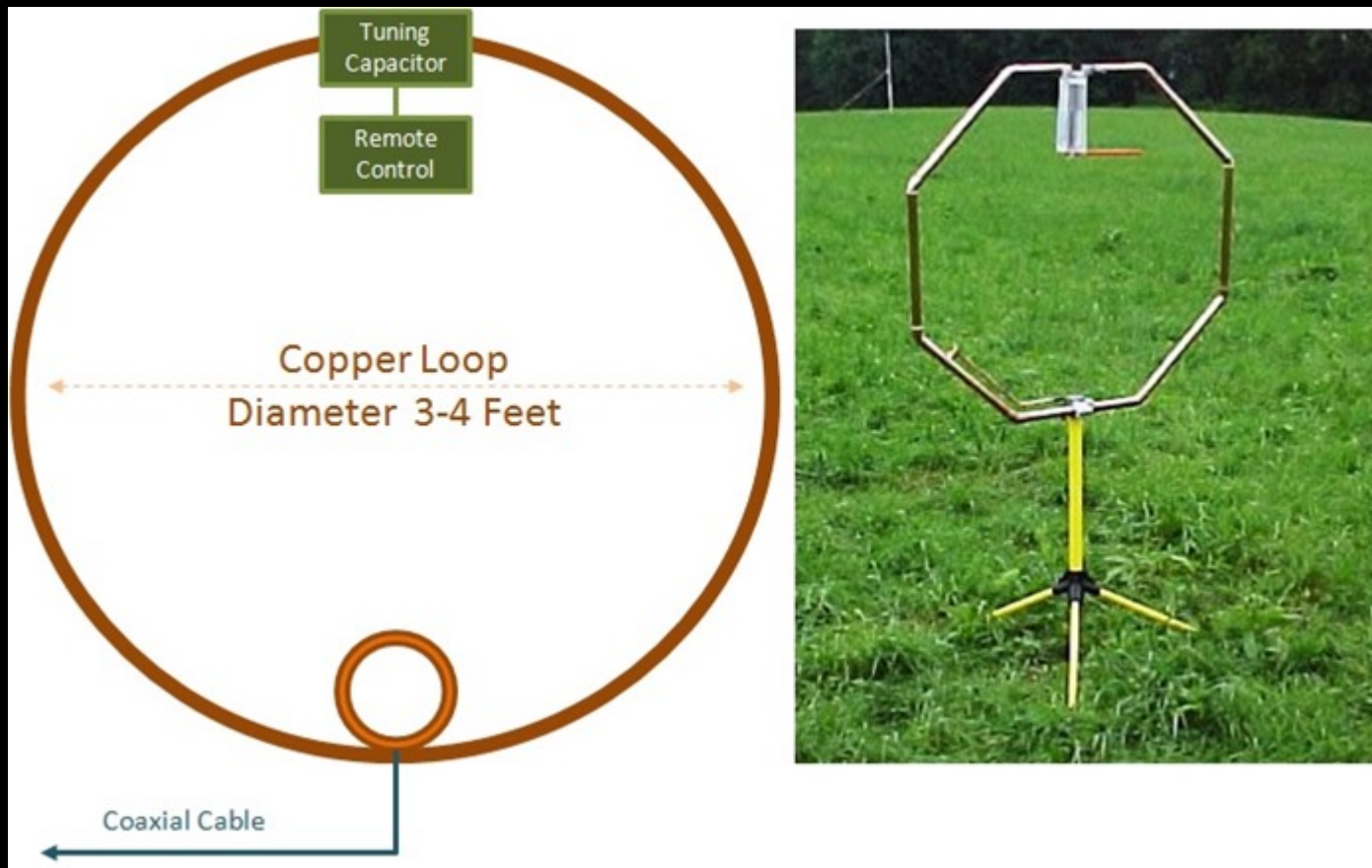


# Making and Ham Radio

*Making a Mostly Printed and  
Motorized Magnetic Loop Antenna*

Calgary Amateur Radio Association April 2017

# About Magnetic Loops



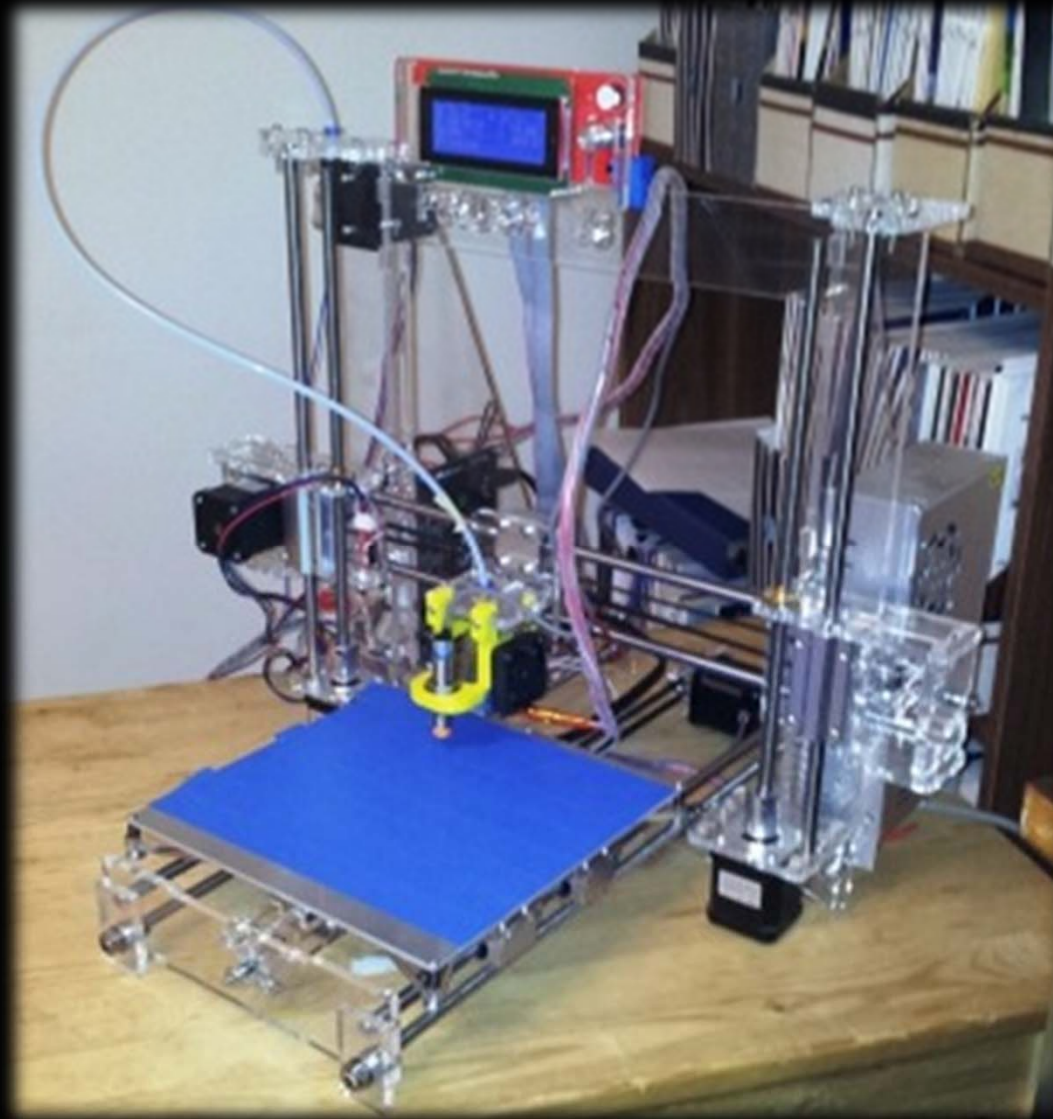
# Presentation

- Tools
  - 3D Printer
  - CNC Machine
- Magnetic Loop Antenna
  - Butterfly Capacitor
  - Loop Construction
  - Remote Control
  - Loop Testing and Performance

Making It Up

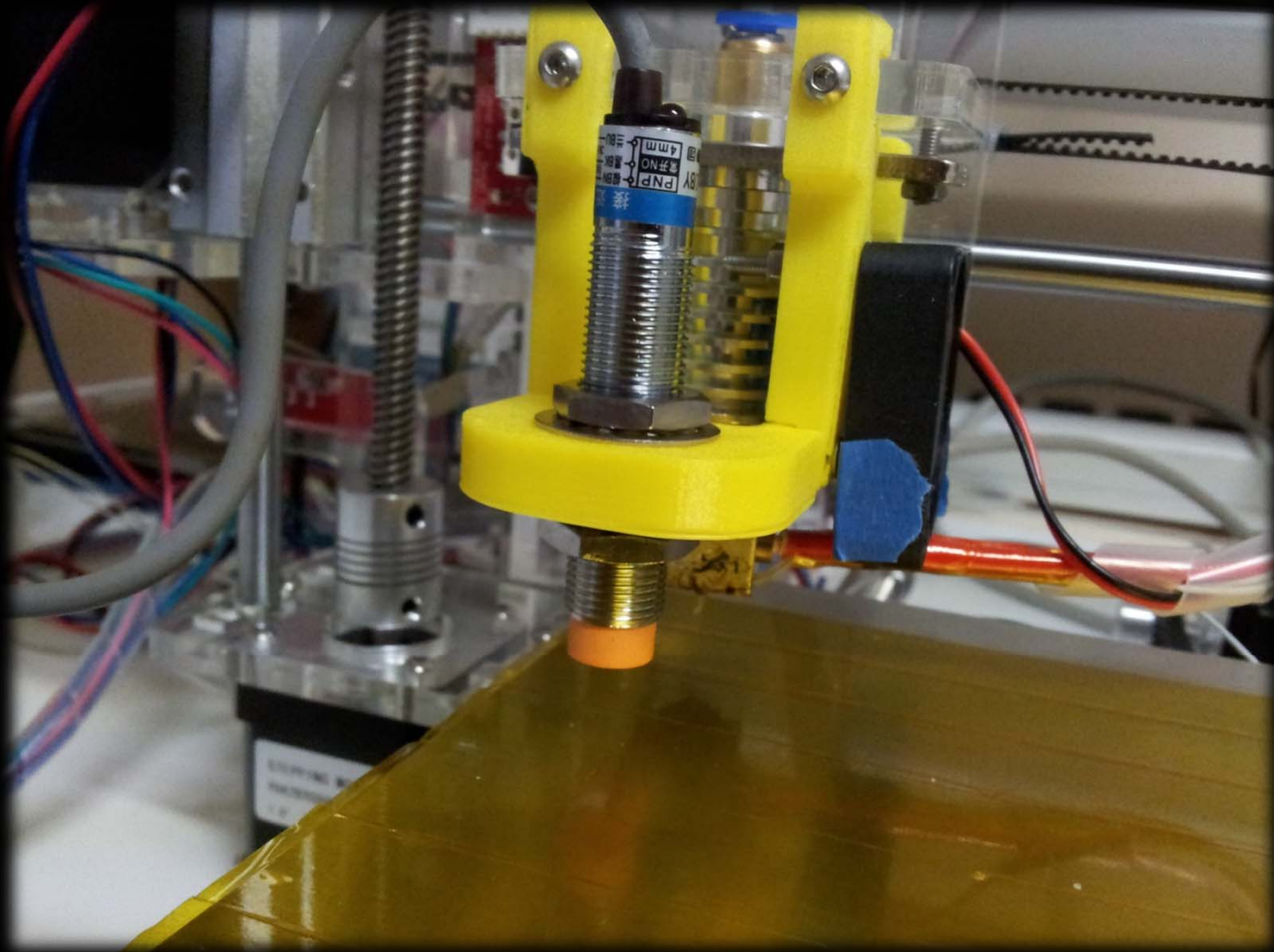
**3D PRINTER**

# Sunhokey Prusa i3



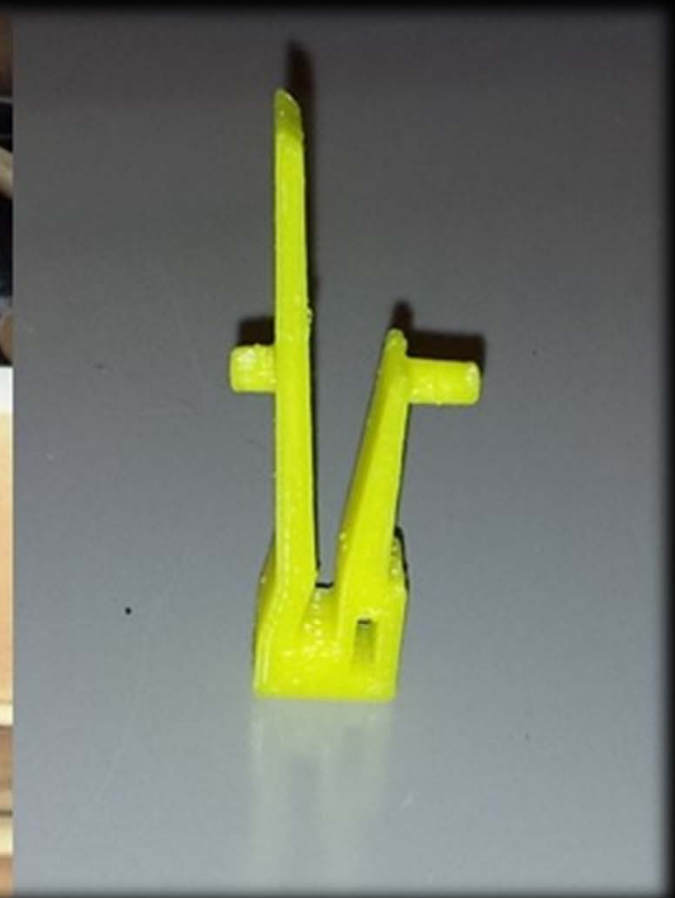
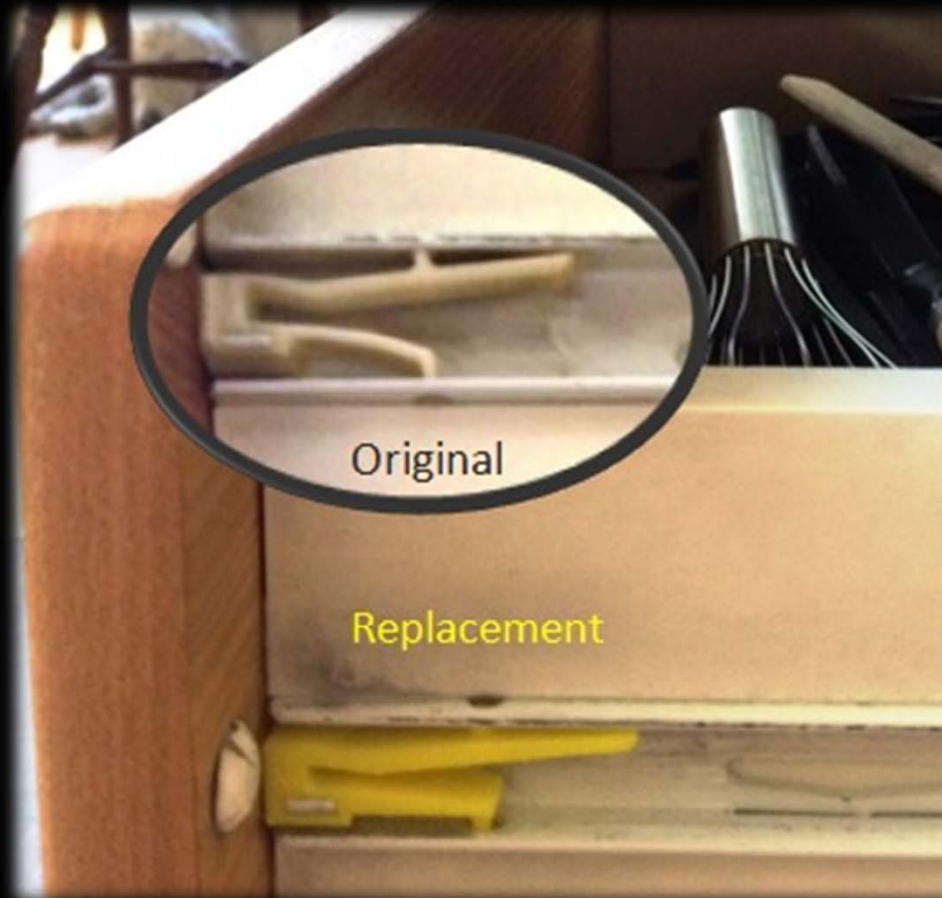
\$300 kit

# Automatic Bed Leveling

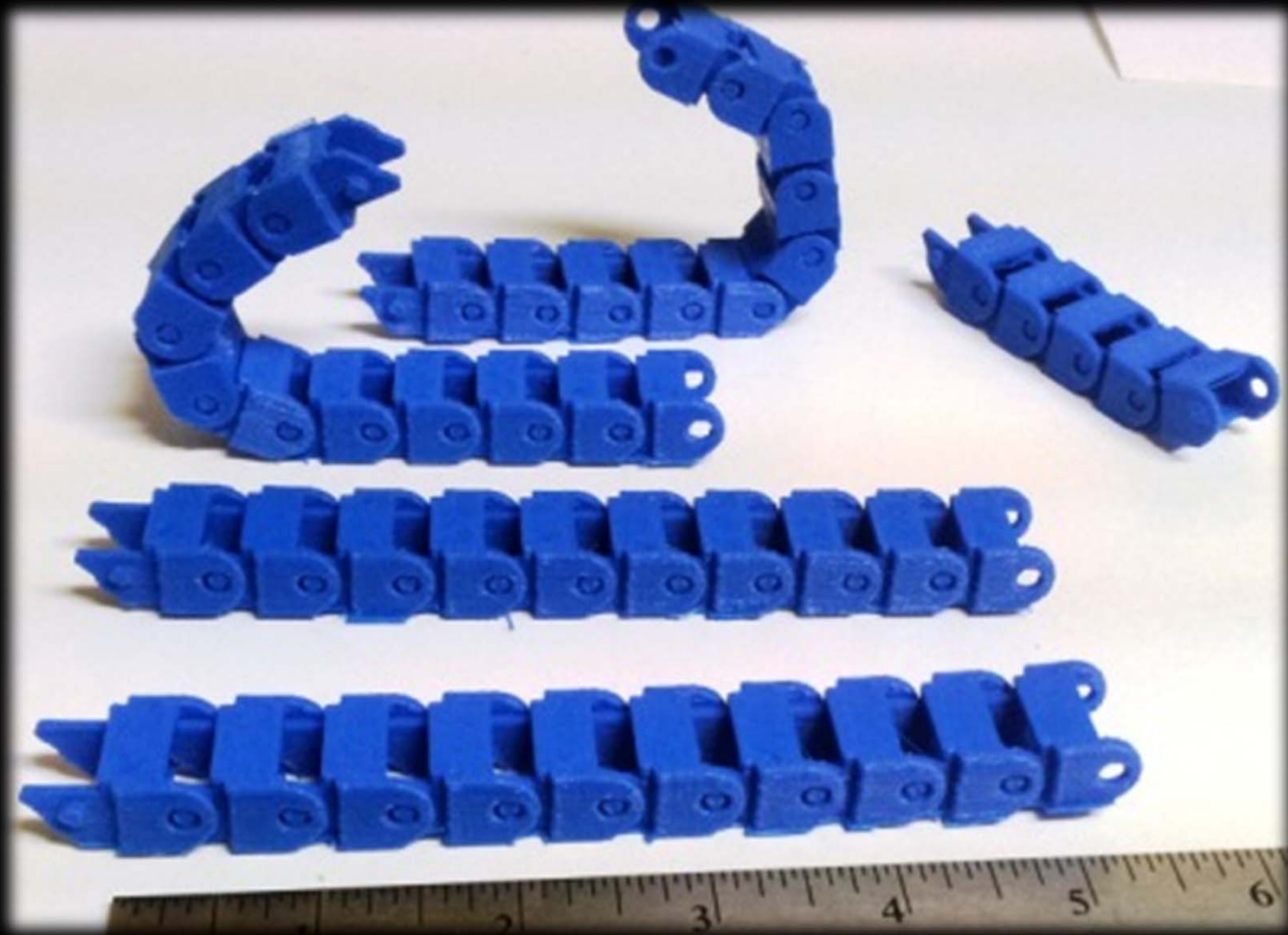


\$ 3 sensor

# Print Replacement Parts

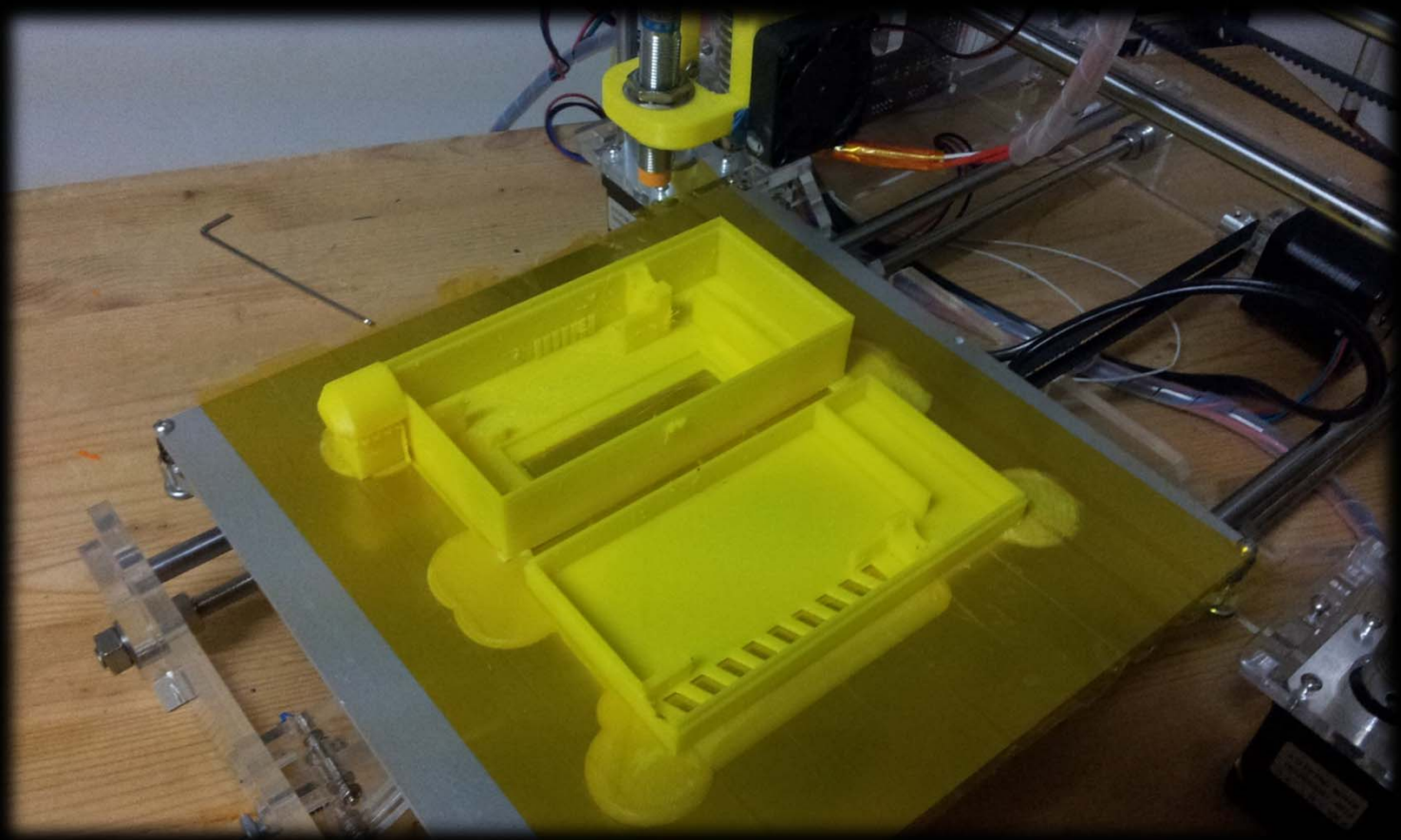


# Print a Cable Chain





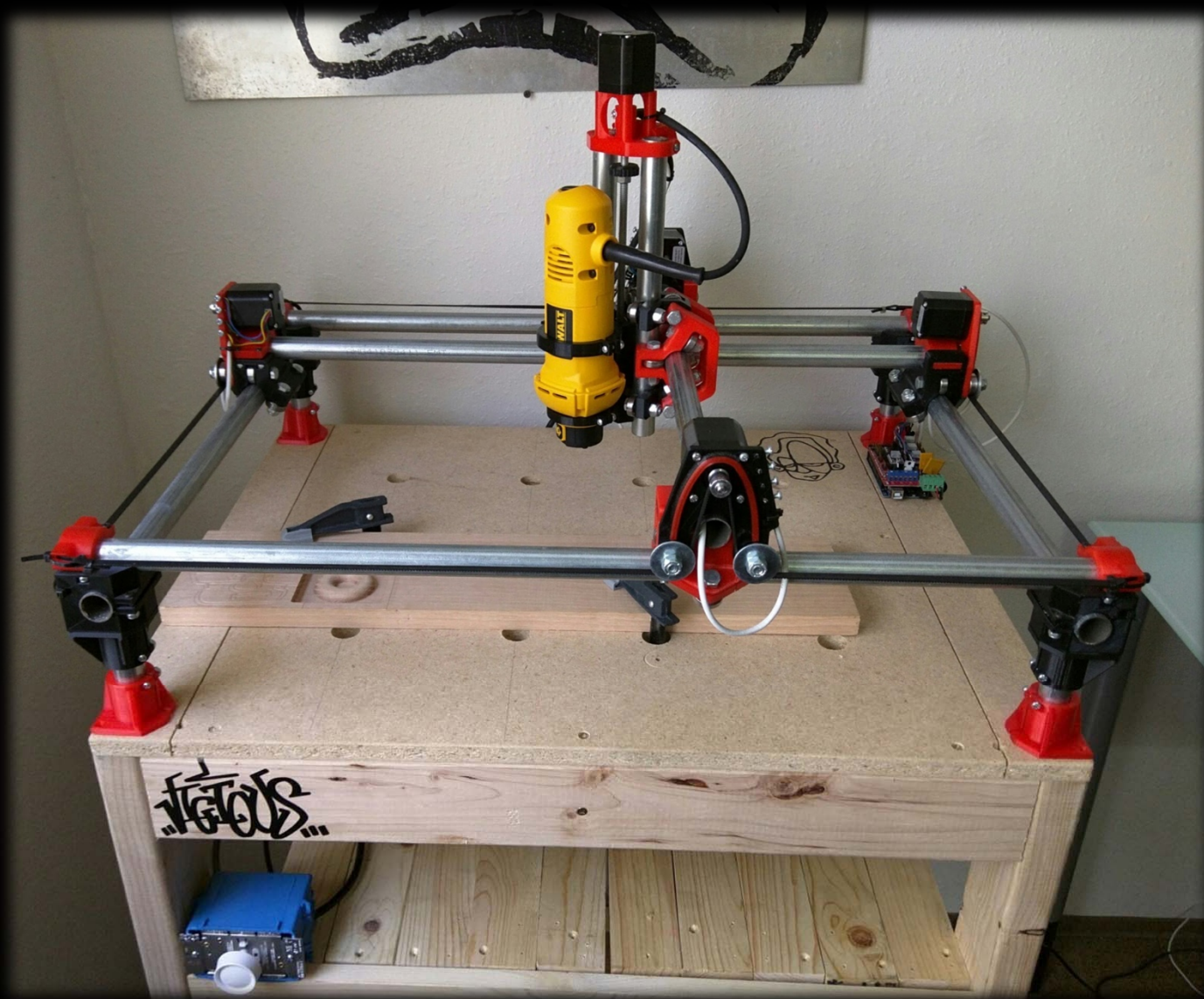
# Print a Project Box



Making It Up

**CNC MACHINE**

# Mostly Printed CNC

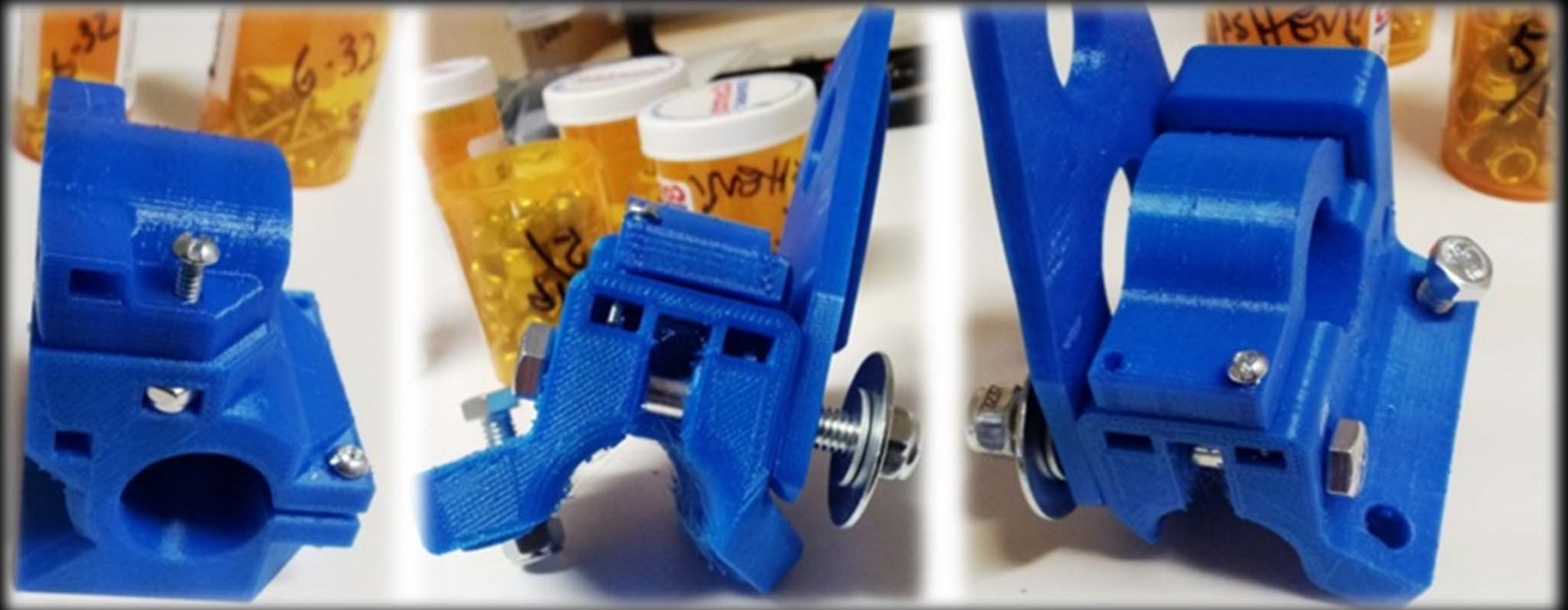


\$ 400 DIY

# MPCNC Hardware, Electronics

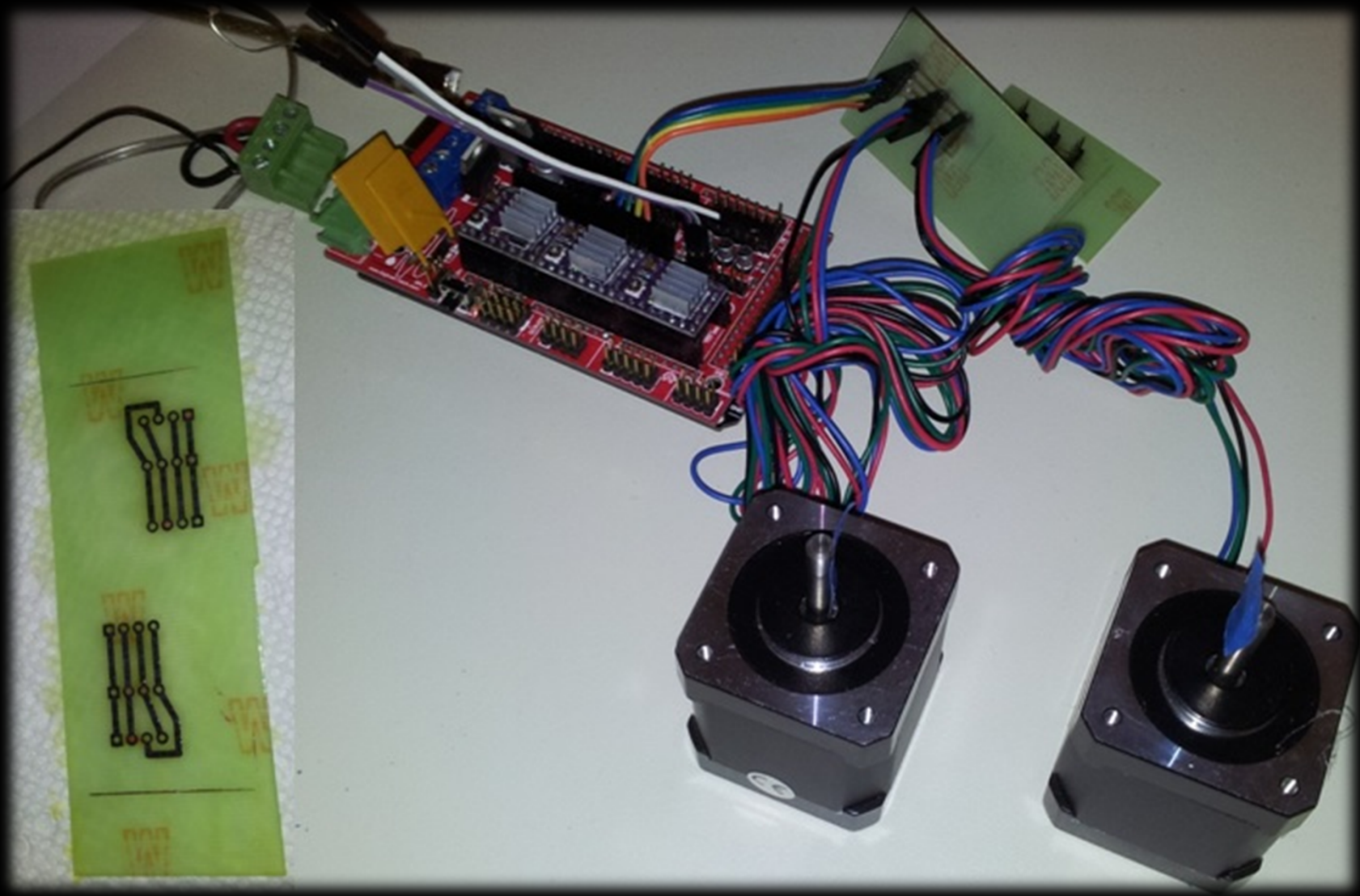


# DIY Parts

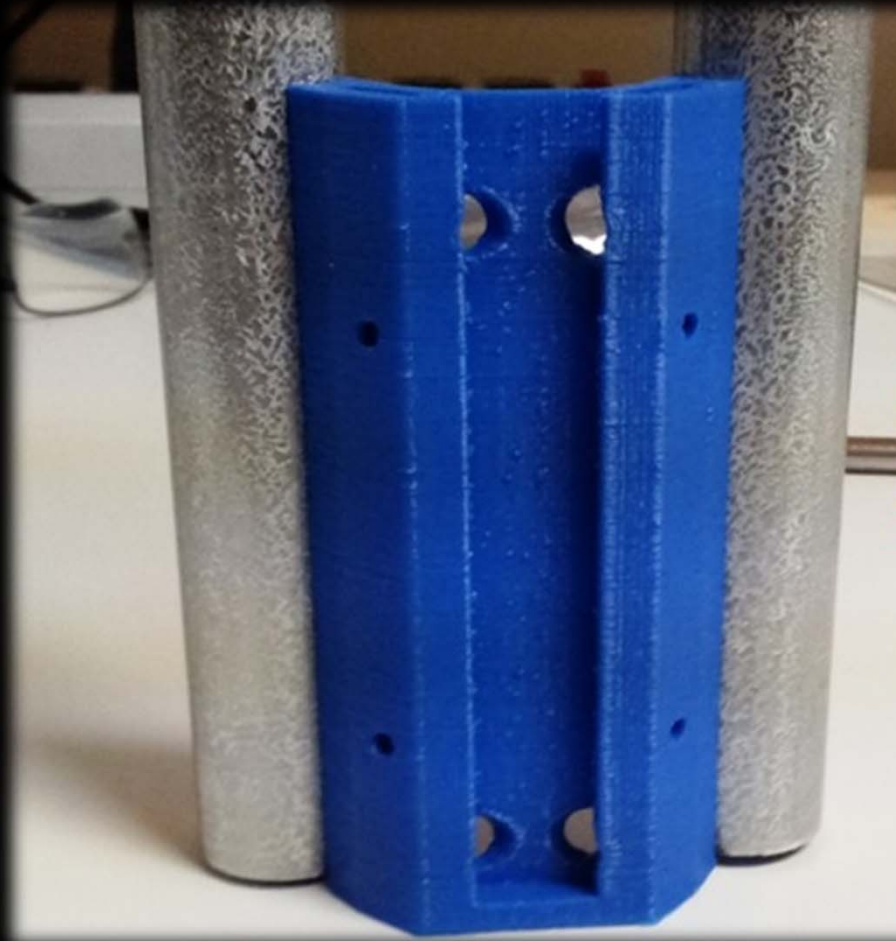




# MPCNC Control



# Universal Tool Holder

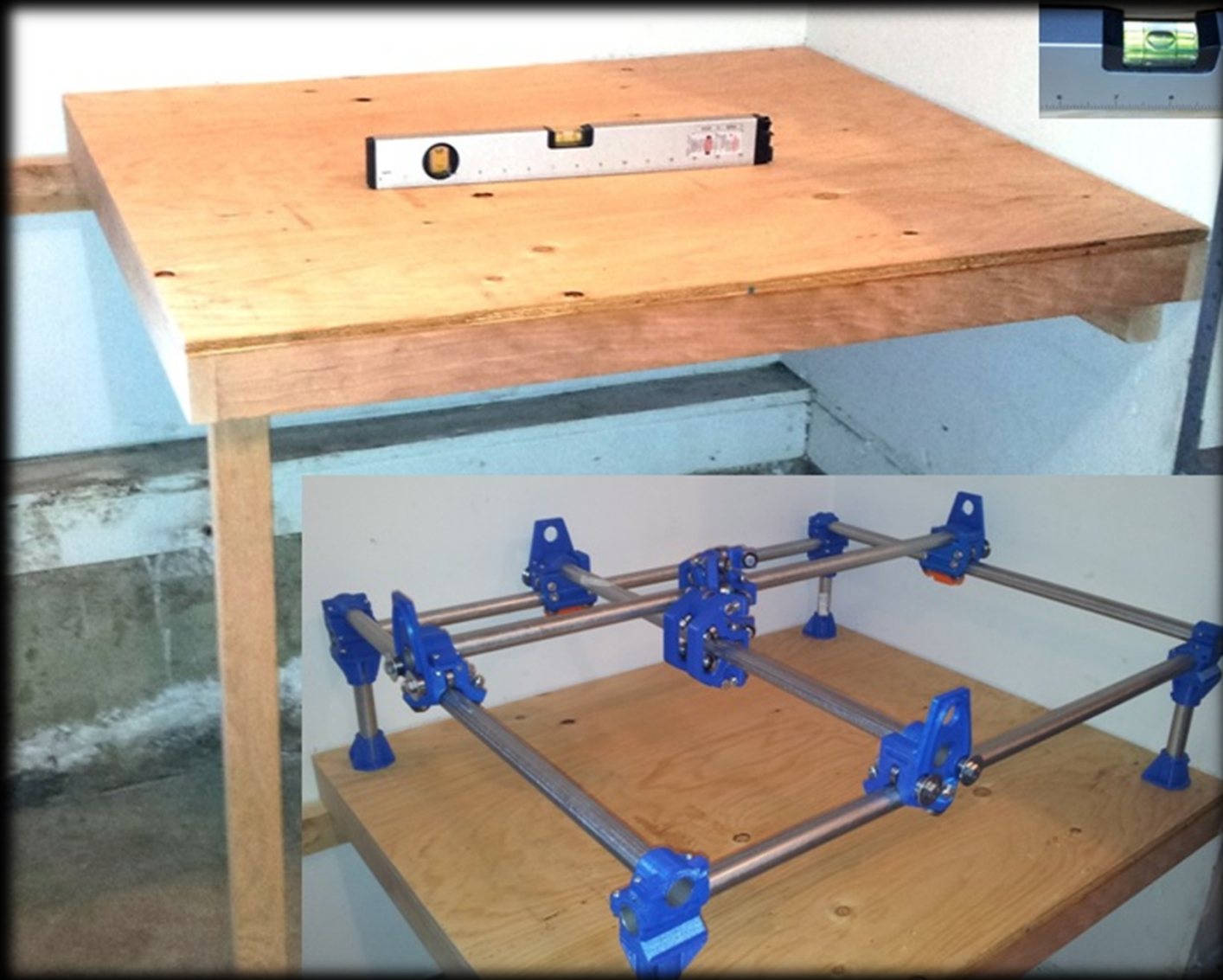




# MPCNC Spindle



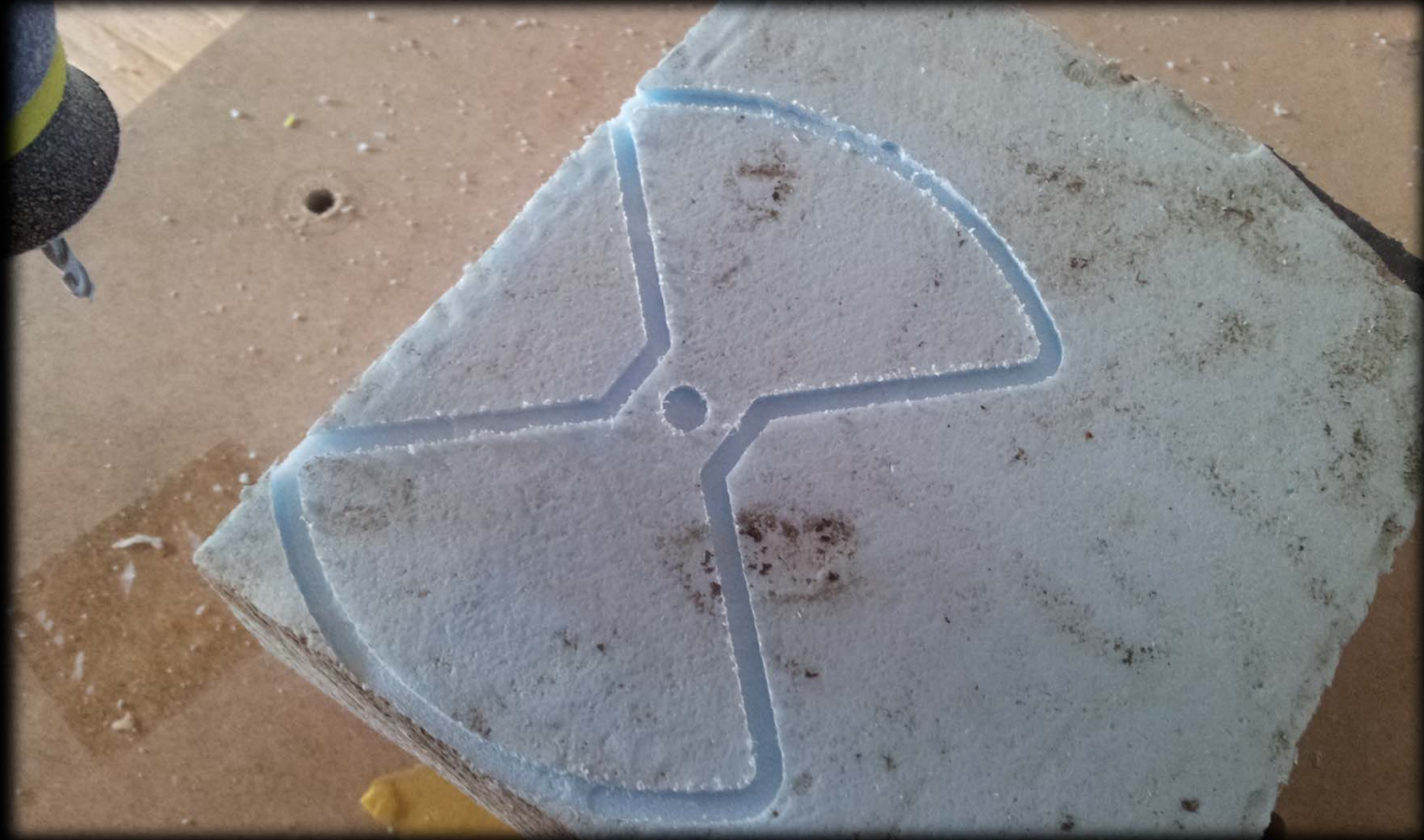
# Work Table



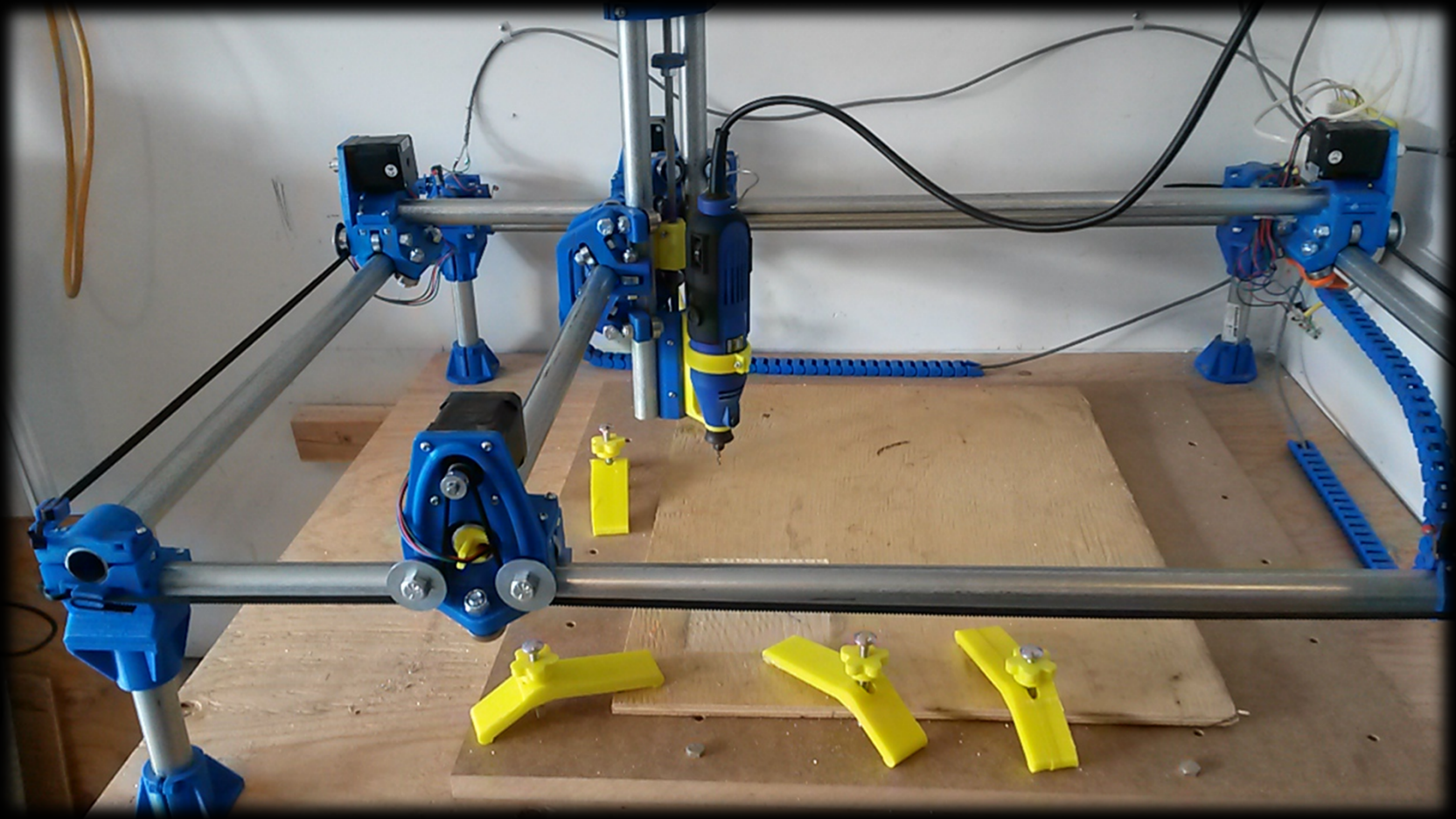
# First "Cut" – 2D Plot



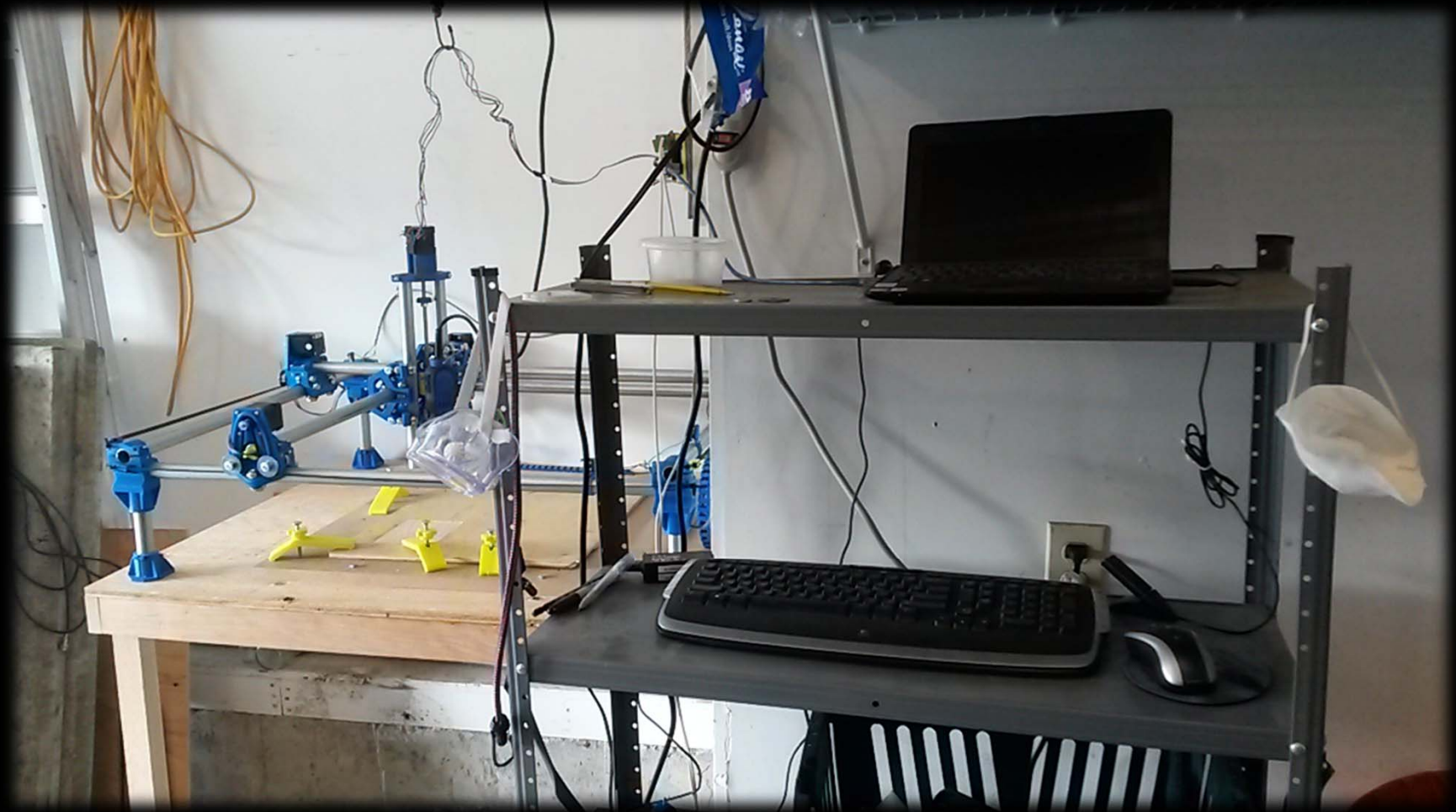
# Second Cut – Styrofoam Block



# Finished CNC Machine



# Garage Workshop



Making It Up

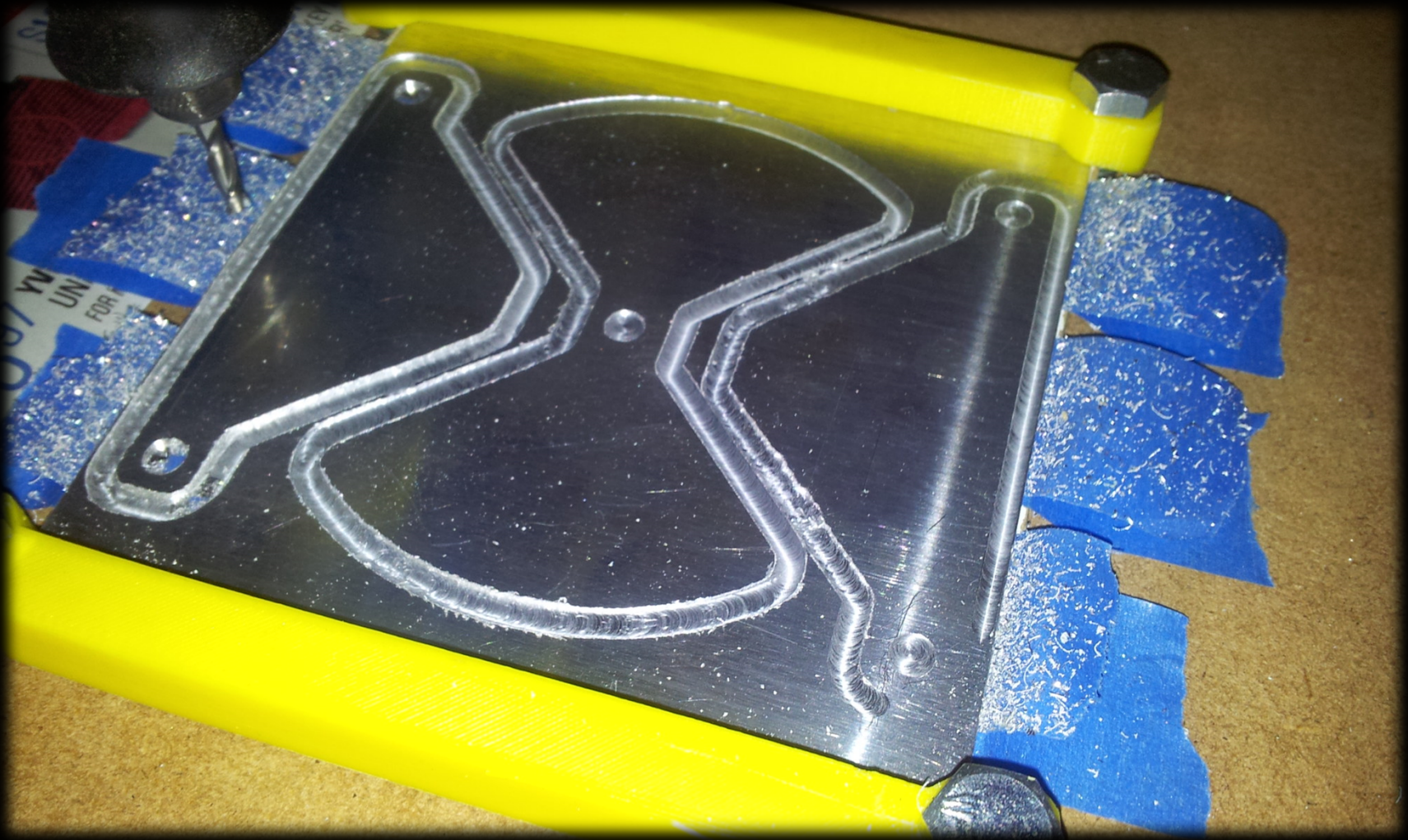
# **BUTTERFLY CAPACITOR**

# Milling Plates 0.025" Aluminum

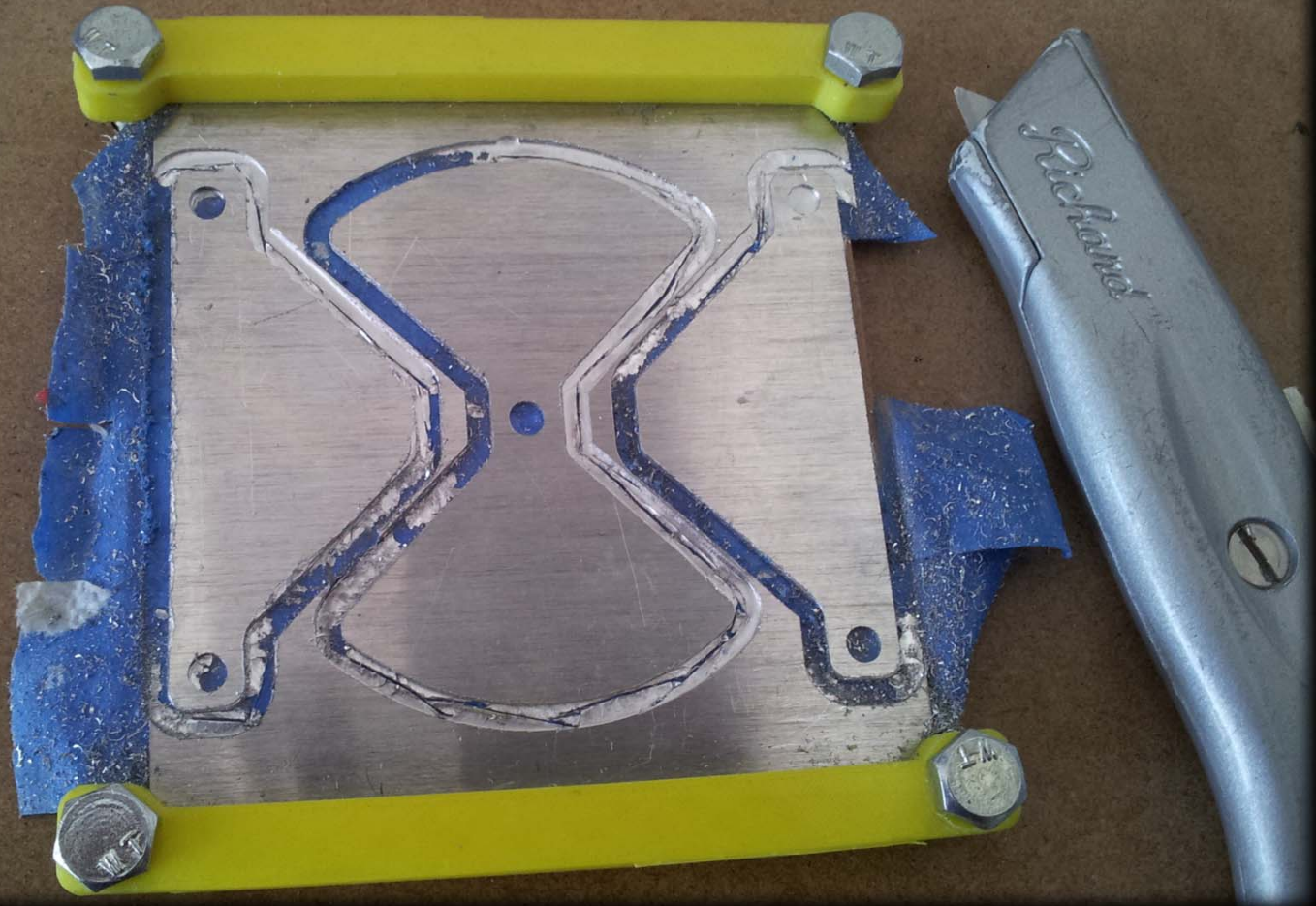




# Milling Plates



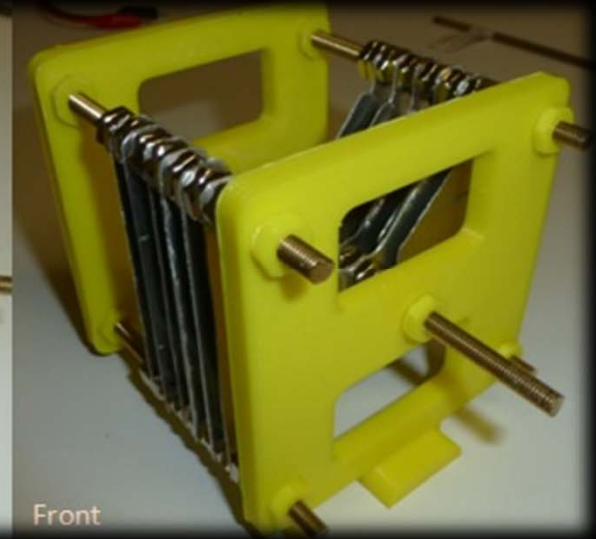
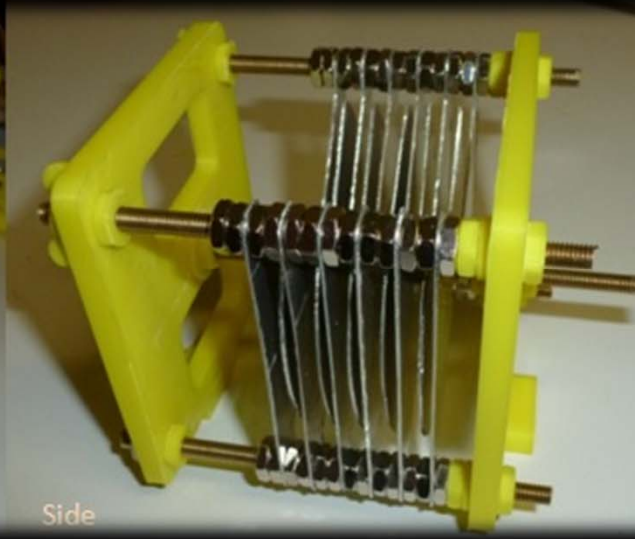
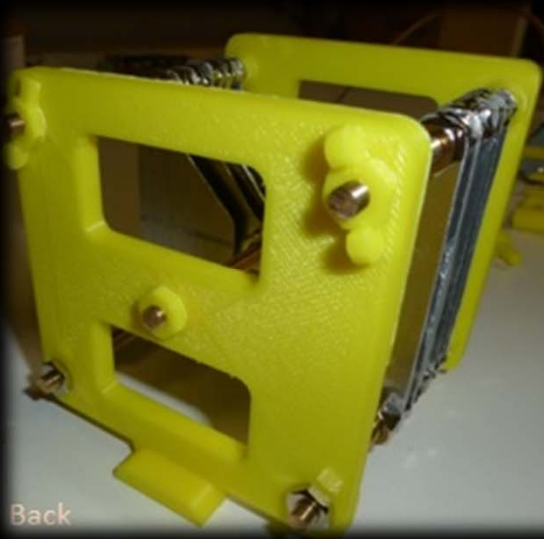
# Milling Plates



# Trimming Plates



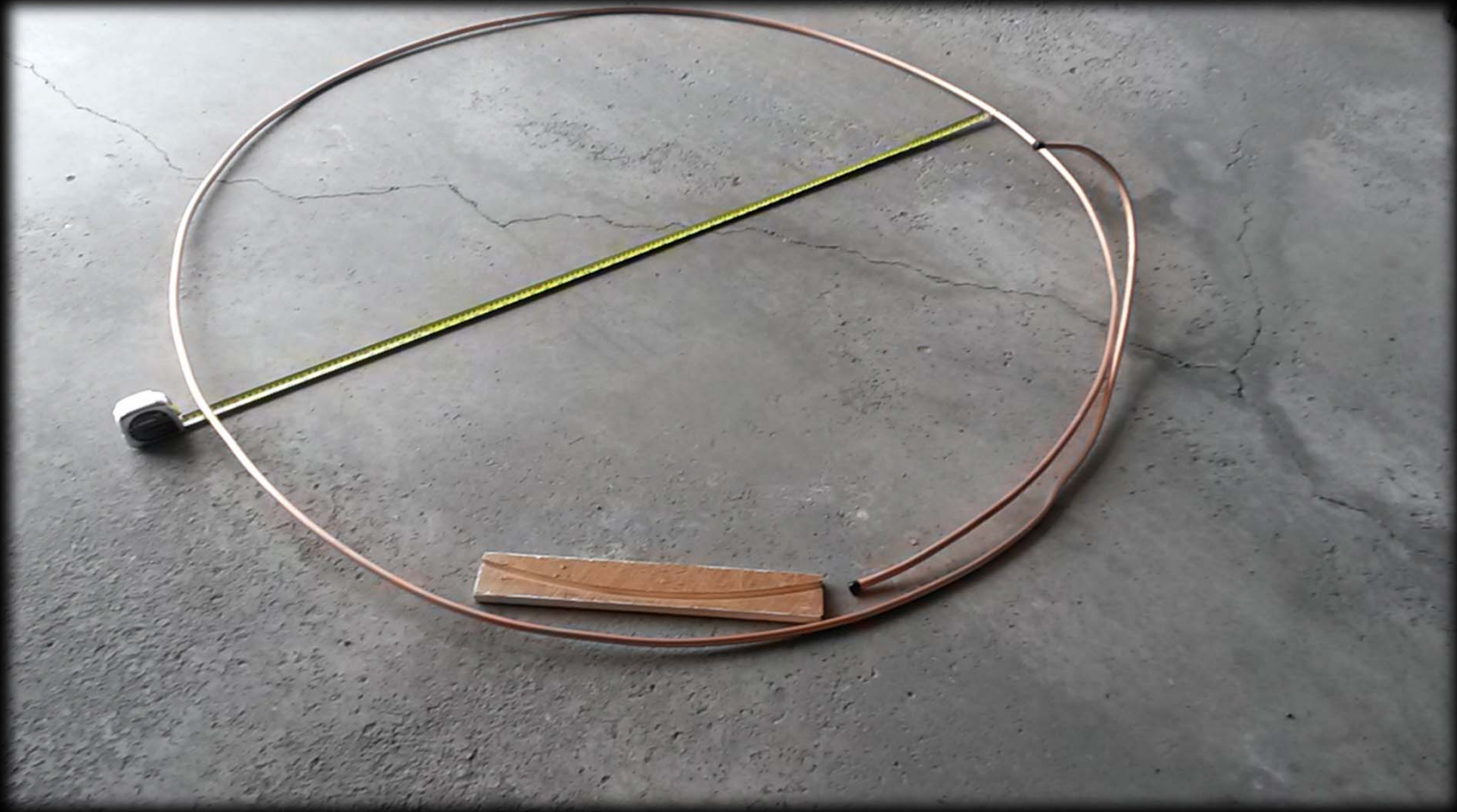
# Assembling Capacitor



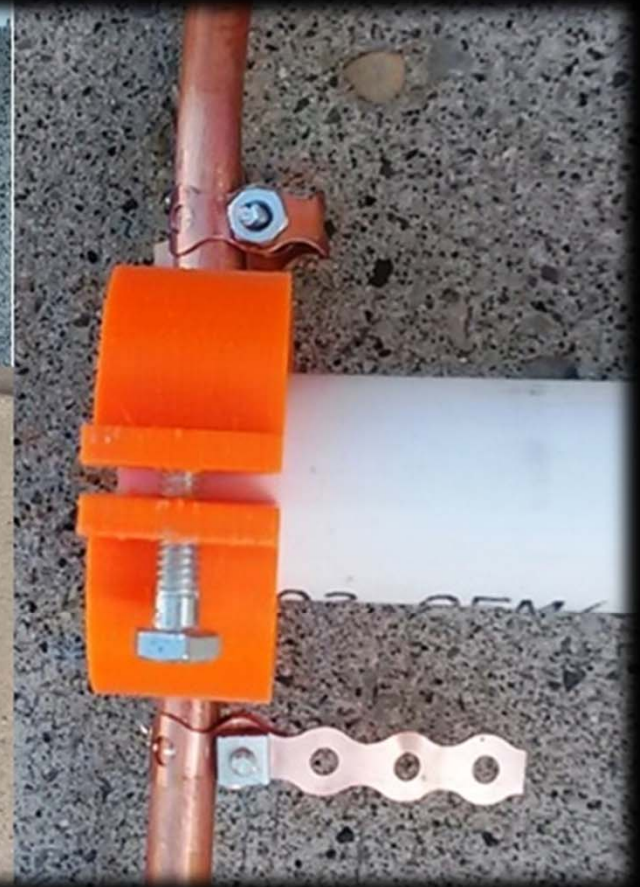
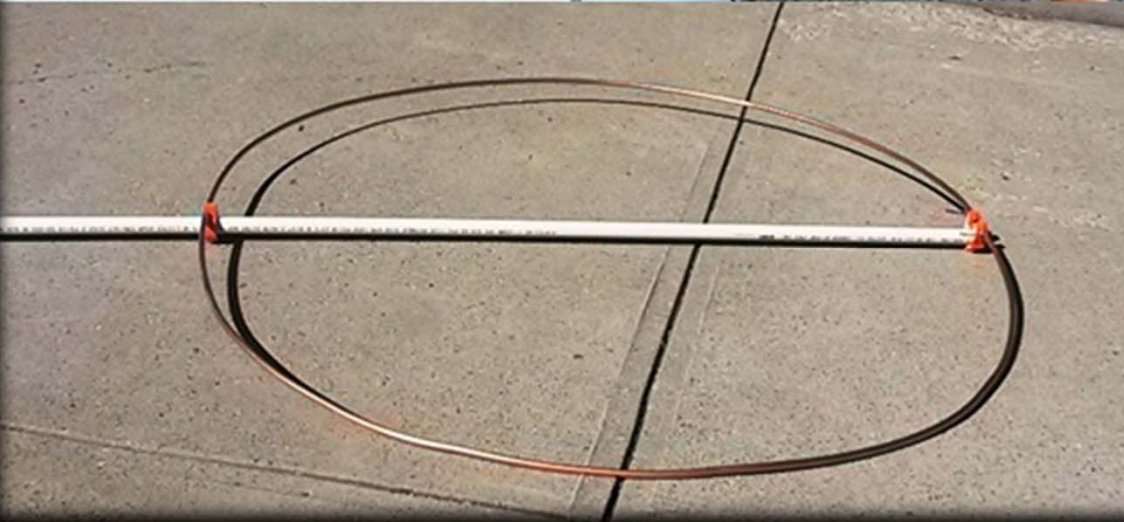
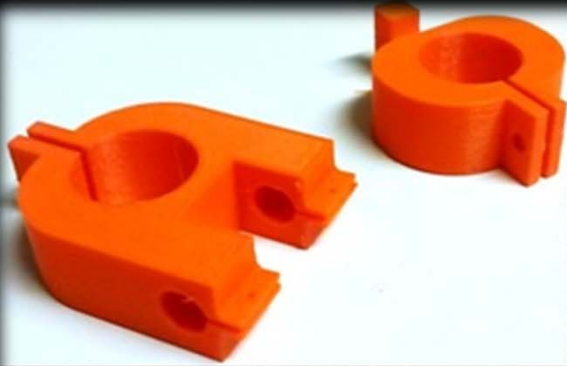
Making It Up

# LOOP CONSTRUCTION

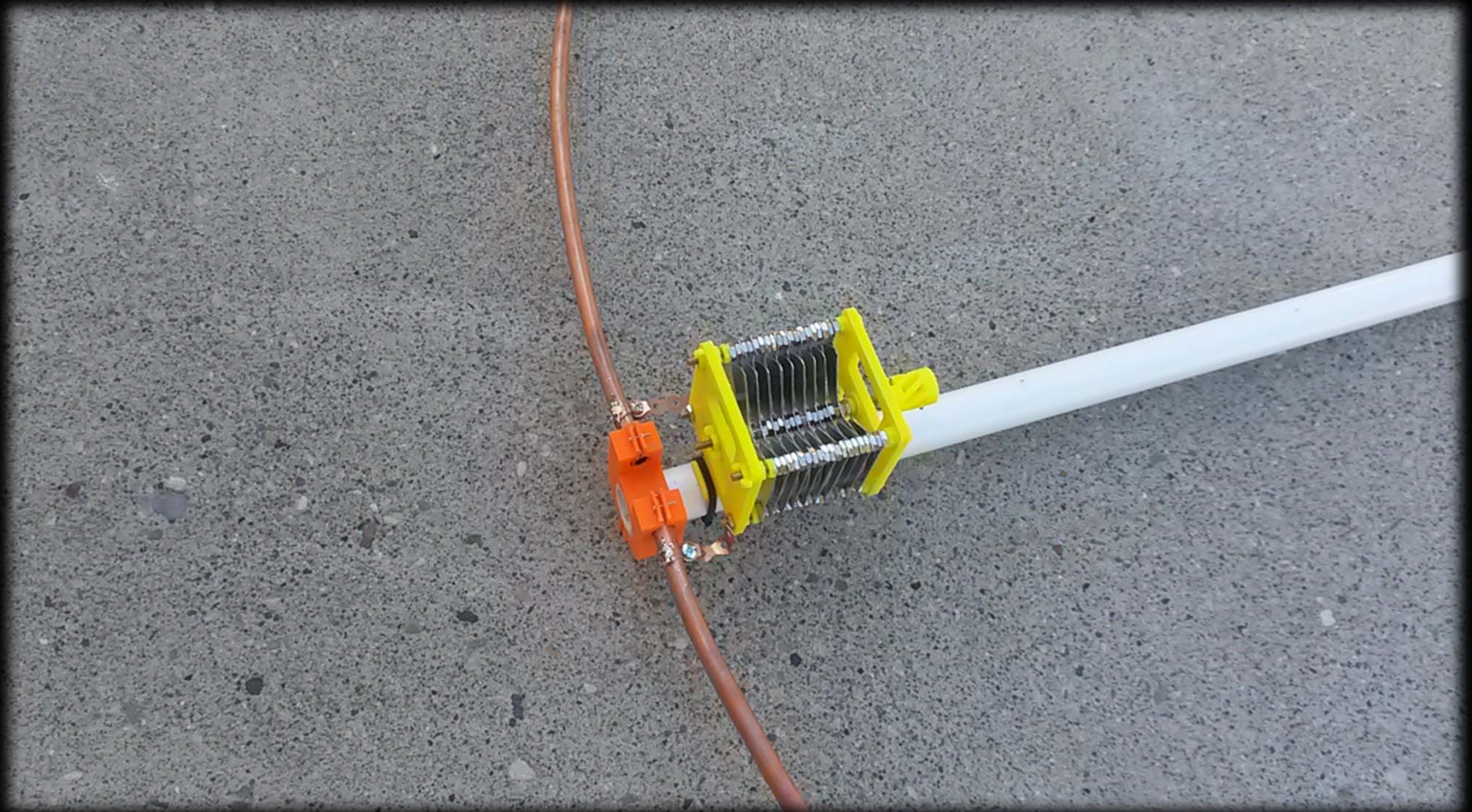
# Forming Loop



# Assembling Loop

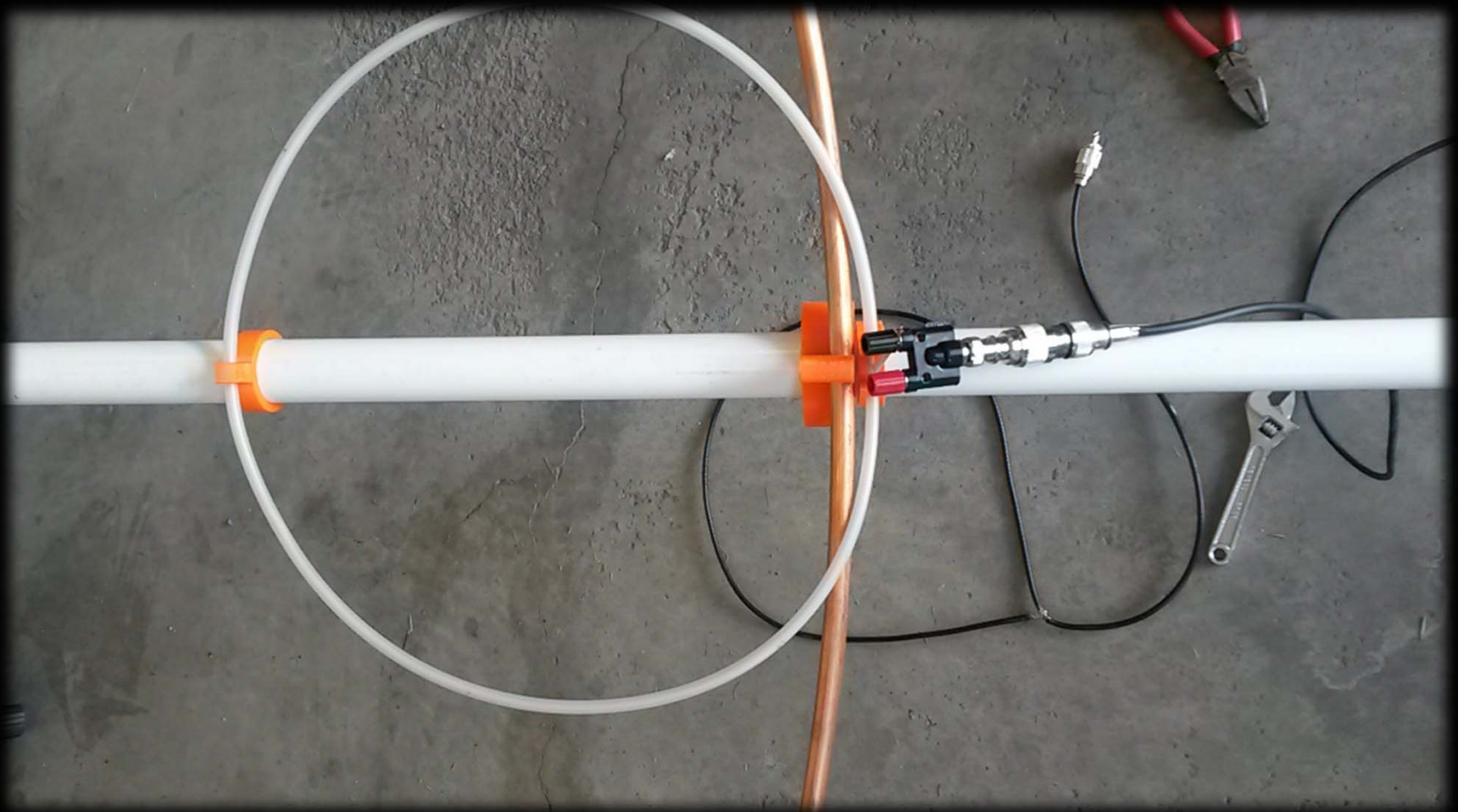


# Attaching Capacitor





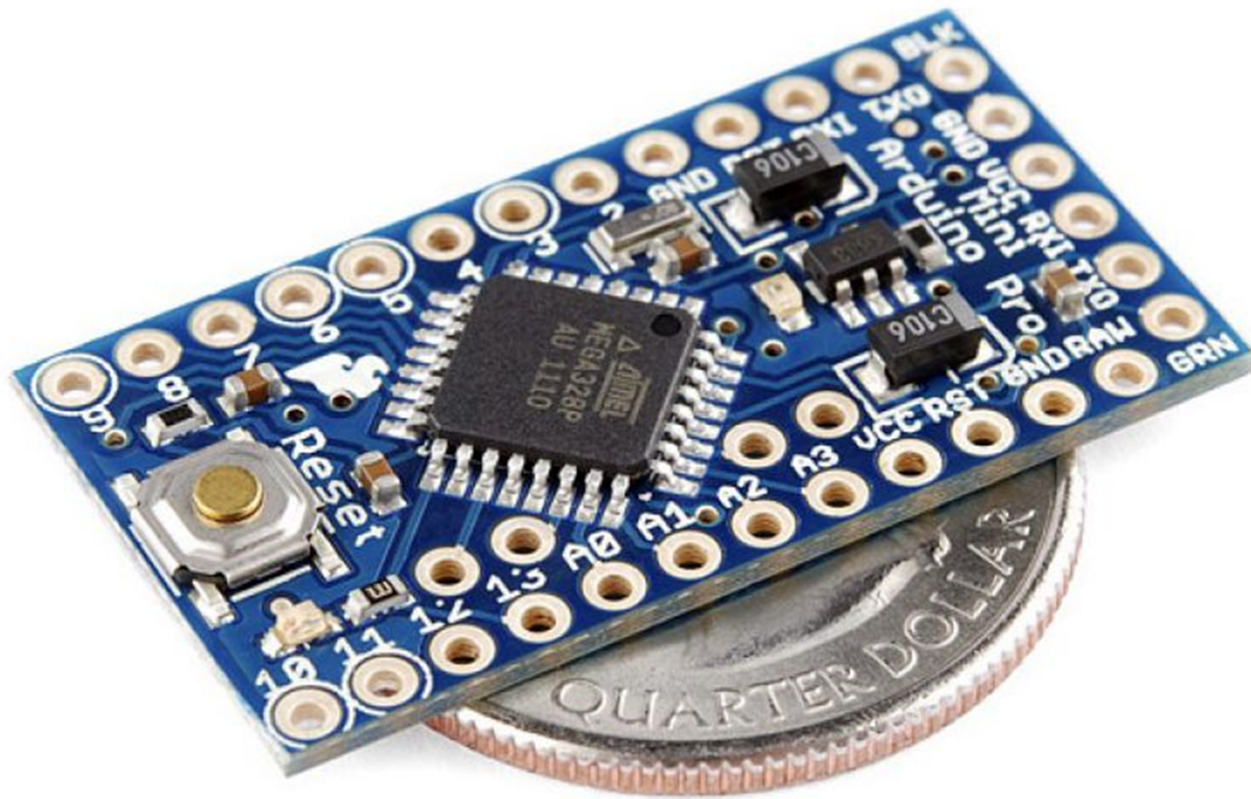
# Feeding the Loop



Making It Up

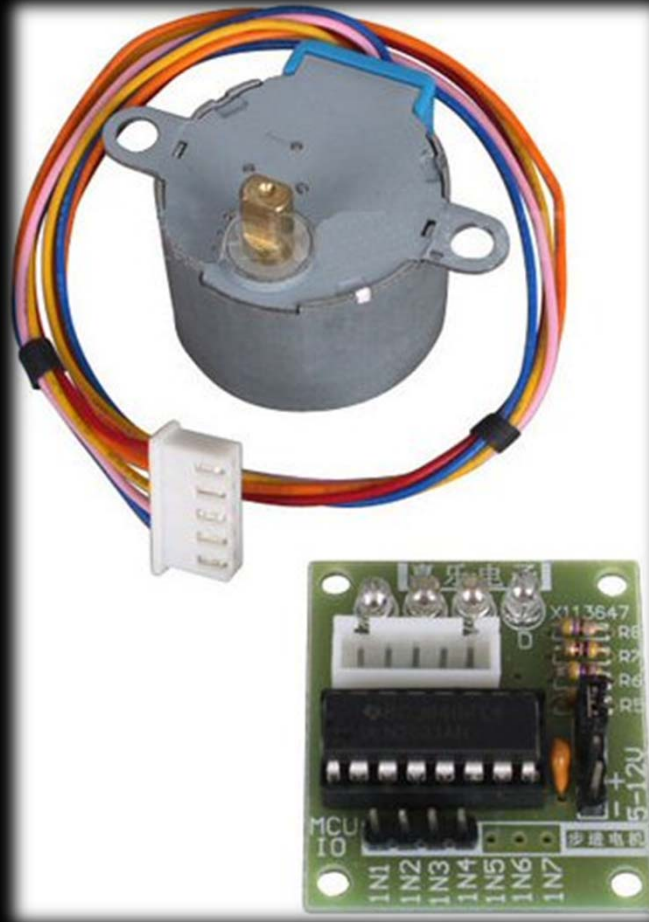
# REMOTE CONTROL

# Arduino Pro Mini 3.3 V



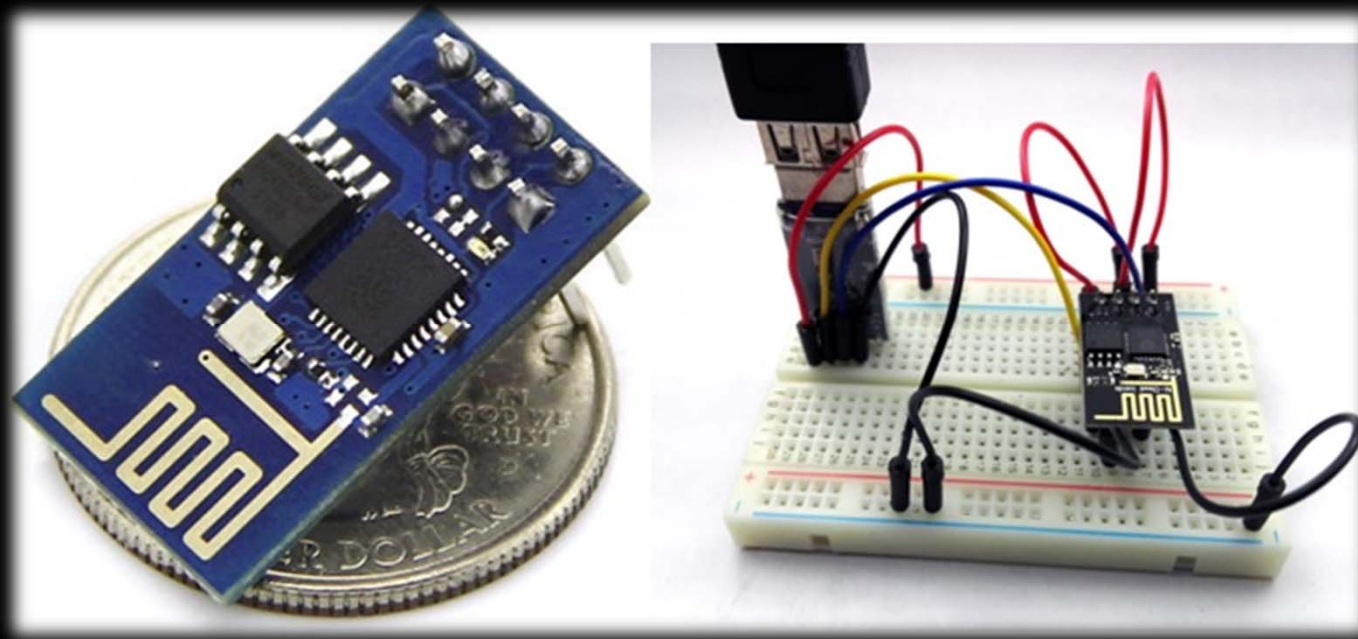
\$3

# Stepper Motor and Controller



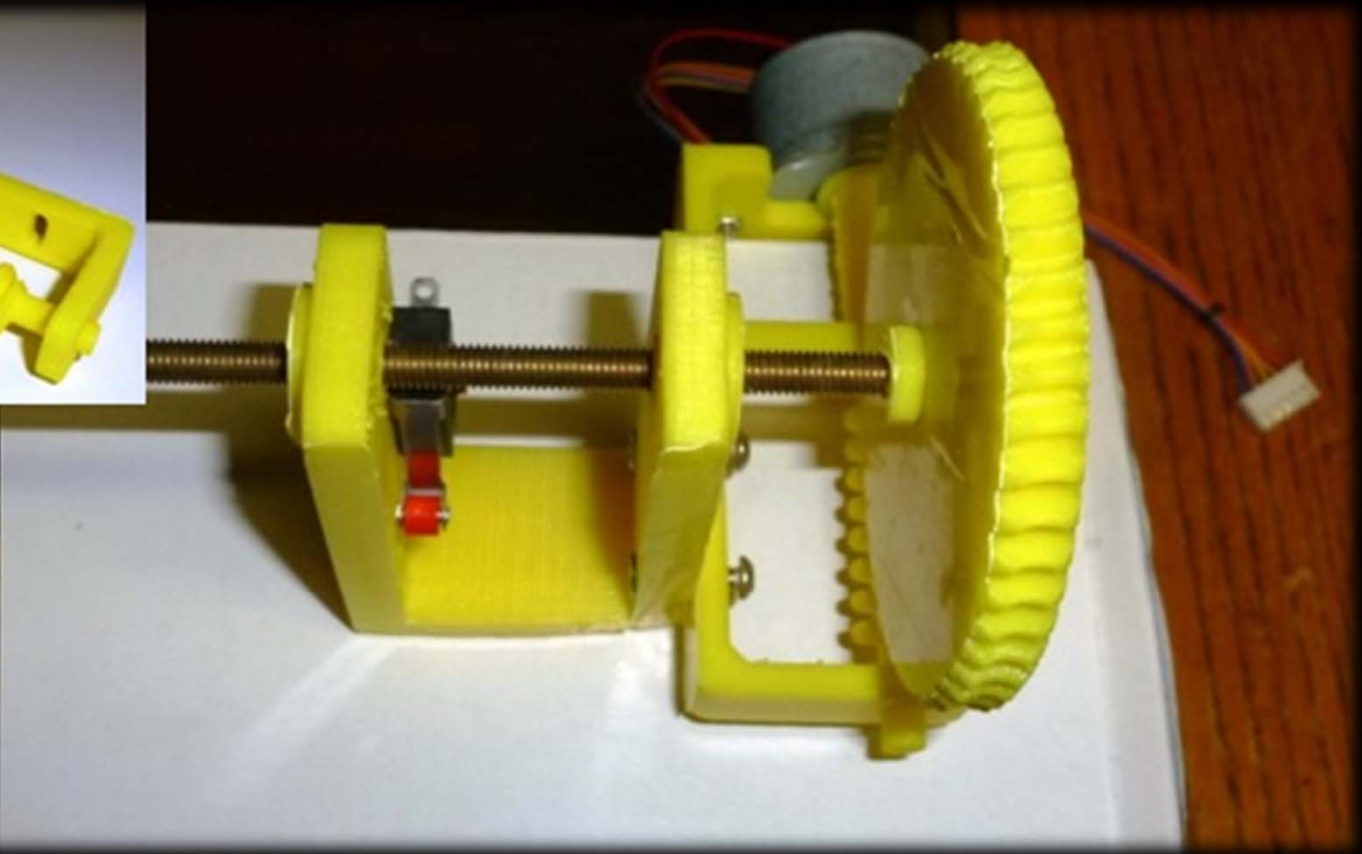
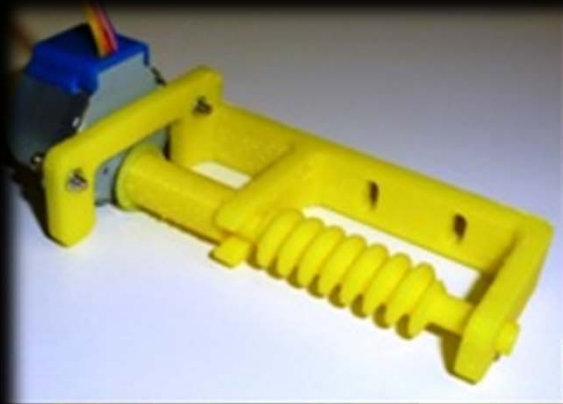
\$3

# ESP8266-01 WiFi Module



\$3

# Motor Mount and Gears



# Writing Software

The image shows a screenshot of Microsoft Visual Studio with an Arduino project named 'MagLoopA'. The code in the main editor is as follows:

```
411 //if (rangeError)
412 //{
413 // rangeError = false;
414 // return SERIAL_ERROR_INVALID_COMMAND;
415 //}
416 if (serialCommandIndex == setMoveUp)
417 {
418     motor.move(stemp);
419     Report(M_MON, "Command U=" + String(stemp, DEC));
420 }
421 else
422 {
423     motor.move(-stemp);
424     Report(M_MON, "Command D=" + String(stemp, DEC));
425 }
426 ChangeState(STATE_RUNNING);
427 Timer1.start();
428 break;
429
430 case setGoTo:
431     if (sysState != STATE_IDLE)
432     {
433         return SERIAL_ERROR_BUSY;
434     }
435     if ((ParamValue >= QUADRANT_STEPS_MIN) && (ParamValue <= QUADRANT_STEPS_MAX))
436     {
437         Report(M_MON, "Command G=" + String(ParamValue, DEC));
438         Timer1.start();
439         ChangeState(STATE_SEEKING);
440         motor.moveTo(ParamValue);
441     }
442     break;
443
444 case setStop:
445     if ((sysState == STATE_RUNNING) || (sysState == STATE_SEEKING))
446     {
447         StopFlag = true;
448     }
449     break;
450
451 case getPosition:
452     if (sysState == STATE_IDLE)
```

The Serial Monitor window on the right shows the following output:

```
Serial | COM7
Opening port
Port open
Mon: Command +

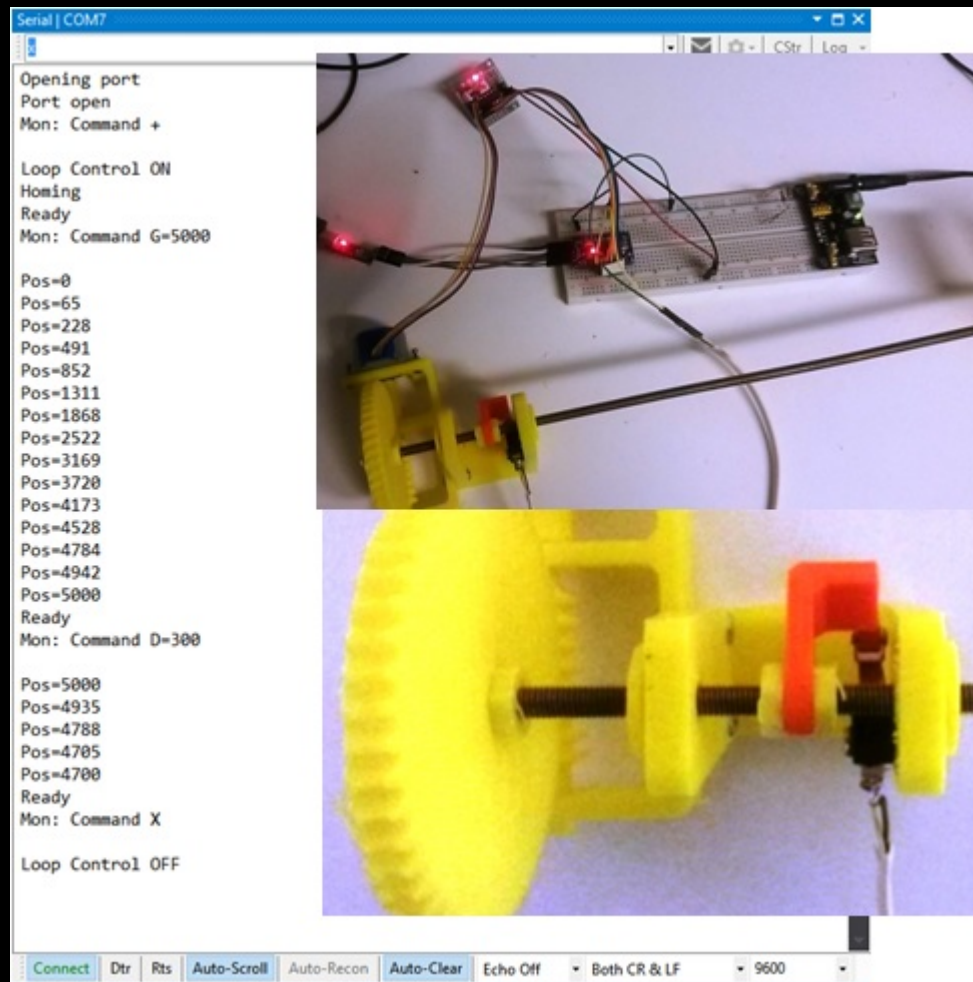
Loop Control ON
Homing
Ready
Mon: Command G=5000

Pos=0
Pos=65
Pos=228
Pos=491
Pos=852
Pos=1311
Pos=1868
Pos=2522
Pos=3169
Pos=3720
Pos=4173
Pos=4528
Pos=4784
Pos=4942
Pos=5000
Ready
Mon: Command D=300

Pos=5000
Pos=4935
Pos=4788
Pos=4705
Pos=4700
Ready
Mon: Command X

Loop Control OFF
```

# First Test on Serial Port



The screenshot displays a serial terminal window titled "Serial | COM7" with the following text output:

```
Opening port
Port open
Mon: Command +

Loop Control ON
Homing
Ready
Mon: Command G=5000

Pos=0
Pos=65
Pos=228
Pos=491
Pos=852
Pos=1311
Pos=1868
Pos=2522
Pos=3169
Pos=3720
Pos=4173
Pos=4528
Pos=4784
Pos=4942
Pos=5000
Ready
Mon: Command D=300

Pos=5000
Pos=4935
Pos=4788
Pos=4705
Pos=4700
Ready
Mon: Command X

Loop Control OFF
```

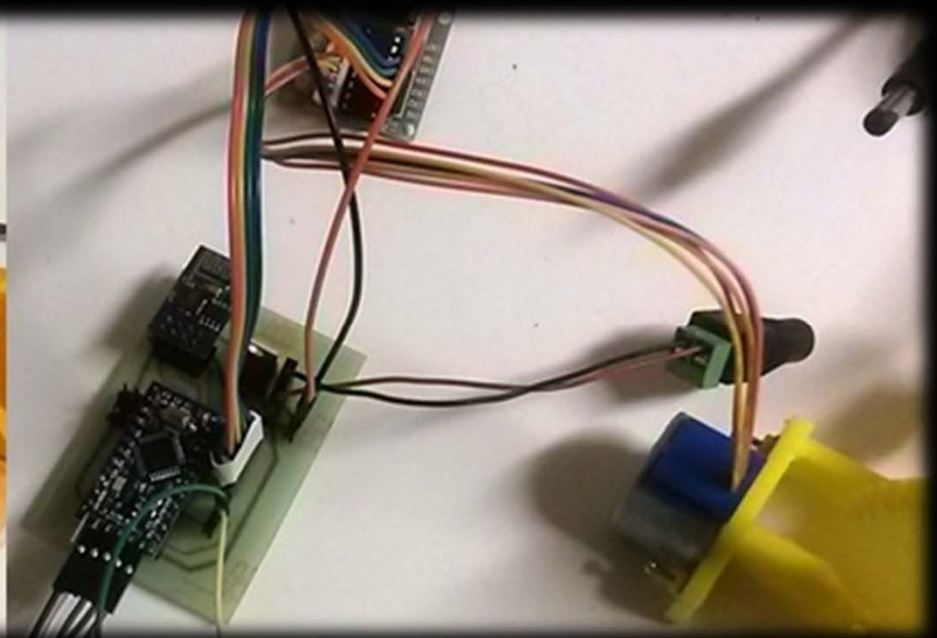
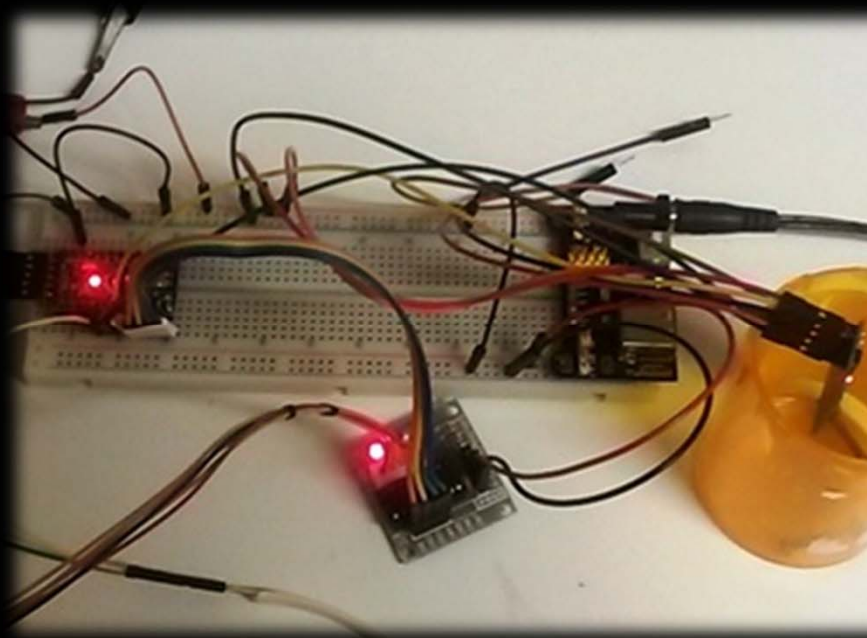
Two photographs are included in the terminal window:

- The top photograph shows a breadboard with a microcontroller, several red LEDs, and a motor assembly connected to it.
- The bottom photograph is a close-up of the motor assembly, featuring a large yellow gear, a smaller yellow gear, and a red component.

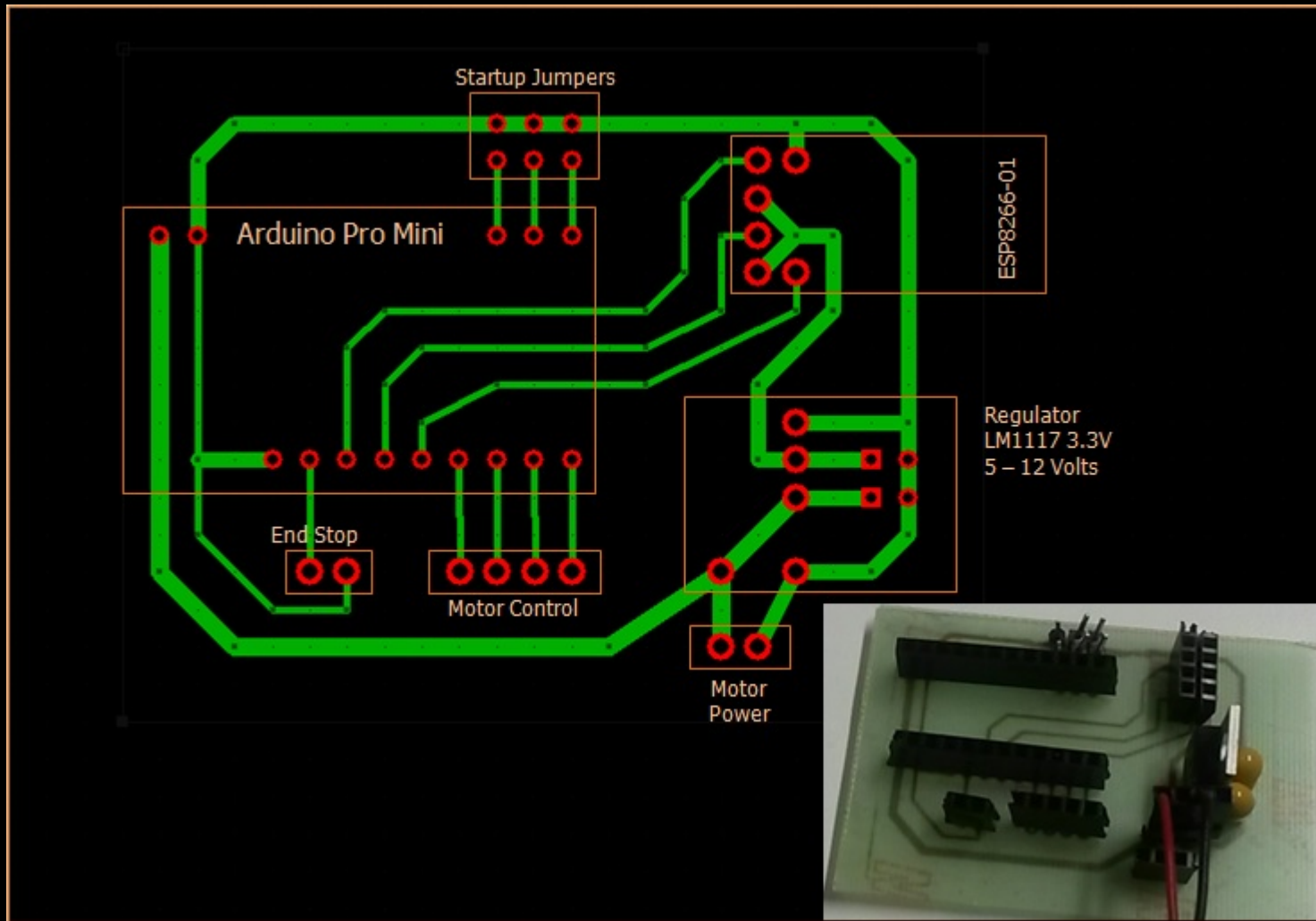
The terminal window includes a status bar at the bottom with the following settings: **Connect**, **Dtr**, **Rts**, **Auto-Scroll**, **Auto-Recon**, **Auto-Clear**, **Echo Off**, **Both CR & LF**, and **9600**.



# Electronics



# Control Board



Making It Up

# **LOOP PERFORMANCE**

# First Test



# Demonstration Videos

- HF Span
  - Shows the loop resonance point moving across spectrum as motor advances
- 20 meter DX
  - Compares reception on loop versus beam with European DX

# Loop Specs

- Main Loop 5' diameter, ½" copper tube
- Feed loop 1' diameter, #18 wire
- Butterfly Capacitor 5-60 pF, 5 KV
- SWR < 1.5:1
- Frequency Range 8 – 18 MHz
- Efficiency compared to full size dipole (theory)
  - 30 meters 60% (down 2.2 dB) [BW=20 kHz]
  - 20 meters 85% (down 1.0 dB) [BW=50 kHz]
  - 17 meters 90% (down 0.5 dB) [BW=150 kHz]

# Remote Control Specs

- Wired (Serial Port) or Wireless (using TCP/IP Client or App)
- WiFi Modes – Home Network (Station) or Stand Alone (Access Point)
- Motor Gear Reduction 64:1
- Worm Gear Reduction 50:1
- Tuning Range 0 – 50,000 steps (560 steps = 1 degree)
- Automatic Homing at Startup (End Stop Switch)

# Challenges

- Gear train backlash
  - Can correct in software
- Top Heavy 5' loop
  - Will need some guys
- Permanent Power Supply (5-12V)
  - Too much current draw for battery
- 40 meter coverage
  - Need another 40 pF on capacitor



# Making It Up

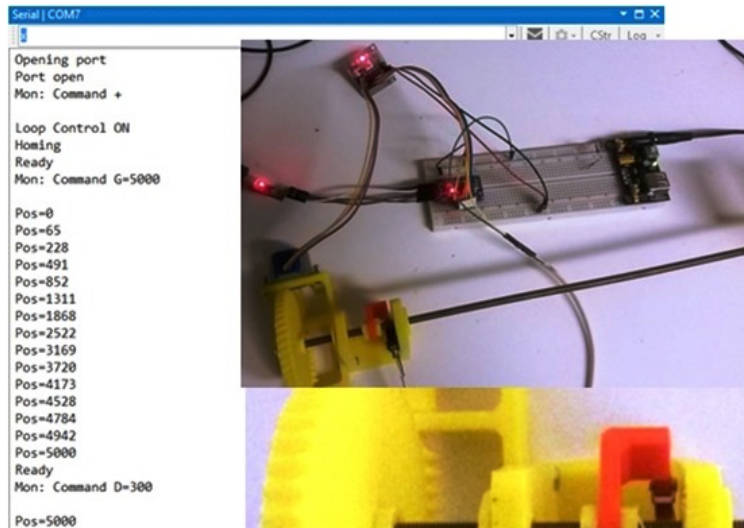
ENJOYING RADIO AND MAKER HOBBIES

[Home](#) » [Projects](#) » [Electronics Projects](#) » Serial Loop Control Running

SEPTEMBER 12, 2016

## Serial Loop Control Running

No Comments



### Recent Posts

[From Breadboard to Circuit Board](#)

[Calgary's Green Summer](#)

[ESP8266-01 Wifi Module](#)

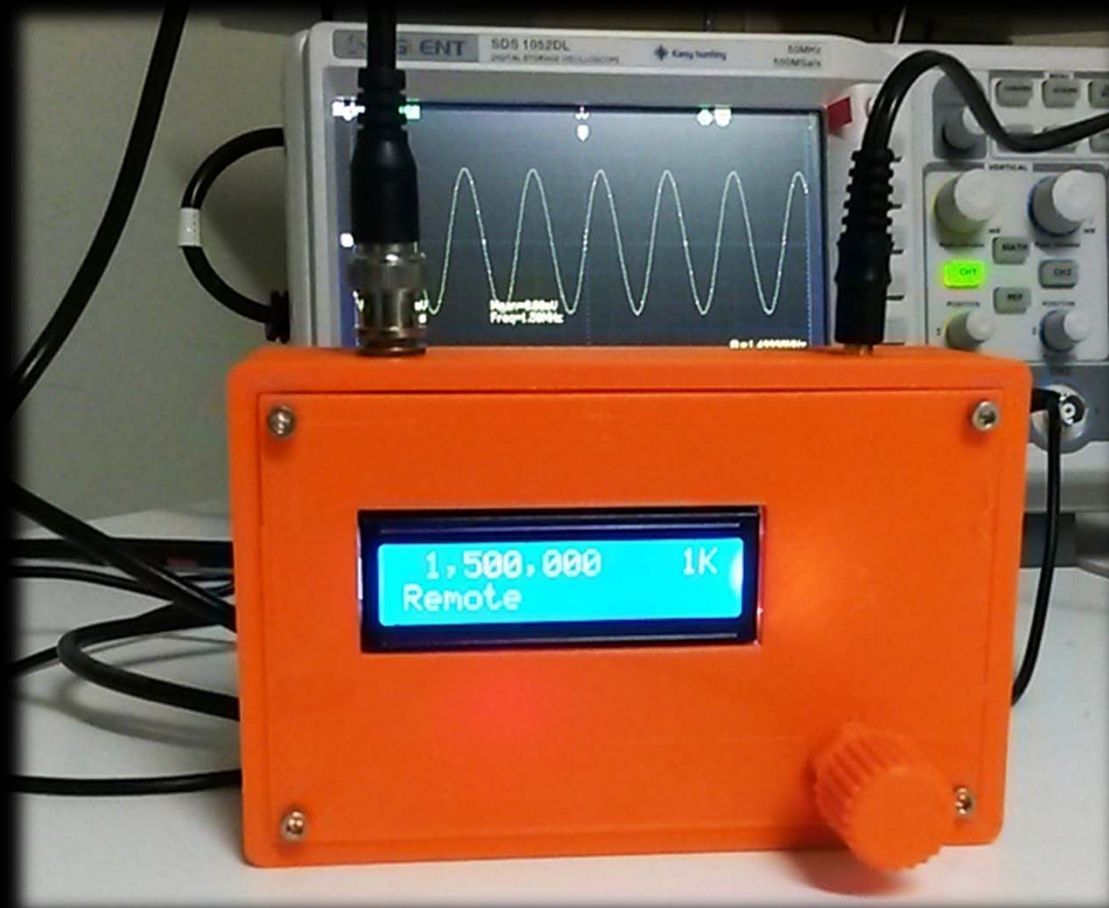
[Serial Loop Control Running](#)

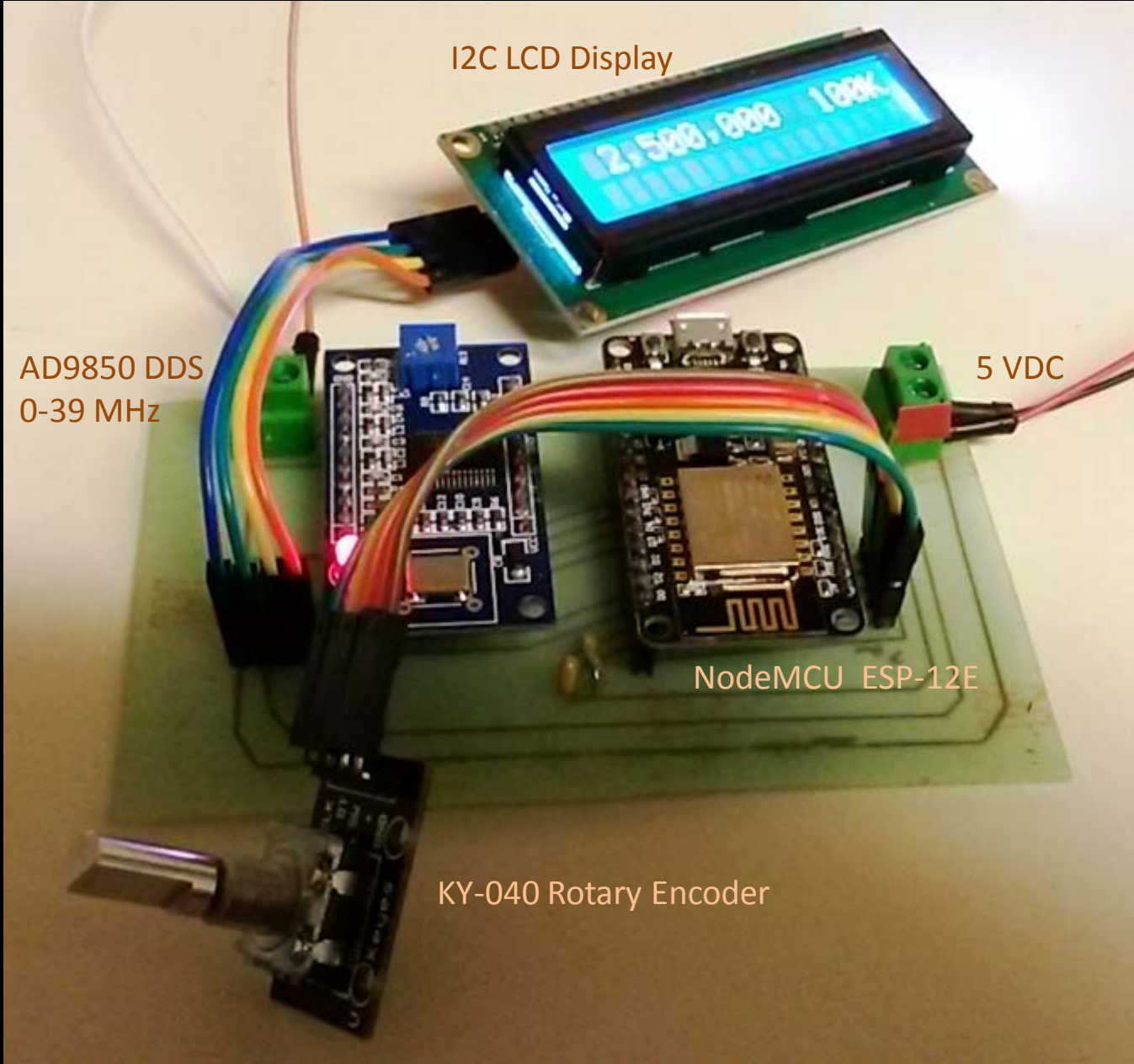
[Shaw Internet Speed Gets Faster](#)



<http://play.fallows.ca>

# Arduino Signal Generator





I2C LCD Display

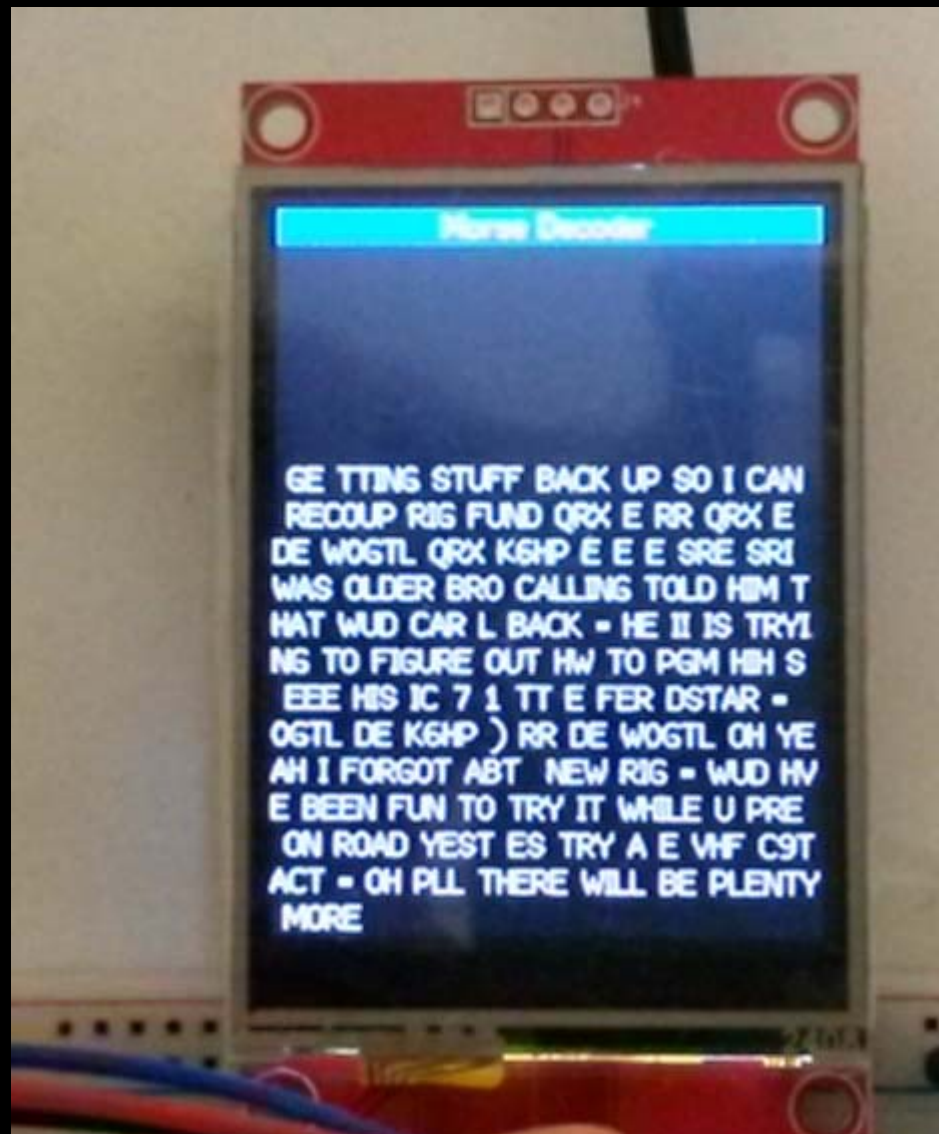
AD9850 DDS  
0-39 MHz

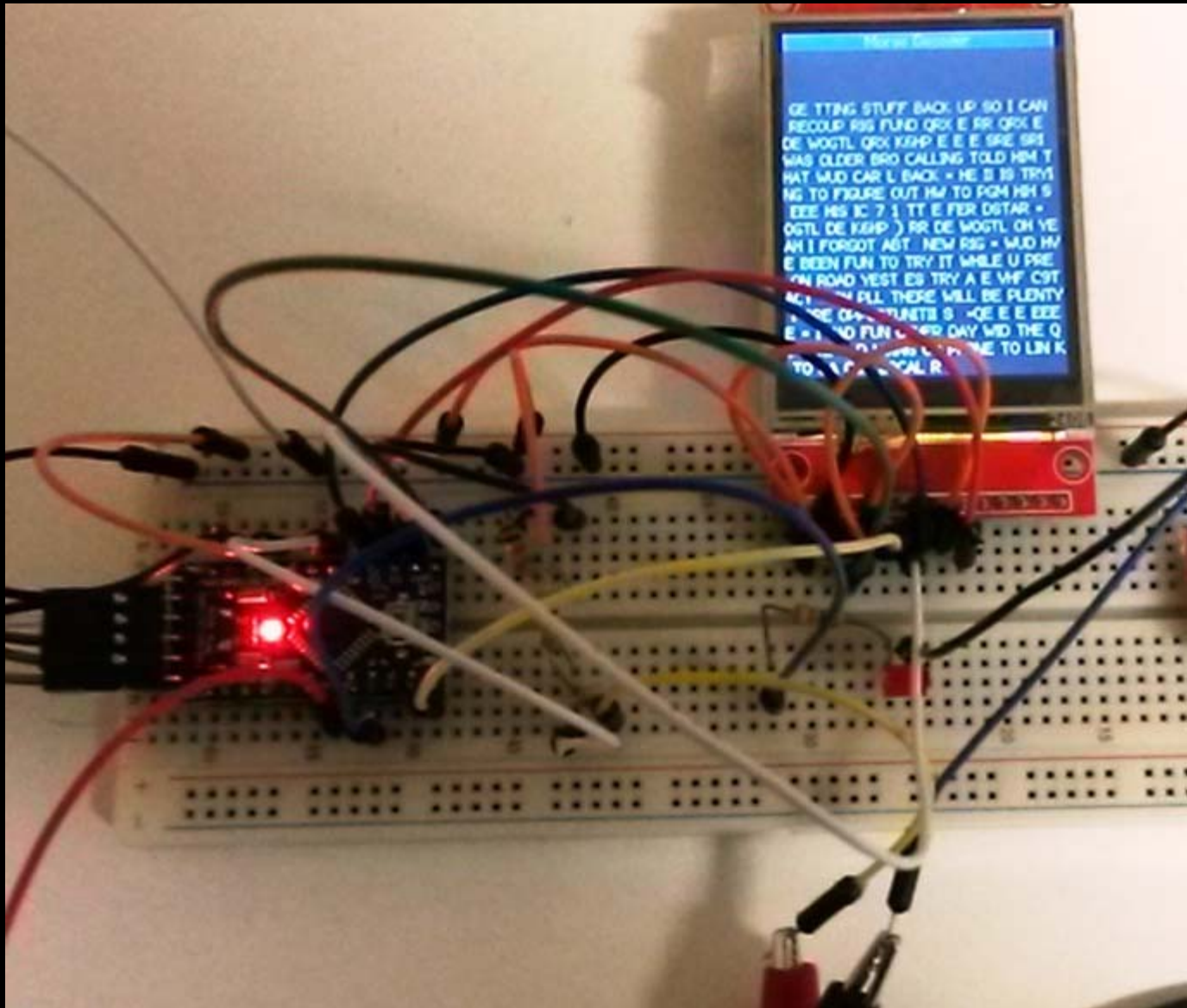
5 VDC

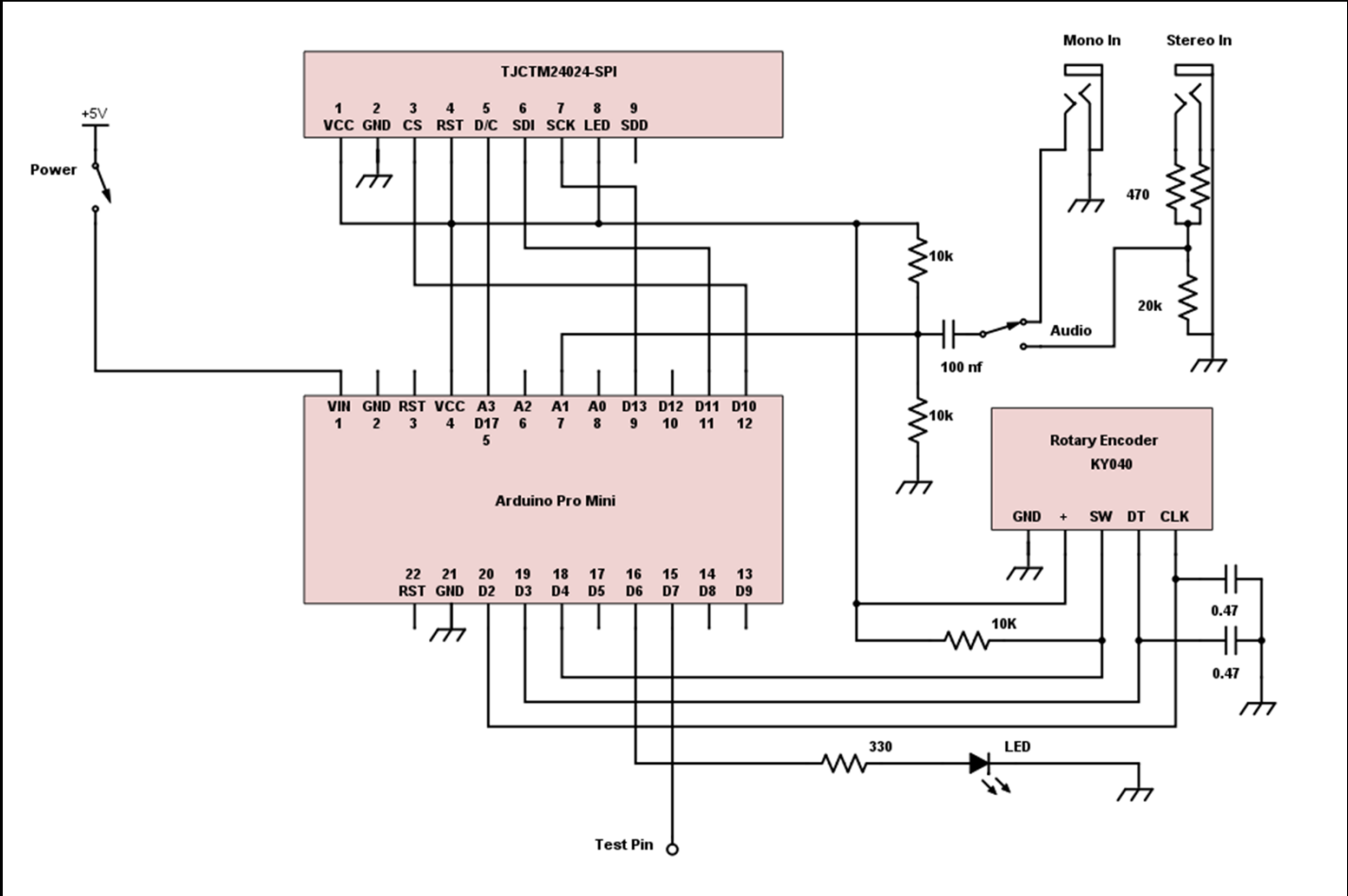
NodeMCU ESP-12E

KY-040 Rotary Encoder

# CW Decoder

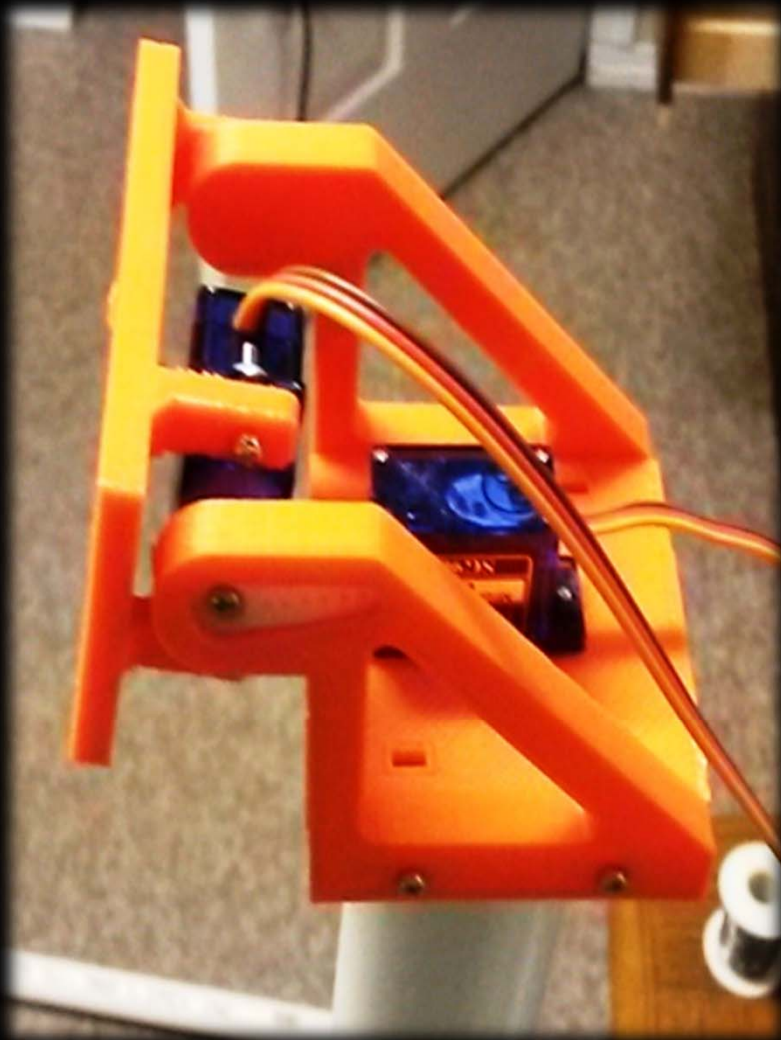






# Recent Project – RX Loop







<http://play.fallows.ca>

