

A Small Portable SWR Indicator and Signal Source

This small battery powered microprocessor controlled device will help you rapidly tune your 630 – 10 meter antenna to resonance or serve as a 100 KHz – 112.5 MHz signal source.

Gene Marcus W3PM GM4YRE



Figure 1.
External view

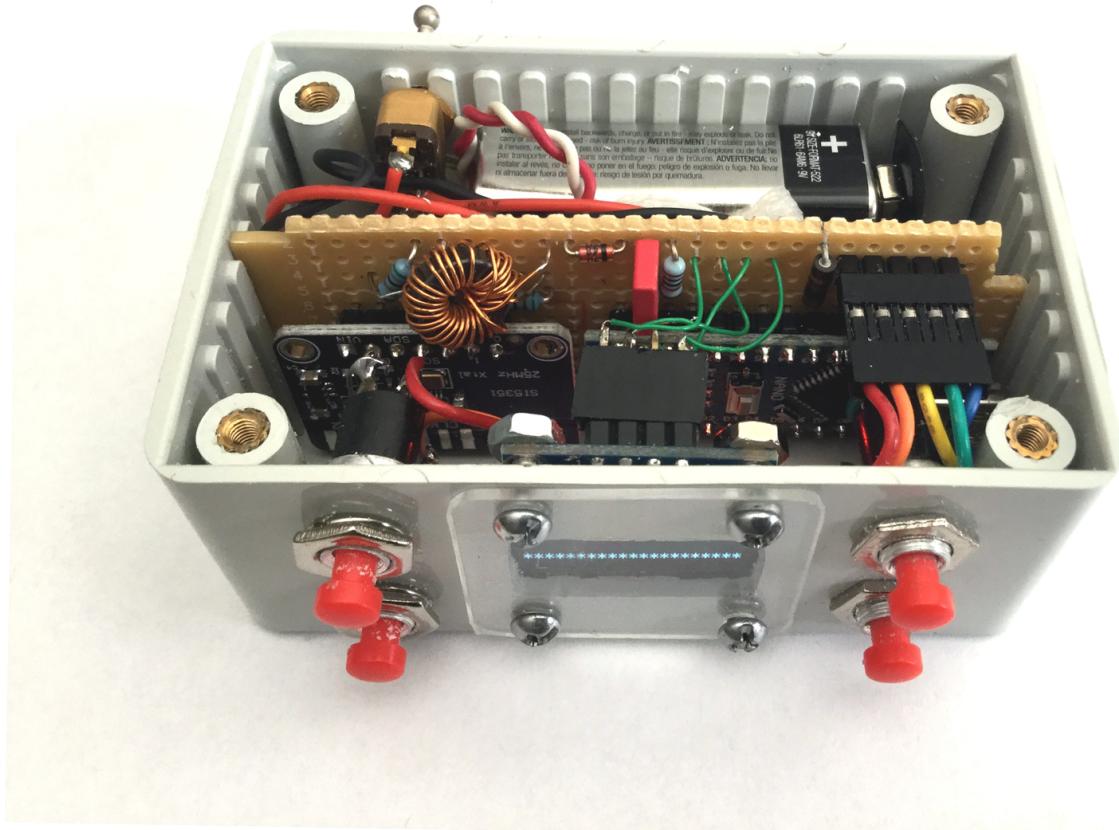


Figure 2.
Internal view

W3PM Antenna Tuning Indicator and Signal Source

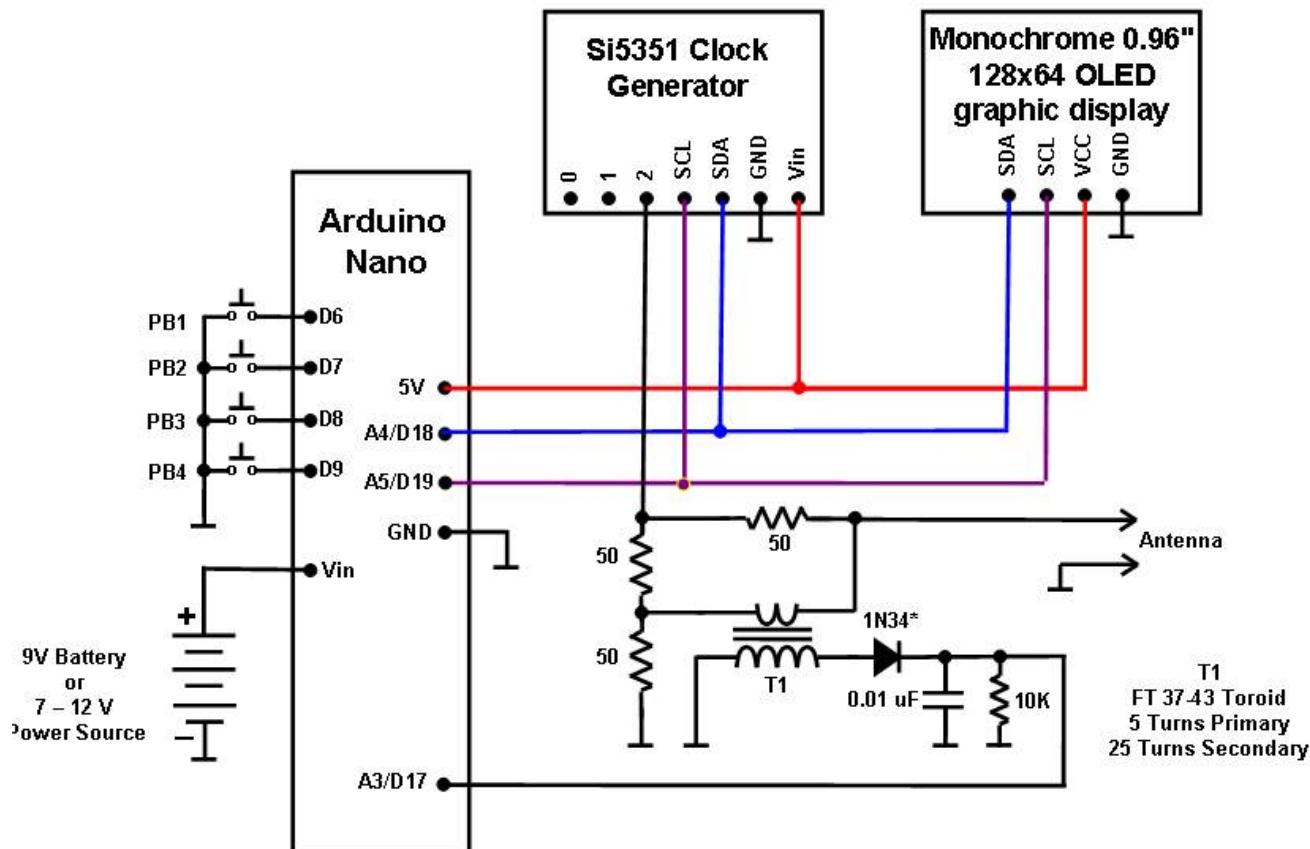


Figure 3.
Schematic

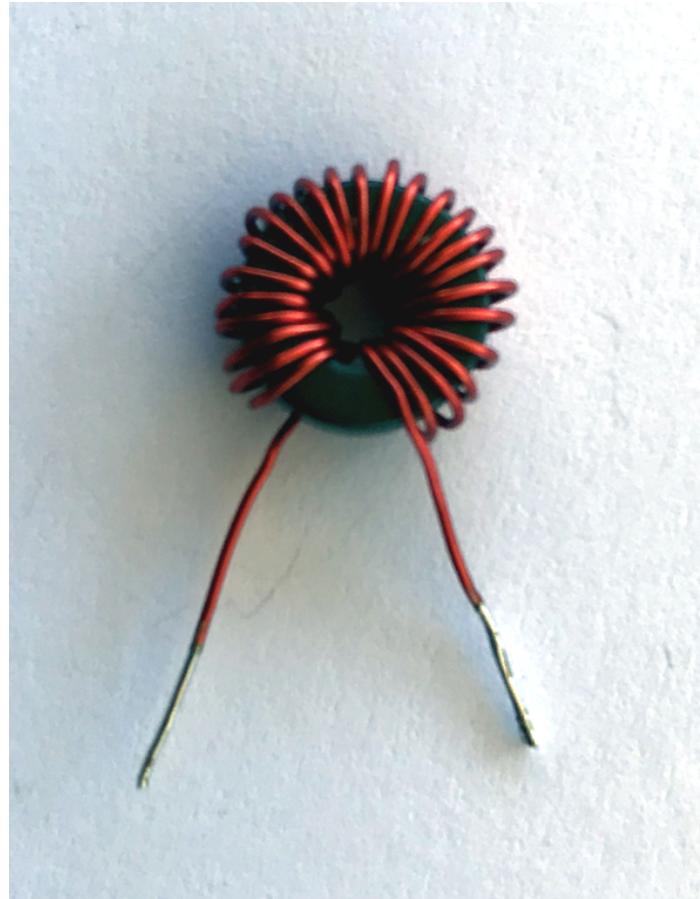


Figure 4.
Secondary 25 turn winding of T1

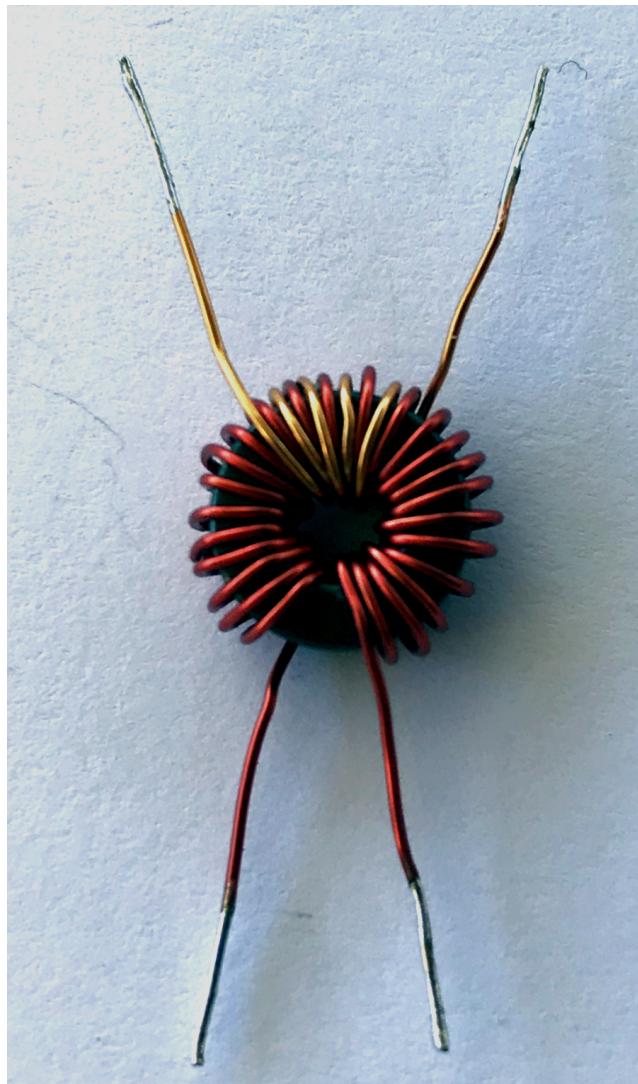


Figure 5.
Completed 5 turn primary and 25 turn secondary winding of T1

Introduction

I have been experimenting with homebrew magnetic loops for some time and wanted a quick and easy way to adjust them to resonance. Dan Tayloe, N7VE, developed a unique resistive SWR bridge that uses a simple LED to identify the dip in SWR. His SWR indicator was simple to build and worked very well. Unfortunately, low power from a transmitter was required to use the indicator. Sometimes accidentally applying too much power would destroy the load resistors within the device. I wanted something very simple that I could carry in my pocket to tune an antenna in the field. While experimenting with an Arduino Nano and a Si5351 RF clock generator for a different project, I realized that this low cost set of boards could be used as the RF source for a resistive SWR bridge. As an added bonus, the device could also be used as an accurate signal source.

This device will not tell you what the actual SWR is, but instead serves as an indicator to let you know when the antenna is tuned to a nominal 50 ohm impedance. Simply set the device to the frequency of interest, then adjust your antenna tuner or magnetic loop antenna for a dip in the display's bar graph and your antenna is tuned. Alternately, use the frequency up/down buttons to sweep the antenna and find its resonant frequency.

An approximate 1V p-p square wave into 50 ohms at the selected frequency is available at the output port at all times when the device is powered on. It is a stable and fairly accurate signal source that could be used for a number of purposes limited only by the builder's imagination. A simple VFO or LO readily comes to mind. If greater frequency accuracy is required, a calibration procedure is outlined below.

The Components

Arduino Nano (or Uno)

The Arduino Nano, as the name suggests is a compact, complete and user friendly microcontroller board. These boards are widely used in robotics, embedded systems, and electronic projects where automation is an essential part of the system. Based on the ATmega328P processor, it is quite powerful and easy to program using Arduino IDE software. All that is needed is a USB cable to transfer the program from your computer to the board.

Si5351A

The Si5351A is a very popular clock generator used in many commercial and homebrew projects during the last few years. The Si5351A board does have limitations. Although it is a highly capable and stable board, the output is a square wave with odd harmonic frequencies present in the output. The 7 dBm (5 milliwatt) square wave output does make a good source for some mixers. Phase noise is also higher than other popular programmable signal sources. A quick search of the internet will yield a wealth of data concerning the performance of the Si5351A IC.

Excellent library routines are available on the internet to simplify Si5351A frequency programming. Instead, I chose to program the Si5351A PL and MultiSynth functions directly without the use of library routines. The resultant code is very simple compared to other routines, but works quite nicely in this application. I found I could easily program the Si5351A board up to 150 MHz using PLL divider techniques. Unfortunately, PLL divider techniques create glitches each time the frequency is changed. Fixed PLL frequencies using MultiSynth division provide glitch-free tuning, but the frequency range is limited to 112.5 MHz using this method.

The Arduino sketch is developed for a Si5351A using a 25 MHz clock frequency. Minor sketch changes are required to accommodate other clock frequencies.

OLED Display

The monochrome 0.96" 128x64 OLED graphic display is very small, but highly readable due to the high contrast OLED display. It uses no backlight and uses very little power. Although this is a graphic display, only text data is used in this application.

Construction

Construction of the unit is not critical provided adequate RF techniques are used. I first built the system using a solderless breadboard without any problems. The circuit (Figure 3) was then transferred to a piece of perfboard and placed in a box along with a 9 volt battery and an ON/OFF switch (Figure 2).

Winding the FT37-43 ferrite toroid transformer may seem difficult, but is quite easy and forgiving. I used #26 (#28 or #30 may also be used) enameled wire for both the primary and secondary windings of the transformer. The most important thing to remember is that one turn is counted each time the wire passes through the toroid's hole. Refer to figures 4 and 5. The most difficult part of the entire project was cutting an opening in the box for the OLED display.

The Arduino Nano in my unit is powered from a 9 volt battery connected to "VIN". You can use the USB programming cable, a separate 5VDC source connected to the "+5V" pin, or 7-12VDC source connected to the Arduino Nano's "VIN" pin to power your unit. The Si5351A, DS3231SN, and the display are powered by the "5V" pin of the Nano.

Software Installation and Setup

The Arduino download website <<http://arduino.cc/en/Main/Software>> outlines installation instructions for the first-time Arduino user.

The sketch requires an open source library; "SSD1306Ascii" by Bill Greiman. The library is found in the Arduino IDE at; Sketch > Include Library > Manage Libraries (Windows users will find the menu bar on top of the active window. Linux and MAC users will find the menu bar at the top of the primary display screen.).

In the unlikely event of operational problems, I suggest the builder check the Arduino, and OLED display individually to ensure proper operation. The Arduino board can be checked using some of the example sketches provided with the open source Arduino software. The simple "blink" example sketch will confirm that a sketch can be loaded and the Arduino board is functioning. The OLED display may be checked by running one of the AvrI2C examples found with the SSD1306Ascii library

There are a number of pre-set frequencies divided into bands within the configuration data section of the sketch. Multiple frequencies may be configured in this manner. Instructions are found in the sketch if changes are required.

Calibration

Calibration is not required for general use.

Perform the following procedure if the unit is to be used as a signal source and greater accuracy is required:

- Connect the tuning aid to a frequency counter
- Set to 25 MHz
- Annotate counter frequency in Hz
- Subtract 25 MHz from counter reading
- Annotate the difference in Hz (i.e. -245)
- Update the variable “CalFactor” on line 108 to reflect the difference
(i.e. “CalFactor = -245”)

Operation

Operation is very simple and straightforward. After turning the unit on, depress pushbutton 3 to select the desired band. Pushbutton 4 is be used to select a convenient frequency change step and pushbuttons 1 and 2 are used to increase or decrease the frequency. Connect the unit to an antenna tuner or magnetic loop antenna and tune for a bargraph null on the OLED display.