Volts' = ", 40, 20, 4);
  if (voltInd != vLast) {                     //  if voltInd changed
    vLast = voltInd;
  }
  if (voltPeak > peakLast) {                    // if voltPeak changed
    peakLast = voltPeak;
```

```
  }
  sprite.setTextSize(1);
  sprite.fillRect(0,70,316,60,TFT_BLACK);   // clear voltInd field
  sprite.drawString(String(voltInd), 60, 70, 7);
  if (voltPeak < voltInd) {                  // save if new peak
    voltPeak = voltInd;
  }
  sprite.setTextSize(2);
  sprite.drawString("peak=", 40, 130, 4);
  sprite.setTextSize(1);
  sprite.fillRect(180,120,180,80,TFT_BLACK);  // clear voltPeak field
  sprite.drawString(String(voltPeak), 180, 130, 7);

  if (delayPeak < delayReset) {
    delayPeak++;                             // increment how long peak has been
displayed
    if (delayPeak >= delayReset) {           // if peak displayed long enough
      delayPeak = 0;                         // reset
      voltPeak = 0.0;
      peakLast = 0.0;
    }

  }
  sprite.pushSprite(1,35);
  delay(200);
}
```

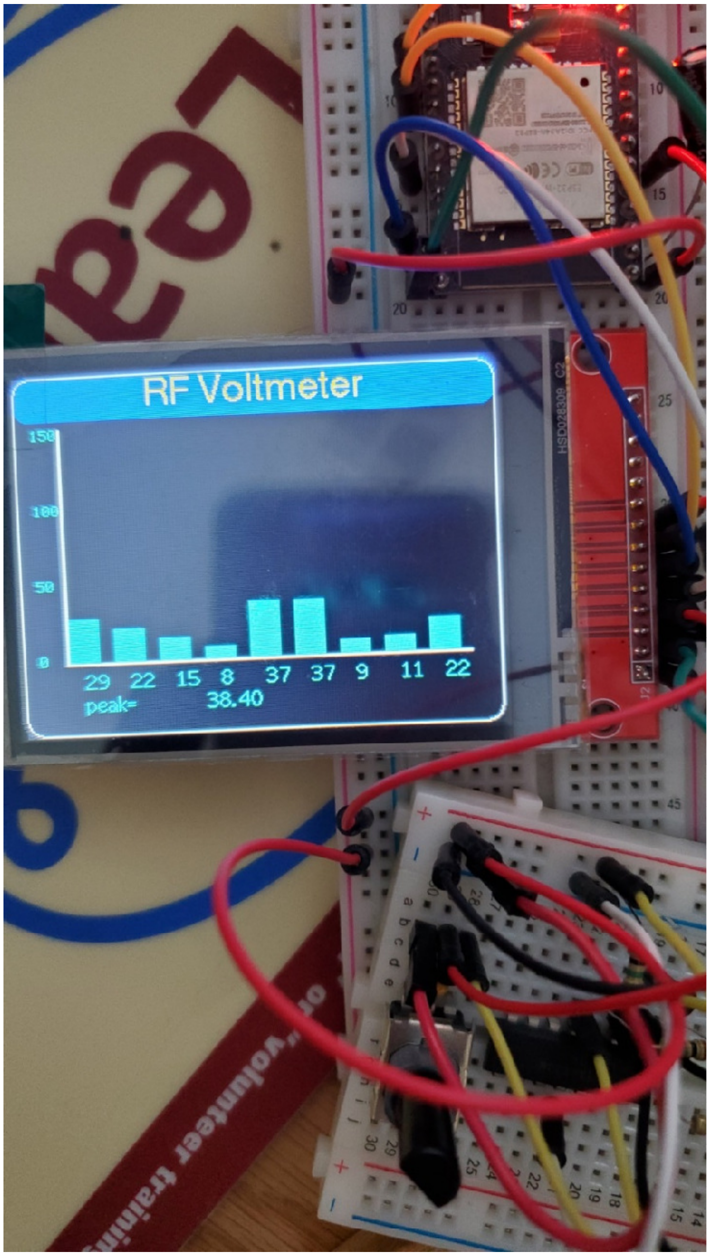
The next version of the software displays the data as a scrolling bargraph so you can get a feeling for how the RF voltage varies over time.

```
/* sketch_RFMeter_bargraph.ino
20 volt RMS RF Voltmeter using ESP32 and TFT LCD display
based on WB3JOB article in QST 2/2023
modified to work with TFT display using a sprite
and modified to show a bargraph
AA6BD
*/
#include <TFT_eSPI.h>                       // graphic library
#include <SPI.h>

#define LABEL_FGCOLOR   TFT_YELLOW                // color of label text
#define LABEL_BGCOLOR     TFT_BLUE                // color of label background
#define TITLE             "RF Voltmeter"
```

```
int analogPin = 35;             // Analog input pin
float voltInd = 0.0;            // display voltage
float voltPeak = 0.0;           // peak display voltage
int delayPeak = 0;              // delay time since last zero reading
int delayReset = 30;            // when to reset delay timer
float vIn = 0.0;                // Variable when analog inut is read
float vScale = 0.1;             // scale factor for voltage in
float vLast = 0.0;              // last value of voltInd
float peakLast = 0.0;           // last value of voltPeak
int volts[10] = {0};            // last nine volt values for bargraph array

TFT_eSPI tft = TFT_eSPI();      // Invoke custom library
TFT_eSprite sprite= TFT_eSprite(&tft);  // also add sprite functions

void setup() {
  tft.init();
  tft.setRotation(1);
  tft.fillScreen(TFT_BLACK);                        // fill screen with black
  tft.fillRoundRect(0,0,319,30,10,LABEL_BGCOLOR);   // title bar
  tft.drawRoundRect(0,0,319,239,10,TFT_WHITE);      // draw edge of screen
  tft.setTextColor(LABEL_FGCOLOR,LABEL_BGCOLOR);    // set label colors
  tft.drawCentreString(TITLE,160,2,4);              // show sketch title on screen

  sprite.createSprite(316,190);                     //size is limited by ESP32
memory
  pinMode(analogPin,INPUT);                         // Set pin as analog input
}

void loop() {
  sprite.fillSprite(TFT_BLACK);                  // fill sprite with black
  sprite.fillRect(28,0,2,155,TFT_WHITE);         // vertical line
  sprite.fillRect(28,155,270,2,TFT_WHITE);       // horizontal line
  sprite.setTextDatum(0);
  sprite.setTextColor(TFT_CYAN,TFT_BLACK);       // set to cyan
  sprite.setTextSize(1);
  for (int i=0; i<4; i++)
    sprite.drawString(String(i*50), 10, 150-(i*50));  // draw vertical values
  voltInd = analogRead(analogPin)*vScale;        // read digital representation of
voltage
  if (voltPeak < voltInd) {                      // save if new peak
    voltPeak = voltInd;
  }
    volts[0] = voltInd;                          // save current value in array[0]
  for (int i=0; i<9; i++) {
    volts[9-i] = volts[8-i];                     // shift volts over one cell
```

```
    sprite.fillRect(30+30*(i), 155-volts[i], 20, volts[i], TFT_CYAN);  // show 
each value in array as bar
    sprite.drawString(String(volts[i]), 10+30*(i+1), 160, 2);  // show each value 
below bar
  }
  sprite.drawString("peak=", 40, 175, 2);
  sprite.fillRect(120,175,120,20,TFT_BLACK);    // clear voltPeak field
  sprite.drawString(String(voltPeak), 120, 175, 2);  // show peak value

  if (delayPeak < delayReset) {
    delayPeak++;                              // increment how long peak has 
been displayed
    if (delayPeak >= delayReset) {            // if peak displayed long enough
      delayPeak = 0;                          // reset
      voltPeak = 0.0;
      peakLast = 0.0;
    }

  }
  sprite.pushSprite(1,35);                    // display sprite on screen
  delay(200);                                 // wait to get next value
}
```

Here is a screenshot of how it looks.  The bars will scroll to the right when the sketch is running.




## CQ ARTICLE

I just received the February, 2023, issue of CQ magazine. On page 56, Irwin Math, WA2NDM, shows simple RF circuits for a field strength meter and an RF power meter. His circuit for the field strength meter is very similar to the one shown above, which corroborates our approach. He notes that the component values are not critical, so you should feel comfortable using whatever components you have on hand.

He says that you don't need to intentionally calibrate it but to use it to adjust your transmitter and antenna system for maximum power output. If you have a directional antenna, you can use it to maximize power in the direction you wish. You can also use the field strength while walking around the horizontal antenna and record readings at various points to get a sense of the antenna's directional pattern.

He suggests that the circuit is most effective if contained in a metal box with just the antenna outside the box.

## CALIBRATION

You can calibrate your field strength meter by using a radio with known transmitter power and a tape measure. First, measure your transmitter power using a power meter connected directly to the transmitter. Then set your transmitter a distance from your field strength meter. I suggest a distance that is safe for your meter such as 3 feet for a 5 watt HT. Use the potentiometer on the field strength meter to calibrate it to your selected value, such as 100. It should give a reasonable relative strength after this.

Another way to calibrate would be to use a TinySA Spectrum Analyzer with its antenna connected. Use the TinySA right next to our RF Field Strength Meter, and use the result from the TinySA as calibration for the field strength meter.

## FUTURE EXPERIMENTS:

Some items I ran into while working on this project can be improved and investigated.

- I used the potentiometer and sketch value to set the scale for the display value. This might be calibrated to show actual field strength values.
- The circuit could be checked with an oscilloscope to see what the signals are like and to determine where it might be improved.
- The Op Amp design uses only one voltage while it often has both a plus and minus supply. This might improve the performance.
- The circuit could be built on a perf board or a printed circuit board could be designed.
- There is an article in the 2011 ARRL Handbook, page 25.33, "A Compensated, Modular RF Voltmeter," that might be valuable to pursue for more detail.

## CONSTRUCTION NOTES:

1. I purchased some packs of components to provide values for workshops. These included kits from Sparkfun of resistors, capacitors, and semiconductors, and Adafruit Parts Pack for LEDs, resistors, capacitors, and more.
2. I always use a component tester to check individual resistors, capacitors, inductors, diodes, and transistors to verify their stated values before using them. On this exercise, I needed some 10nF capacitors. Four were within tolerance, and one was half of the

value it should have been. My component tester was a kit from Banggood.com which is no longer offered but many are available from Amazon as well for $10-$15.

3. If you try a sketch and it doesn't seem to work, note that not all GPIO pins are available to use for any purpose. Try a different set of pins.

## KEN POLLOCK WB3JOB'S ARTICLE:

### RF Measurements using Homemade Equipment

**Introduction;** There are many times that I have a need to do various RF voltage measurements. Although I have different meters to aid in experiments and repairs, there is a "gap" in what I can do. I do have a meter that can measure up to 3 Volts, but, I needed something to measure higher voltages. Although I can purchase the needed equipment, why not build it? I have been building various devices since my younger years and so it was easy to decide.

I made a list of needed and desired measurements. I soon realized that most of the needs could be built as probes for the required applications. The list included measurements such as RF voltages, field strength measurements, and a simple wave-meter. These can be constructed as probes and a "measurement head" could display the readings. I looked at my trusty DVM (a Fluke 177) but then I came to the conclusion that, on several occasions, I needed the DVM while doing another measurement simultaneously. That caused me to investigate my options and the following is the result.

**Probes;** There are four basic probes that I would design and construct; two RF detector probes, a field strength probe, and a wave-meter probe. The resulting designs are shown in Figure 1. Two RF detector probes are needed because they have different characteristics. The "Shunt Probe" uses a capacitor in series with the probe and will only respond to AC as the capacitor blocks DC. The "Series Detector" probe will pass DC and will be used for calibration. The wave-meter probe covers the lower HF frequencies. Finally, the field strength probe is useful for HF and VHF frequencies. What is common for these probes is that they all use a standard 1N34 germanium diode. This diode will detect RF at lower levels and has a lower voltage drop than silicon detectors.

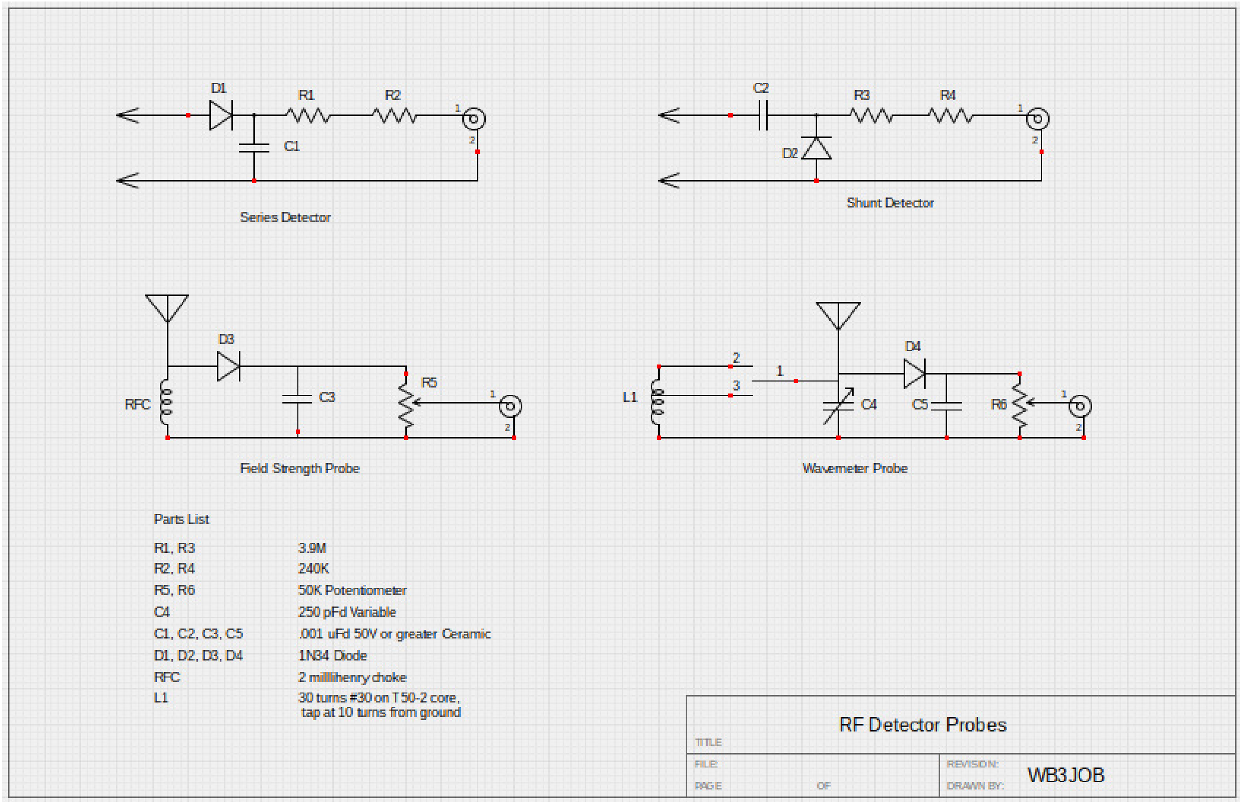
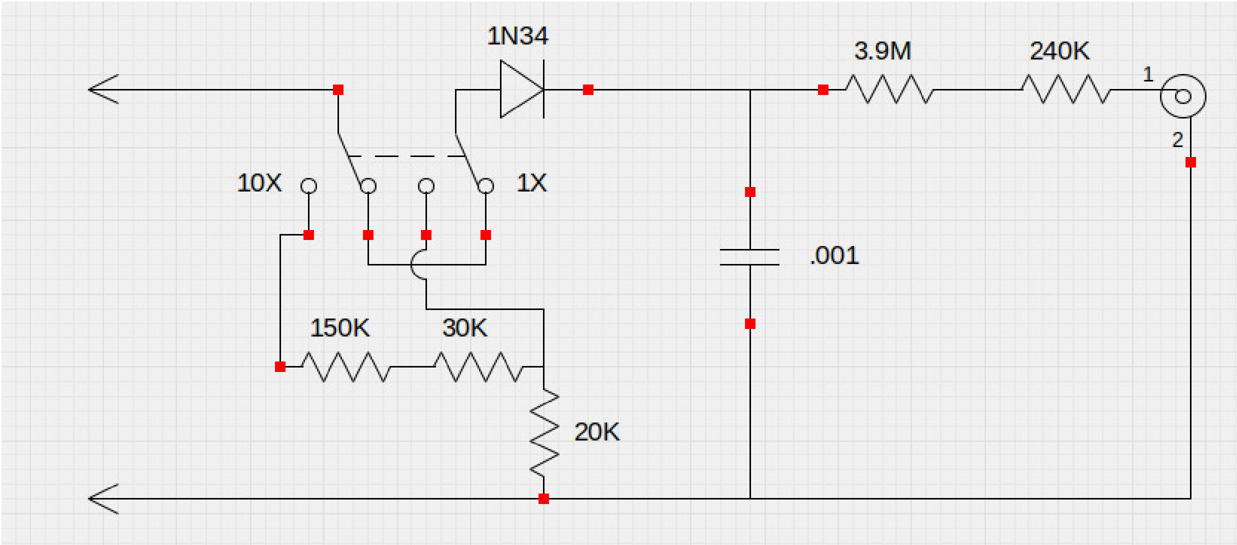



Figure 1 RF  Probes

Once the RF is detected, the signal is passed through series resistors to the measurement device. The values of these resistances are important. When an alternating current is detected, a DC voltage is produced that will be the peak voltage. This is 1.414 times the RMS voltage. The series resistances  will have a voltage drop equal to the the peak voltage minus the RMS voltage.

For example, if an RMS voltage of 10 Volts is measured, the peak voltage will be 14.14 volts. Selecting the proper resistance will result in having a 4.14V drop and 10 volts would be measured by the measuring instrument. This is a voltage divider with the probe being being the one resistor and the input resistance of the measurement device being the other resistance.

Soon after I started to use the system, the need arose to measure higher voltages. I then added another probe that was switchable for X1 or X10. The circuit is shown below;




Switchable 10X probe

This allows me to measure much higher voltages than 20 Volts. Up to 200 Volts may be measured due to the resistive divider. Use carbon composition or non-inductive resistors for the divider. I built the probe using point-to-point wiring on a small slide switch.

**Measurement Devices;** Out of curiosity, I measured the input resistance of a very inexpensive DVM that I had on hand and found it to be around 8 meg-ohms. The input resistance of the older VTVMs were around 11 meg-ohms and so I decided to build my own measurement “head”. It was to have an input resistance of 10 meg-ohms and be portable. Two different indicators were built. The first uses an analog meter for general measurements while the second uses a digital readout and is more accurate. The analog indicator is useful when adjusting a circuit as any meter movement is easy to see. Both are portable and use a standard 9V battery for power although a wall-wart may

also be employed.

The circuit for the analog DC meter is shown in Figure 2. The analog meter I used was surplus from a hamfest and is a 0-1ma meter. Conveniently, it had a scale that was already labeled as 0-15 volts. If another meter is substituted, it is totally possible to calibrate the meter up to 20V if desired. An LM324 quad op-amp is used because it is designed for operation at lower supply voltages, it has 4 independent op-amps, is inexpensive, and I had several in the junk box.

One section of the op-amp is configured a an inverting amplifier. The gain is set at 1/10 by the 10 meg-Ohm input resistor and the 1 meg-Ohm feedback resistor. A second op-amp is used as a divider for the power supply and generates an artificial ground. If the battery is 9V, the ground will be at ½ of this or 4.5 volts. This is necessary as as the input amplifier will invert the signal and the output swing will be a negative voltage, referenced to this ground. Construction was by point to point wiring using a piece of perfboard mounted on the back of the meter. Only DC signals are present and so any type of wiring could be used. The circuit is shown in Figure 2.

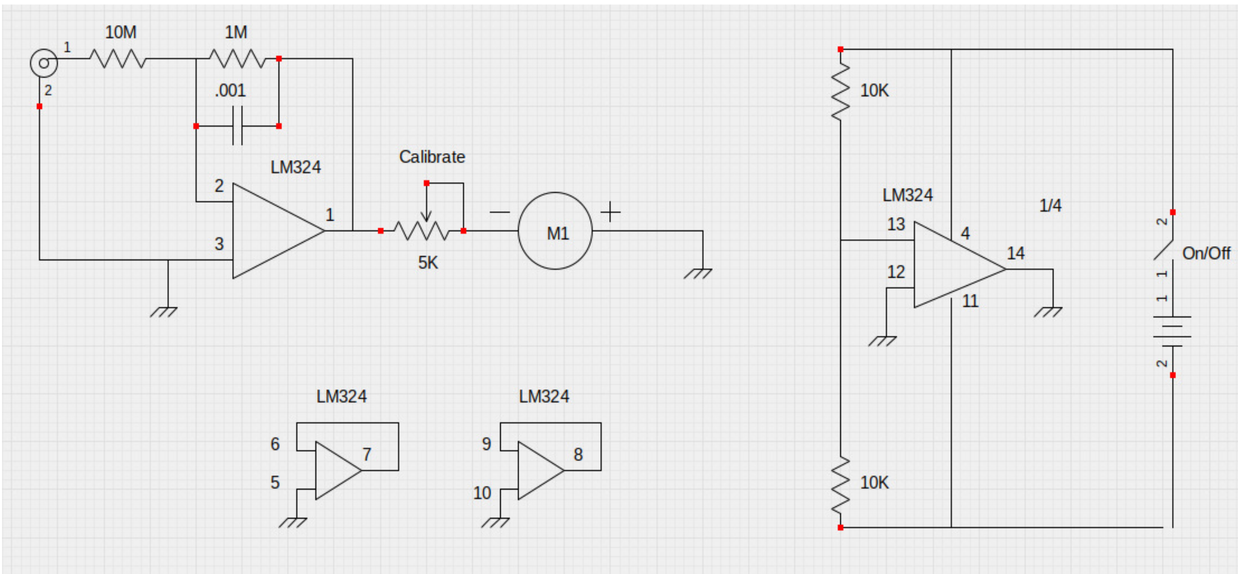



Figure 2. Analog Measurement Circuit

Soon after constructing the analog meter, it became desirable to increase the accuracy. I have been using the Arduino Nano micro-controllers for a number of years and so it seamed a natural as they are less than 5 dollars each. The display, the Nano, and the development board can be purchased for under $20 on-line. The Nano has several analog inputs that are 10 bit SAR type of converters. Even though the input resistance is several meg-Ohms, the impedance of the analog input has to be less than 10K. This is due to only one input is sampled at a time through an analog switch as used by the SAR. Measurements are in microseconds and the internal capacitances affect the measurement. The output impedance of an op-amp is low and can easily drive the micro-controller, therefore an LM324 was used. A .001uFd capacitor eliminates any RF.

A general purpose silicon rectifier is in series with the 9V battery negative terminal in order to obtain a small negative voltage for the op-amp. One section from the LM324 is used in this circuit as a voltage follower for the input divider. A voltage follower has a very high input resistance and doesn't load the divider. The output level of this stage is applied to a trimmer that is used for calibration. Note; It is important that the analog voltage applied to the Nano be between 0V and 5V as voltages outside of these limits can destroy the microprocessor. The circuit is shown in Figure 3.

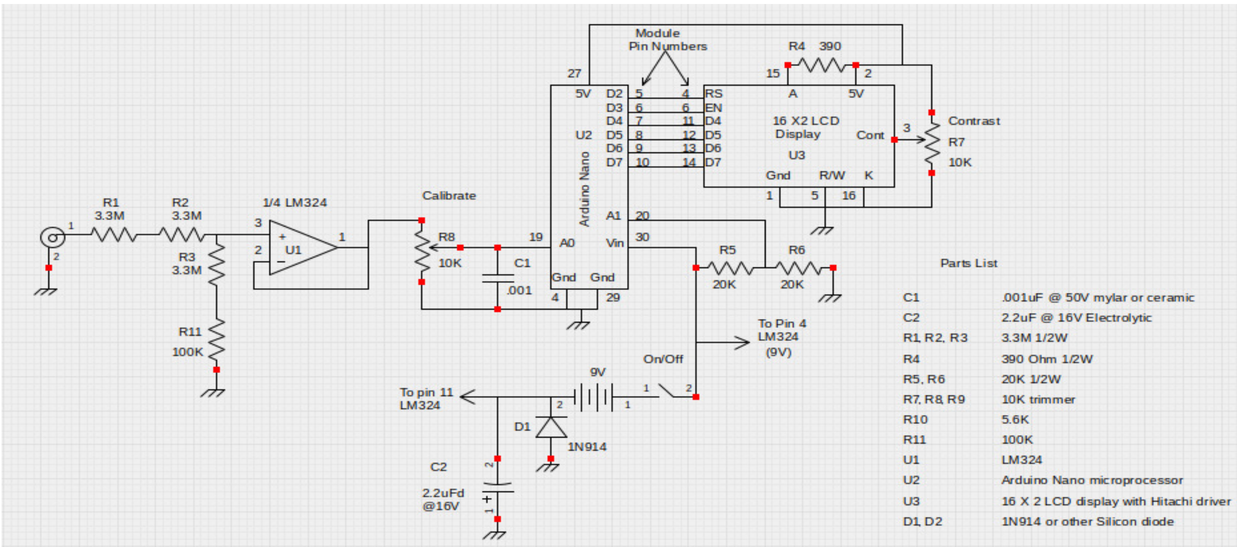



Figure 3. Digital Measurement Meter

**Construction;** The construction of the probes are not really critical at all. Standard components are used, the resistors are all either ¼ or 1/8 watt units. Point to point wiring on a small piece of perfboard cut to fit inside a tube with a 3/4 inch opening. The capacitors are either ceramic or Mylar and were selected to fit. To hold the board in place, a piece of heat shrinkable tubing was placed around the circuit board and then the board was inserted to the tube. The ends of the tubes were sealed using silicon. The wavemeter and the field strength probes were placed in small boxes.

The analog measuring meter was constructed on a piece of perfboard that was mounted on the back of the meter. The battery was secured in place with a couple pieces of Velcro that had a sticky back. All of the resisters were 1/4 watt and not critical at all.

The digital meter used a piece of perfboard that was large enough to hold the 14 pin IC op-amp and hold the LCD display as edge mounted. I like to use a small development board which allows me to remove and reinsert the Nano with ease. This board also has screw terminals on it so that wiring is simple. There is nothing critical about the wiring.

**Calibration;** The meters are easy to calibrate and only require a DC variable power supply and an accurate voltmeter.

To calibrate the analog meter, insert a new battery, adjust the “Calibrate” trimmer to the center of it’s range. Depending on the maximum reading desired, the “Calibrate” trimmer will be adjusted. If the desired maximum reading is 20 volts, adjust the external power supply for 28.28 volts (the peak of the RMS voltage of 20 volts times 1.414). Connect the Series Detector probe and connect to the variable power supply. Adjust the “Calibrate” trimmer for a meter reading of 20 volts. Since I calibrated my meter for 15 volts full scale, I set the power supply for 21.21 volts and performed the same procedure. The analog meter is now calibrated and ready for use.

The calibration of the digital meter is almost as simple. Turn on the meter. The display will light and you should see writing on the display and adjust the “Contrast Control” for the best viewing. Next, connect the adjustable power supply through the Series Probe and adjust the supply for a voltage of 21.21 volts. Watching the display, slowly adjust the “Calibrate” trimmer until a reading of “15.0” is displayed. Calibration is now complete. The calibration is good for both the Series and the Shunt probes. There are no calibration required for the wave-meter or the field strength meters as these are relative readings. Using a frequency generator, mark the dial indications for the wave-meter.

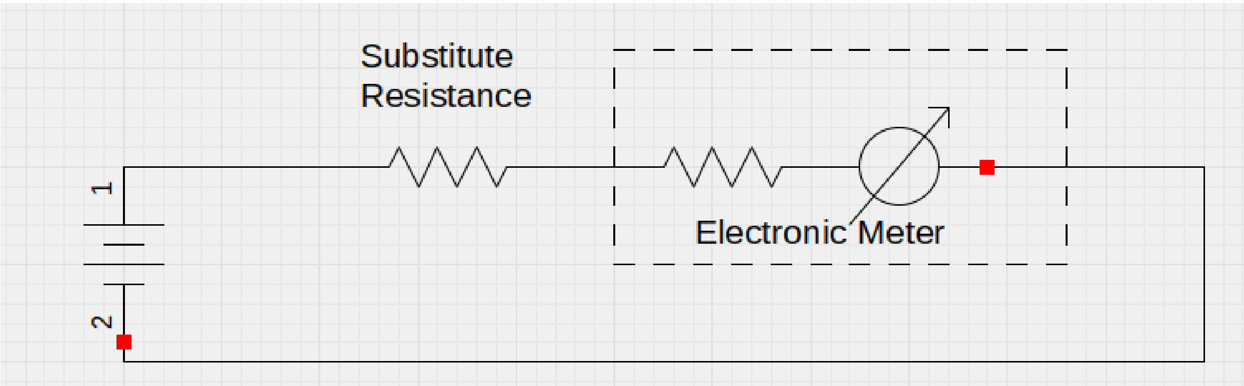
**Results;** The results have been most gratifying to say the least. Battery life is good as the analog meter, even at full scale, draws less than 5ma and the digital meter uses less than 15ma. A wall-wart power supply may be used to power the units if portability is not a concern. The meter is low loading and the field strength probe easily detects an HT located over 5 feet away. The probes can be used with any DVM or multimeter as long as the series resistance of the resistors in the probes are .414 times that of the meters’ internal resistance.

Another benefit of the indicators is that they may also be used for monitoring the station DC supply. If one desires to operate the digital meter from voltages higher than 9 Volts, a 9 Volt regulator should be used to supply the power to the Nano as the micro-controller does have a regulator but it can over-heat when used on voltages greater than 12 Volts.

I had a spare T-200 core on hand and made a simple probe consisting of 20 turns of #30 AWG wire and a detector of a series 1N34 and a .002 capacitor. I can easily slip this over a piece of coax to detect any currents flowing on the outside of the coax. This allows me to determine if a balun is working. The only limit is by ones imagination.

## Measuring The Input Resistance of a Meter

It is simple to determine the input resistance of a meter. All that is required is a voltage source, a couple clip-leads, and an assortment of resistors and possibly a potentiometer and the meter to be measured. First, put together the circuit below;




This is a simple series circuit with a voltage source, a series resistor and the meter to be measured. Start out by shorting out the series resistor and adjusting the supply for a known voltage such as 10 Volts. Next, using various resistors for the substitute resistance, find the combination of resistors that cause the meter to measure half scale or 5 Volts, if the supply is 10Volts. Disconnect the power supply and measure the value of the substitute resistance. The value obtained is the internal resistance of the meter. This is often referred to as the loading of the meter.

What you have done is to use a simple resistive voltage divider consisting of the meter and a series resistor. When a resistive divider divides the voltage in half, the two resistors are equal value. Measuring the resistance in series with the meter gives the resistance of the meter. The substitute resistance is normally called the multiplier resistance as it multiplies the voltage range of the meter to which it is used.

AC Voltages;

When an alternating voltage is to be measured, it is normally converted to a direct current and then displayed. The normal measurement of an AC voltage is displayed as the RMS value. This is the heating value of AC compared to DC. For example, if a voltage of 10 Volts is applied to a 10 Ohm resistance, one Ampere of current will flow and the power dissipated will be 10 Watts. This for DC or AC.

If the AC Voltage is rectified and filtered, the resulting voltage will be the peak voltage. This voltage is 1.414 times the RMS Voltage. The probes used compensate for this discrepancy by dropping the voltage by introducing a voltage drop of .414 times the applied voltage. For 10 Volts RMS , the peak voltage will be 14.14Volts. The resistances in the probe are calculated to drop 4.14 Volts and the meter will display the voltage as 10 Volts.

ABBREVIATIONS:

| | |
|---|---|
| ADC | Analog to Digital converter, takes as input a voltage and sends as output a digital value |
| APRS | Automatic Packet Reporting System |
| DMA | Direct memory access |
| EEPROM | Electrically erasable programmable read only memory |
| GPIO | General purpose input output |
| GPS | Global positioning system |
| IDE | Integrated development environment |
| I2C | Inter-integrated circuit |
| LED | Light emitting diode |
| NTP | Network time protocol |
| OS | Operating system |
| RPi | Raspberry Pi |
| SBC | Single Board Computer |
| SDR | Software defined radio |
| URL | Uniform resource locator |
| USB | Universal serial bus |
| VNC | Virtual network computing |