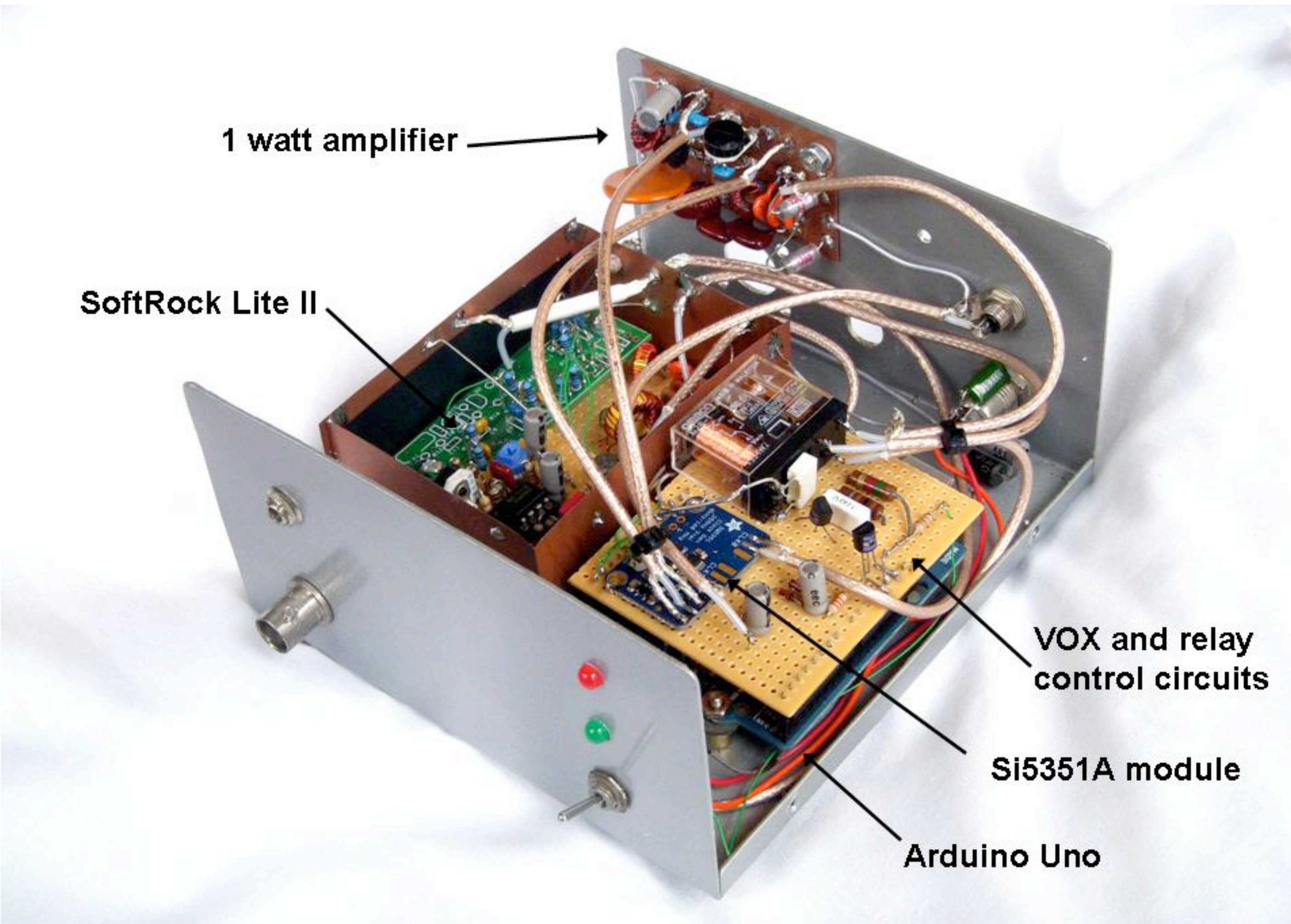
Si5351A WSPR Transceiver

An inexpensive project based around a partially built $21 SoftRock Lite II receiver and an $8 Si5351A module. The transceiver provides single band operation between 630 and 10 meters.

Gene Marcus W3PM GM4YRE



Introduction

There are many transceivers in the marketplace that provide good performance at a reasonable cost. However, I enjoy building and experimenting, so I documented this project to share with those who have similar interests. It can be built as a mono band transceiver as outlined on Figure 1, or turned into a multi-band transceiver with the addition of switched band pass/ low pass filters and minor software changes.

The project began with homebrew software defined receiver (SDR) using a pair of SBL-1 mixers. I wanted to experiment using a FST3253 mixer when I noticed that fivedash.com offered a complete SoftRock Lite II receiver kit based around the FST3253 for only $21 USD.

The Si5351A is an interesting device that was designed as a substitute for crystal oscillator clocks for any three frequencies between 8 KHz and 150 MHz. Although the board is specified for a wider bandwidth, the output from this project is limited to 1 through 112.5 MHz due to software algorithm I used. A Si5351A (25 MHz clock) breakout board is available from Adafruit Industries for less than $8 USD.

An Arduino Uno (or Nano) provides the processing power to control the Si5351A and the antenna relay. The Si5351A is used to provide the Local Oscillator frequency for the SoftRock Lite II. The Si5351A is also used as the WSPR exciter to drive an amplifier of your choice. In this project I describe a 1 watt 80 meter amplifier that I use.

Theory of Operation

The SoftRock Lite II is a typical software defined receiver that converts the received frequency to a pair of audio frequencies in quadrature (separated in phase by 90 degrees) to be processed by a software defined radio program. For WSPR operation this will be a program such as WSPR 2.12 or Linux WSPR 4.0. The SoftRock Lite II requires a local oscillator that is four times the receive frequency to derive the quadrature audio signal. The Si5351A (Figure 2) provides this frequency in lieu of the SoftRock's on-board oscillator.

Unlike a typical transceiver used in WSPR operation, the audio input is not used to modulate the transmitter. Instead, it is only used to trigger the VOX circuit (Figure 4). The station callsign, power level, and grid locator information are included in the Ardrino software which processes the WSPR message to control the SI5351A module. The Si5351A is used as a WSPR exciter to drive the RF amplifier (Figure 5).

The choice of a RF amplifier is not critical. Any non-linear class C or E amplifier may be used provided it is capable of being driven by the Si5351A. I used an modified 80 meter class E amplifier built using design parameters from a spreadsheet by WA0ITP. The spreadsheet is located at: <http://www.wa0itp.com/class%20e%20design.html>. The amplifier is slightly modified to use two BS170 MOSFETS's in parallel to prevent any overheating during two minute WSPR transmission times. The devices run cool to the touch at 1.5 watts when using 13.8 VDC.

The Si5351A, VOX, and the antenna relay (Figure 3) are controlled by an Arduino Uno. An Arduino Nano can be substituted without any modification. Excellent library routines are available on the internet that simplify Si5351A frequency programming. Instead, I chose to program the Si5351A PLL 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.

Construction

There are two versions of Si5351A modules available (25 and 27 MHz). Ensure you are using the 25 MHz clock version.

Layout is not critical. I built the unit as modules using perfboard construction techniques for everything except the RF amplifier. The RF amplifier was built dead-bug style using a scrap piece of PCB board.

There are many amplifier kits and amplifier designs on the internet that will work. The builder may want to eliminate the RF amplifier and simply use a low pass filter on the Si5351A output to run QRPp. If you use the WA0ITP spreadsheet to obtain the RF amplifier design parameters, you will probably end up with non-standard capacitor values. I used series and parallel combinations of standard value capacitors to get close to the calculated values. *Important note:* If the 80 meter amplifier in figure 5 is used, be sure to set the wiper to the ground side of the 5K potentiometer. This will ensure the BS170 MOSFETS's are biased off before applying power.

When building the SoftRock Lite II, skip the build instructions for the local oscillator. If you are using a pre-built SoftRock Lite II, remove R11 and cut the trace going from the collector of Q2 to U2.

Software Installation and Setup

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

Open "Si5351A_XCVR.ino" into the Arduino development environment and customize the script to personalize your station information. The station information is found near the beginning of the script. Only change the data highlighted in red. Do not change any other data. Be sure to remember to save your changes and upload the file into the Uno.

Enter home callsign and grid square:

```
char call2[13] = "W3PM";   //e.g. "W3PM" or "GM4YRE"
```

char locator[7] = "EM64"; // Use 4 character locator e.g. "EM64"

Note:
- Upper or lower case characters are acceptable.
- Compound callsigns may use up to a three letter/number combination prefix followed by a “/”.
- A one letter or two number suffix may be used preceded by a “/”

Enter power level in dBm:

Min = 0 dBm, Max = 43 dBm, steps 0,3,7,10,13,17,20,23,27,30,33,37,40,43

byte ndbm = 30;

Select band from the following list:

```
  474200,   // Band 0
  1836600,  // Band 1
  3592600,  // Band 2
  5287200,  // Band 3
  7038600,  // Band 4
  10138700,// Band 5
  14095600,// Band 6
  18104600,// Band 7
  21094600,// Band 8
  24924600,// Band 9
  28124600,// Band 10
```

// Set default band
int band = 2; // 80 M

Calibration:

The Si5351A should be calibrated to the nearest Hz when putting the transceiver into service for the first time.

Perform the following to calibrate the Si5351A:
- Connect frequency counter to the Si5351A CLK1.
- Hold Arduino pin 2 LOW during a reset.
- Release pin 2
- The counter should now indicate a frequency near 25 MHz
- Annotate counter frequency in Hz.
- Subtract 25 MHz from counter reading.

- Annotate the difference in Hz (i.e. -396).
- Set the CalFactor variable to the value.

const int CalFactor = -396;

Transmit Offset and In-band Frequency Hopping:

Select the transmit offset frequency in Hz. Range = 1400-1600 Hz
int TXoffset = 1525;

If in-band frequency hopping is selected, the transmit frequency will increment by 15 Hz before each transmission. When it reaches the top of the WSPR frequency window it will loop back to the bottom of the window. If in-band hopping is not selected the unit will default to the offset frequency above.

In-band transmit frequency hopping? (1=Yes, 0=No)
const int FreqHopTX = 1;

WSPR I/Q Frequency Offset:

This sets the local oscillator's offset for I/Q processing. Leave this variable set to 12000 unless you use the analog I/Q processor that I describe on my website. In that case, set to 0.

WSPR I/Q Frequency Offset
- Enter 12000 to enable WSPR I/Q mode
- Enter 0 to disable WSPR I/Q mode

const int IQwspr = 12000;

80 Meter Amplifier Bias Adjustment

Attach a dummy load to the unit and monitor RF power and power supply current. Turn on the power and take note of the power supply current reading (approximately 90 mA.). Slowly adjust the 5K bias potentiometer until an increase in current is noted. At this point back off a bit until no additional current is drawn. Set the PC's line audio output to maximum. Trigger the VOX to turn on the transmitter. The amplifier should indicate approximately one watt of power at this point. The bias may be increased to obtain more power at the expense of amplifier efficiency and resultant MOSFET heating.

Wiring Diagram and Schematic

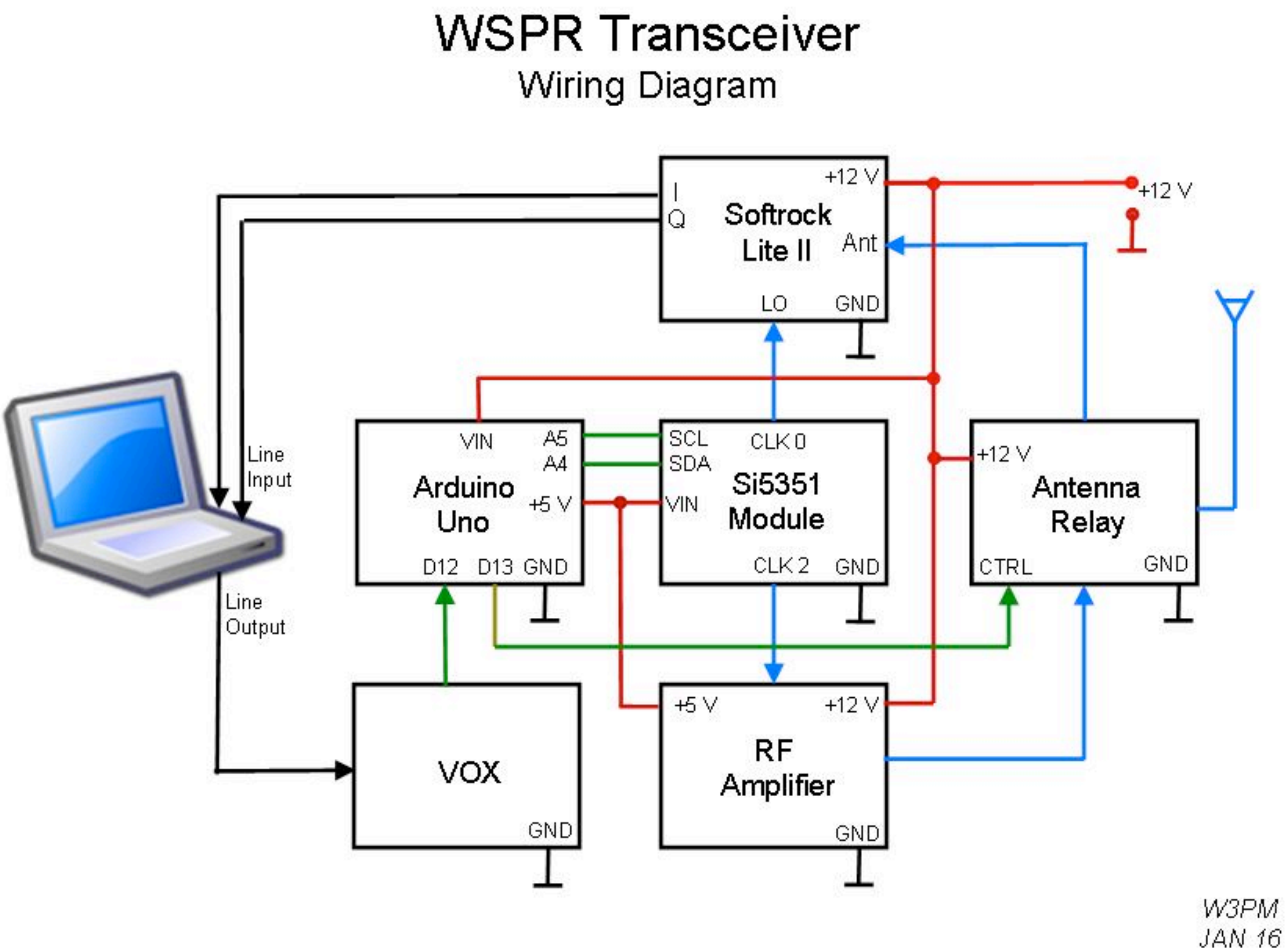


Figure 1. Wiring Diagram

Si5351A Module and SoftRock Lite II

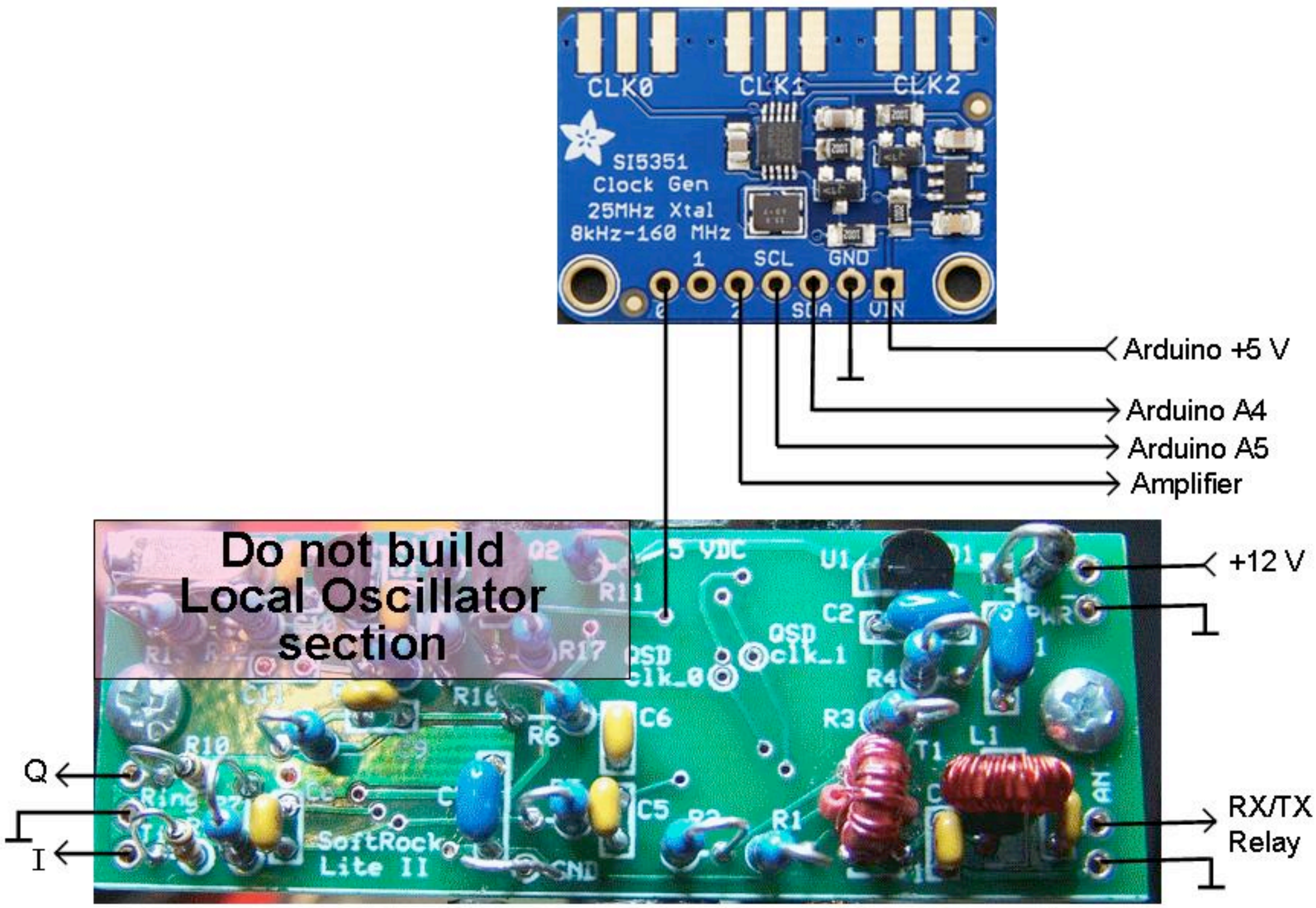


Figure 2. Si5351A Module and Modified SoftRock Lite II

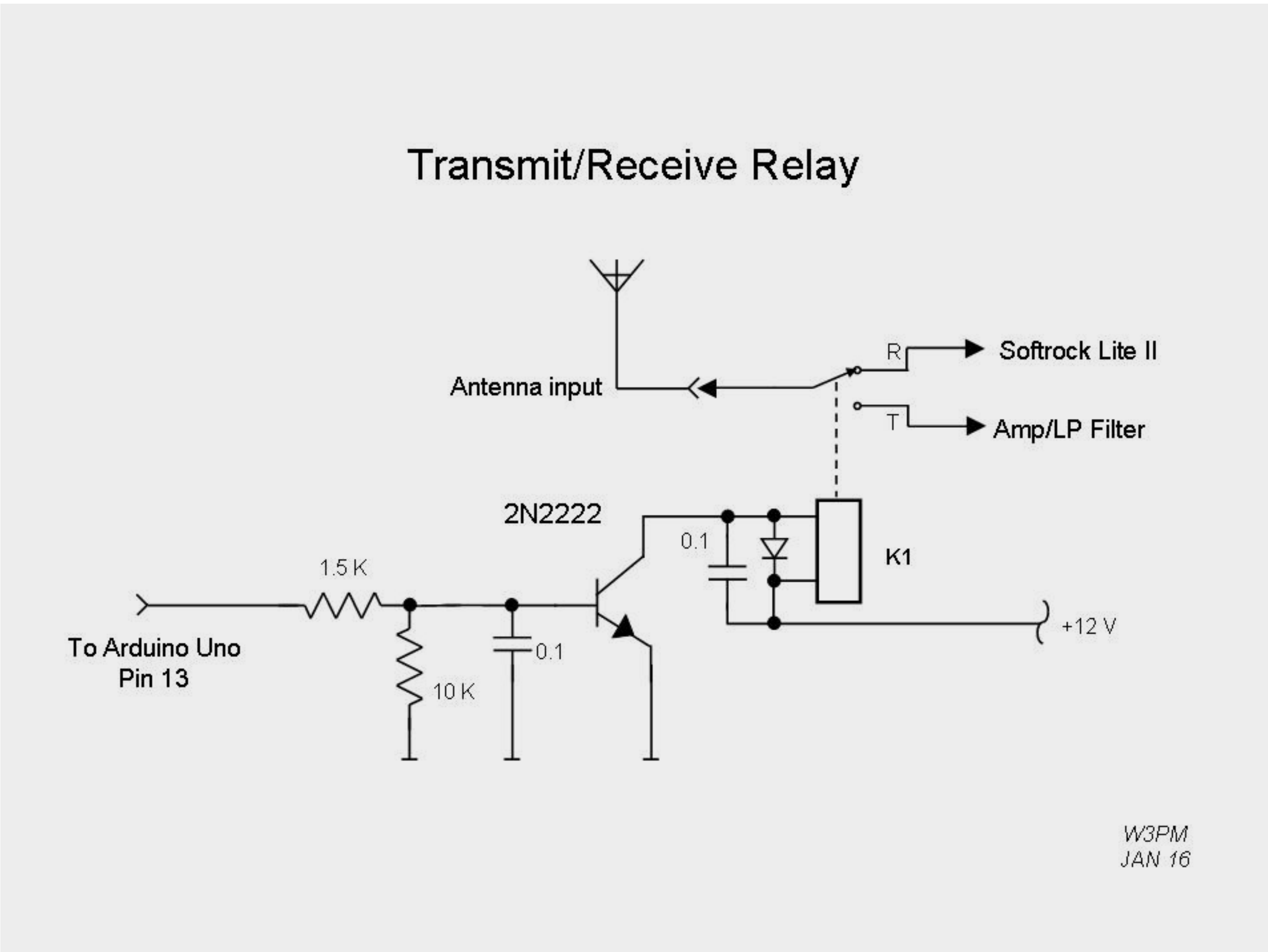


Figure 3. Transmit/Receive Relay

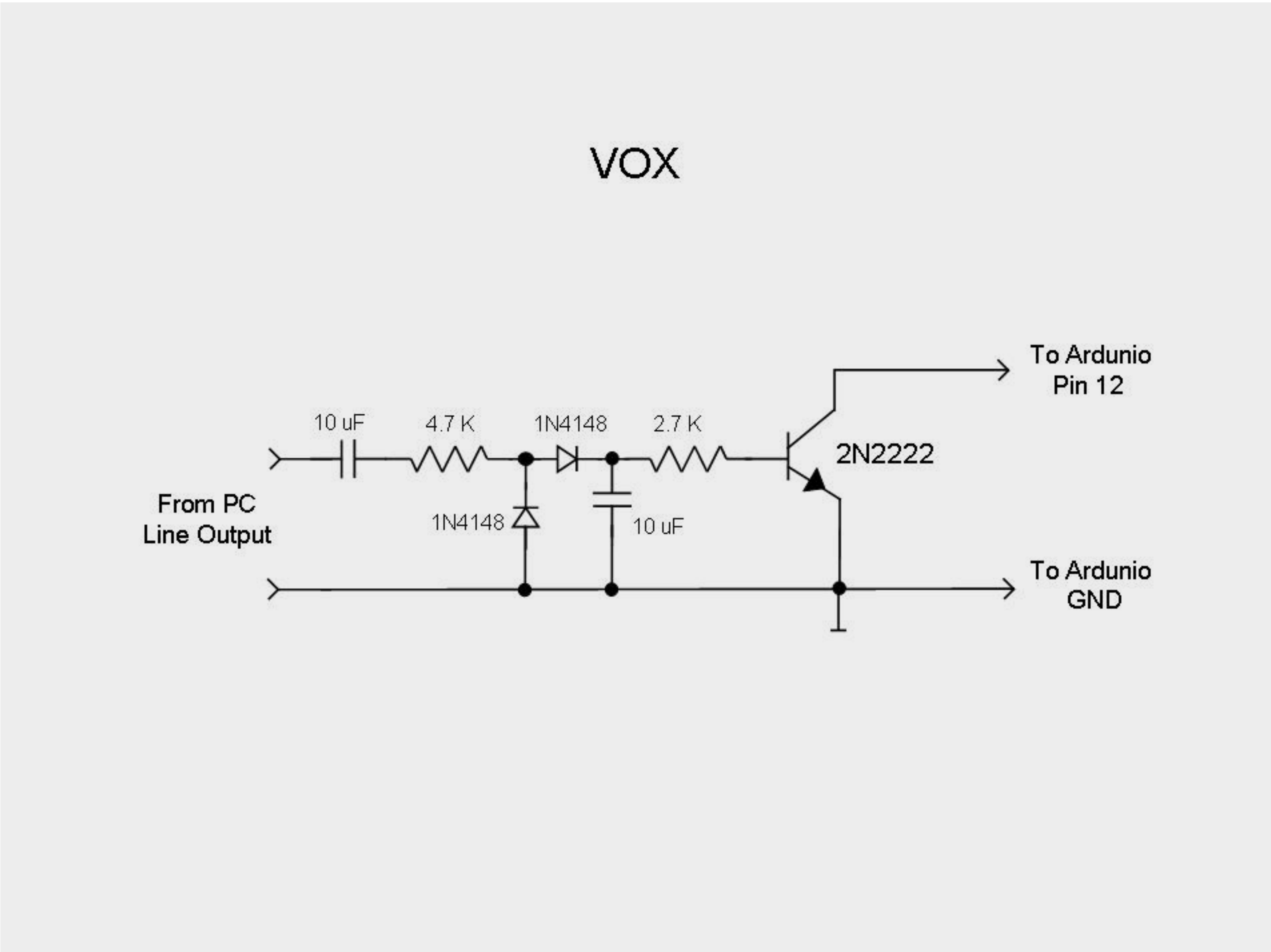

VOX
10 uF
4.7 K
1N4148
2.7 K
To Ardunio
Pin 12
2N2222
From PC
Line Output
1N4148
10 uF
To Ardunio
GND

Figure 4. VOX

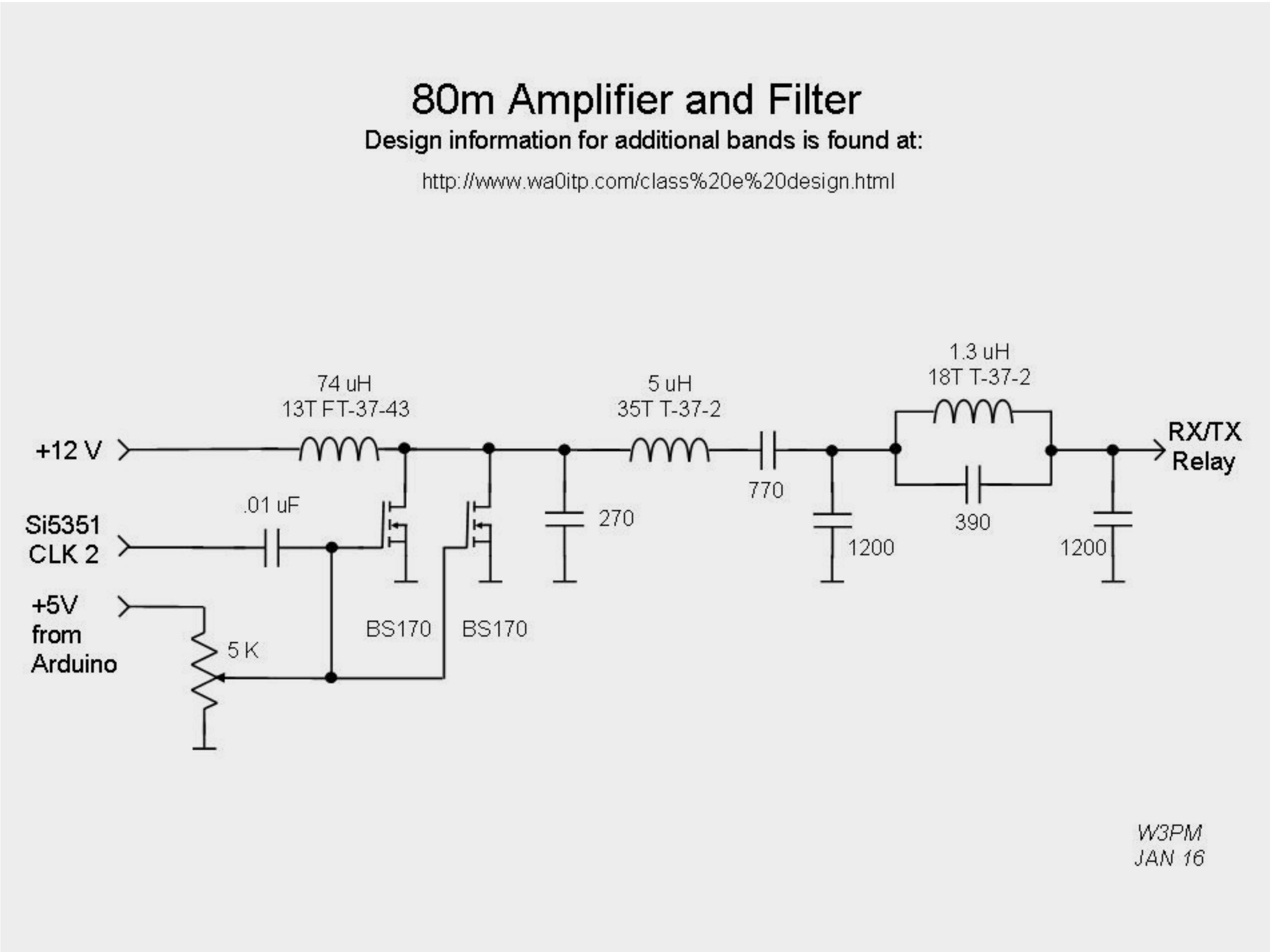


Figure 5. 80m Amplifier and Filter