

WSPR in an educational Project

How to simply program a WSPR transmission with an Arduino?

Anthony LE CREN, F4GOH
f4goh@orange.fr

Abstract

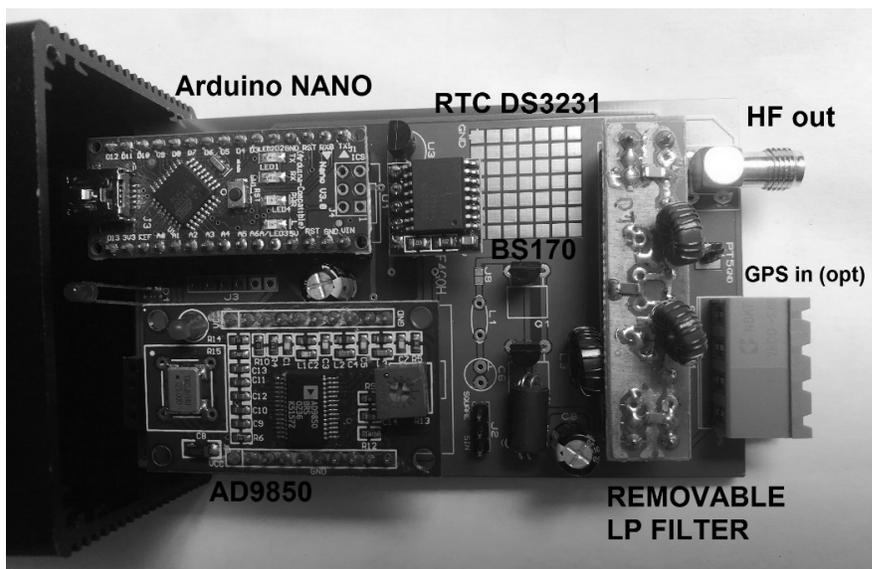
Since several years I have been trying to design projects to promote HAM radio, and especially for young people who want to pass their radio license. But most of the time, projects are complicated and require a good level of electronics. The article described below is for people who know to blink a led with an Arduino and wiring a breadboard.

Introduction

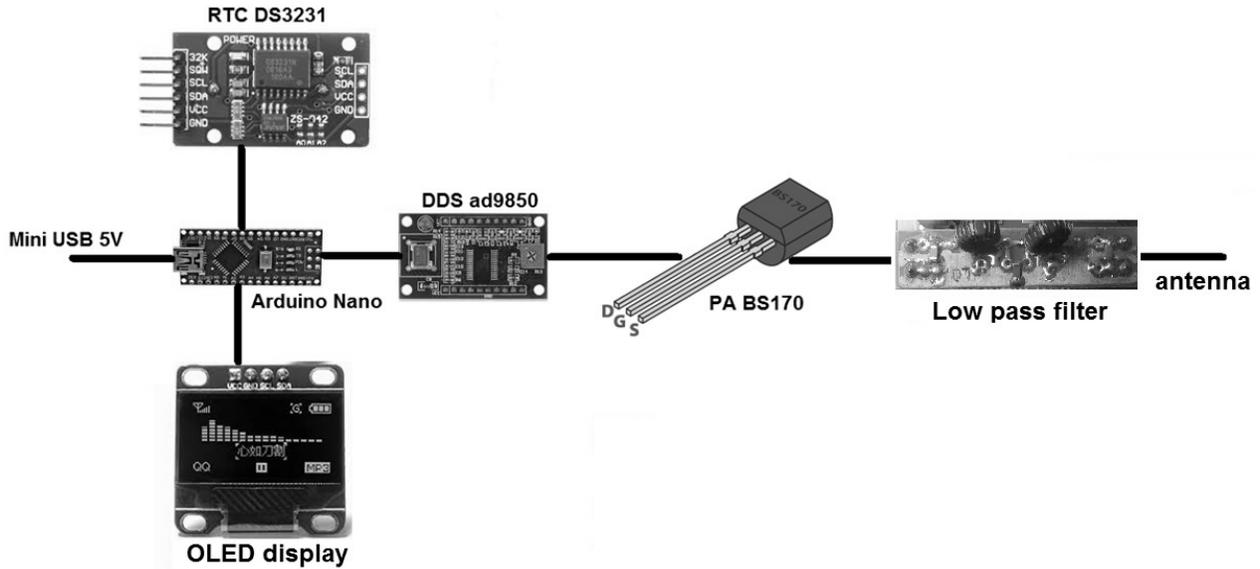
The idea is to generate WSPR signal with an AD9850 controlled by an Arduino, A DS3231 RTC synchronize time to transmit WSPR. The level of programming has been limited to the easiest, in fact, a simple loop makes it possible to transmit the WSPR signal. In addition the assembly can be wired on a breadboard step by step. You will learn how to program a DDS, make filters and setting the transmission frequency.

WSPR beacon features: Arduino nano (atmega328P) based microcontroller :

- Support Arduino IDE 1.0+ (OSX/Win/Linux)
- Power via USB or External Source + 5v
- 2 I/O Pins (for GPS receiver)
- Removable filter
- AD9850 DDS
- RTC DS3231
- BS170 power amplifier 0.1W
- Fit in Aluminium Instrument Box Enclosure Case 100x66x43
- PCB available

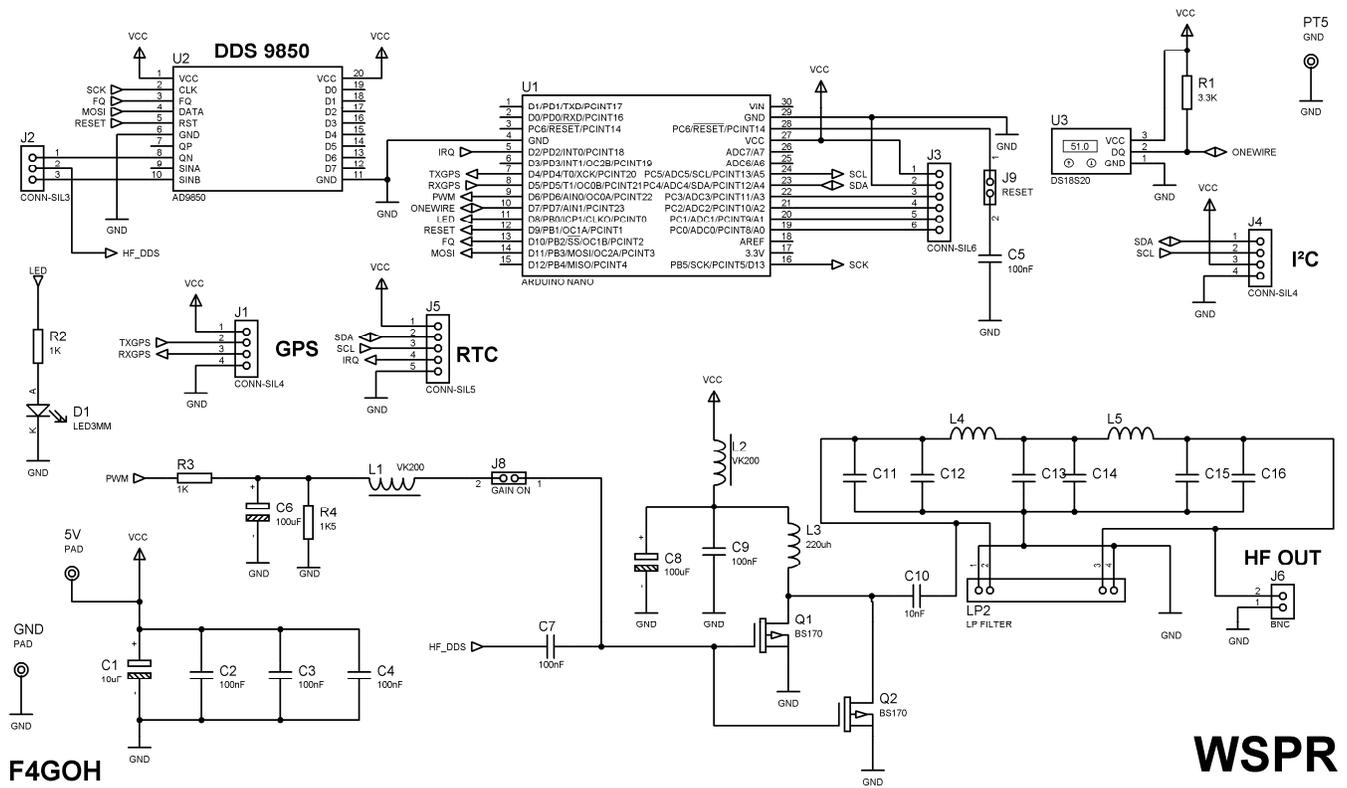


1 Description



The real-time clock synchronizes Arduino to transmit every even minute. The Oled screen is optional and simply displays the time. DDS (Direct Digital Synthesis) generates an RF signal. The transistor BS170 is used as a class E amplifier and requires a low-pass filter.

Schematics:



WSPR

Two ways to use bs170 :

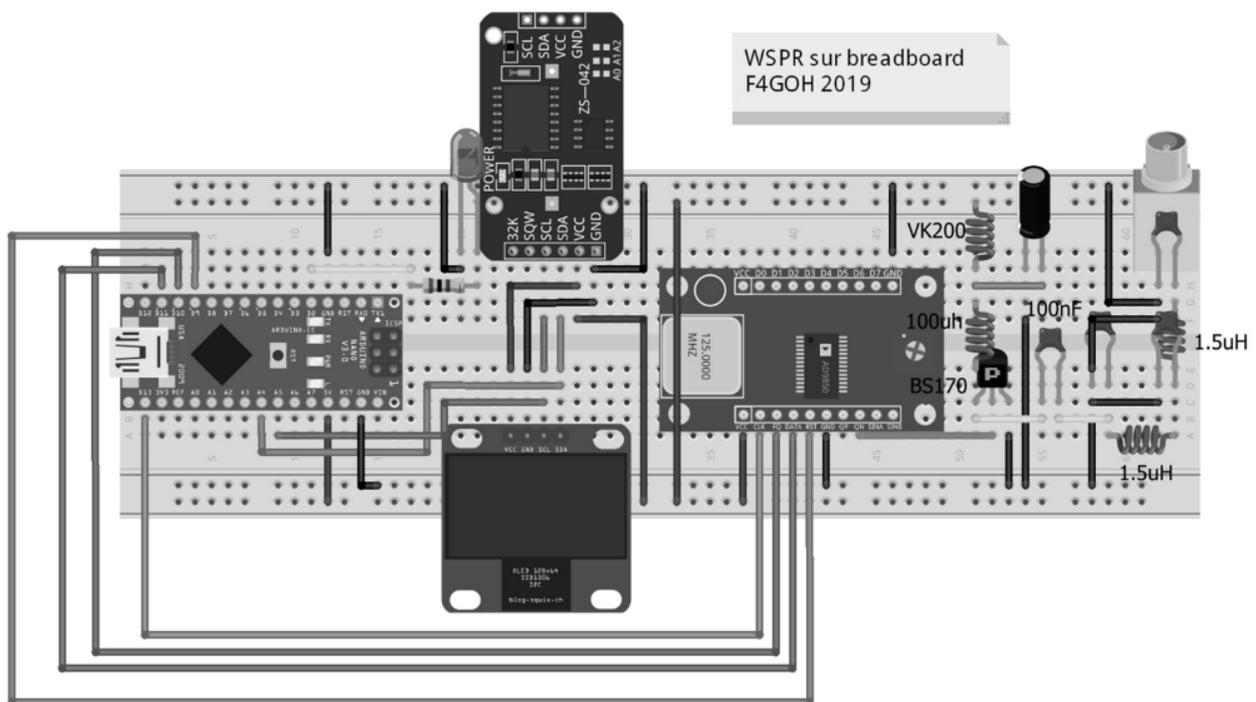
– use QN (J2) DDS output and replace C7 capacitor by a strap. don't put R3,C6,R4,L1, and left j8 open (this is the configuration i used). Adjust DDS 9850 trim to have a ~ « square wave » at the QN output. When transmission is OFF QN voltage should be 0V. Use any scope, you can find GND connection at PT5.

– use SINB(J2) and apply voltage polarization on BS170 gate. (put R3,C6,R4,L1 and strap J8) then apply analogWrite command on nano pin 6. (to adjust gain by software). if gain value is too high bs170 may be destroyed.

You can find a pad space to experiment your own stuff and change bs170 to another transistor.

J1 connector could be dedicated for GPS input. But it can be used for another things like external lowpass filter commutation.

Ds1820 is planned just for fun, to send temperature by RTTY or PSK modulation.



fritzing

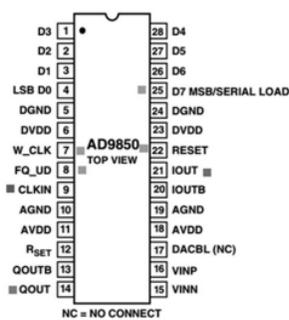
The transmitter can be wired on a breadboard

2 How to program AD9850 DDS?

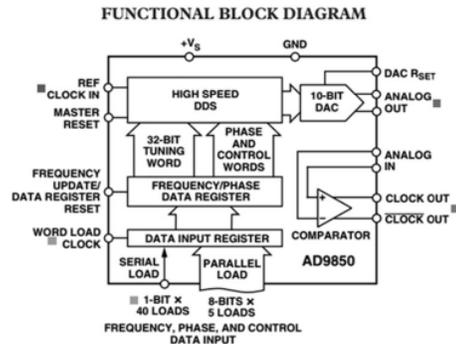


CMOS, 125 MHz
Complete DDS Synthesizer

AD9850



- FEATURES**
- 125 MHz Clock Rate
 - On-Chip High Performance DAC and High Speed
 - Comparator
 - DAC SFDR > 50 dB @ 40 MHz A_{OUT}
 - 32-Bit Frequency Tuning Word
 - Simplified Control Interface: Parallel Byte or Serial Loading Format
 - Phase Modulation Capability
 - 3.3 V or 5 V Single-Supply Operation
 - Low Power: 380 mW @ 125 MHz (5 V)
155 mW @ 110 MHz (3.3 V)
 - Power-Down Function
 - Ultrasmall 28-Lead SSOP Packaging



Arduino nano uses its SPI (Serial Peripheral interface) to update the frequency in the DDS

Table III. 8-Bit Parallel Load Data/Control Word Functional Assignment

Word	Data[7]	Data[6]	Data[5]	Data[4]	Data[3]	Data[2]	Data[1]	Data[0]
W0	Phase-b4 (MSB)	Phase-b3	Phase-b2	Phase-b1	Phase-b0 (LSB)	Power-Down	Control	Control
W1	Freq-b31 (MSB)	Freq-b30	Freq-b29	Freq-b28	Freq-b27	Freq-b26	Freq-b25	Freq-b24
W2	Freq-b23	Freq-b22	Freq-b21	Freq-b20	Freq-b19	Freq-b18	Freq-b17	Freq-b16
W3	Freq-b15	Freq-b14	Freq-b13	Freq-b12	Freq-b11	Freq-b10	Freq-b9	Freq-b8
W4	Freq-b7	Freq-b6	Freq-b5	Freq-b4	Freq-b3	Freq-b2	Freq-b1	Freq-b0 (LSB)

The frequency is not update in HZ. You must send a 32-bit word proportional to the frequency as shown in the formula below.

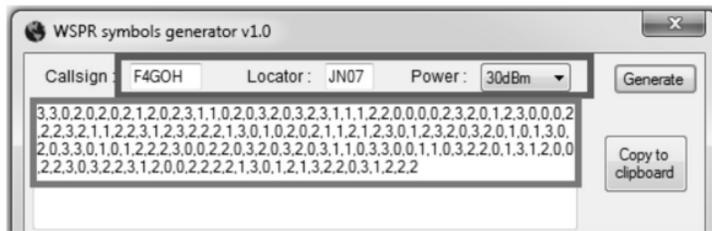
$$\frac{f_{OUT}}{\text{output Frequency}} = \frac{(\Delta \text{Phase} \times \text{CLKIN})}{32 \text{ bit word} \times \text{ref Frequency } 125\text{mhz}}$$

$$\Delta \text{Phase} = \frac{2^{32} \times f_{OUT}}{\text{CLKIN}}$$

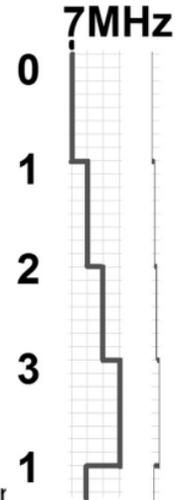
```
void setfreq(double f, uint16_t p) {
    uint32_t deltaphase;

    deltaphase = f * 4294967296.0 / (125000000 + factor);
    for (int i = 0; i < 4; i++, deltaphase >>= 8) {
        SPI.transfer(deltaphase & 0xFF);
    }
    SPI.transfer((p << 3) & 0xFF);
    pulse(FQ_UD);
}
```

3 WSPR encoding



base : 7.000.000 Hz
 0 : 7.000.000 Hz
 1 : 7.000.001.46 Hz
 2 : 7.000.002.92 Hz
 3 : 7.000.004.38 Hz



Resulting in 162 sequential symbols each with a value from 0 to 3

Modulation

Each symbol represents a frequency shift of $12000 / 8192$, or approximately **1.46Hz**, per Symbol Value giving four-level Multi-FSK modulation. The transmitted symbol length is the reciprocal of the tone spacing, or approximately **0.683** seconds, so the complete message of 162 symbols takes around **110.6** seconds to send and occupies a bandwidth of approximately **6Hz**.

To generate symbols, execute WSPR symbols generator :

- Put your callsign
- Put your locator
- And finish by power (20 dBm for 0.1W)
- Click Generate
- Paste symbols on wsprSimple.ino program.

```
wsprSimple | Arduino 1.8.9
Fichier Édition Croquis Outils Aide
wsprSimple
#define RESET 9 // or 10
#define frequence 7040100 //base freq

long factor = -1500; //adjust frequency to wspr band
int secPrec = 0;

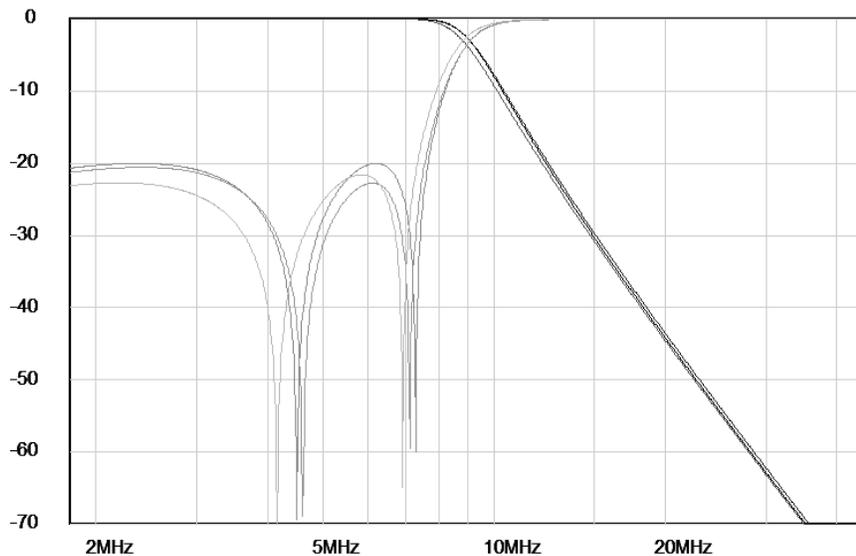
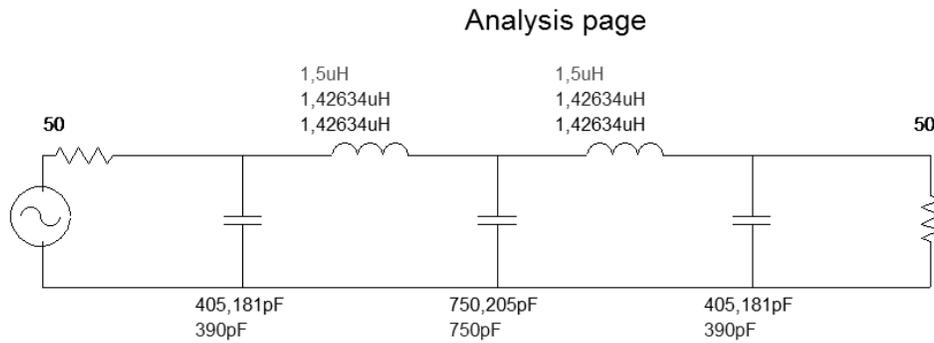
int wsprSymb[] = {3, 3, 0, 0, 0, 2, 0, 2, 1, 2, 0, 2, 3,
                  2, 2, 3, 0, 1, 3, 2, 0, 3, 3, 2, 3, 2,
                  0, 1, 3, 2, 1, 2, 1, 2, 2, 2, 3, 0, 0,
                  2, 1, 0, 1, 2, 0, 3, 3, 2, 2, 0, 2, 2,
                  };
```

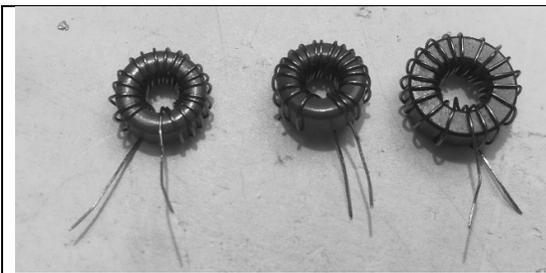
The FOR loop simply transmit the WSPR signal :

```
void sendWspr(long freqWspr) {  
  
  int a = 0;  
  for (int element = 0; element < 162; element++) { // For each element in the message  
    a = int(wsprSymb[element]); // get the numerical ASCII Code  
    setfreq((double) freqWspr + (double) a * 1.4548, 0);  
    delay(682);  
    Serial.print(a);  
    digitalWrite(LED, digitalRead(LED) ^ 1);  
  }  
  setfreq(0, 0);  
  Serial.println("EOT");  
}
```

4 Build a low pass filter (40 meters, 7.2 Mhz)

The best way is use svcfilter designer. This software calculates inductors and capacitors for any bandwidth.

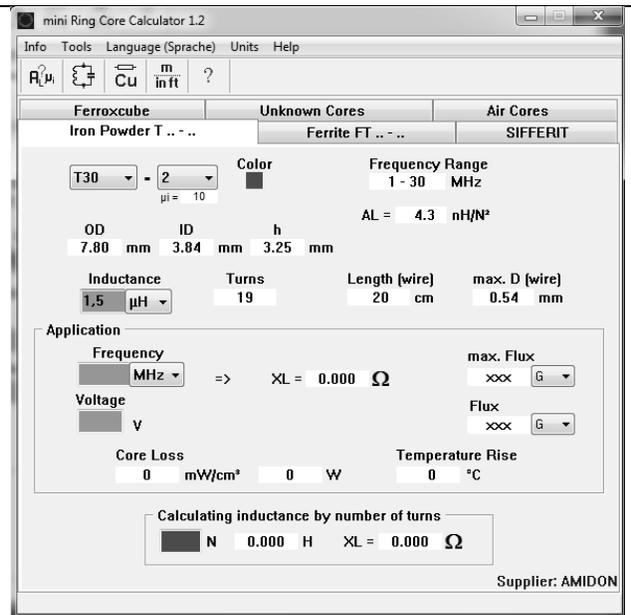




Use mini tore calculator software to check numbers of turns:

t30-2 :19 turns

ft37-43 (BS170 drain inductor): 18 turns



5 Conclusion

It was a pleasure to study DDS and WSPR modulation. This project was designed essentially for makers. Adapt it as you want. There are more screenshots on my web page. Reports are incredible with an 5V USB power.



References

<https://hamprojects.wordpress.com/>

<https://github.com/f4goh/WSPR>

http://www.g4jnt.com/wspr_coding_process.pdf

<http://tonnesoftware.com/svcdownload.html>

<https://constructions.f6fkn.com/downloads/minirk12-install.exe>

<https://kitsandparts.com/> (toroids)