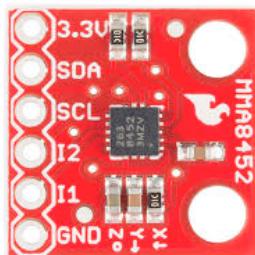


Brett Ratner

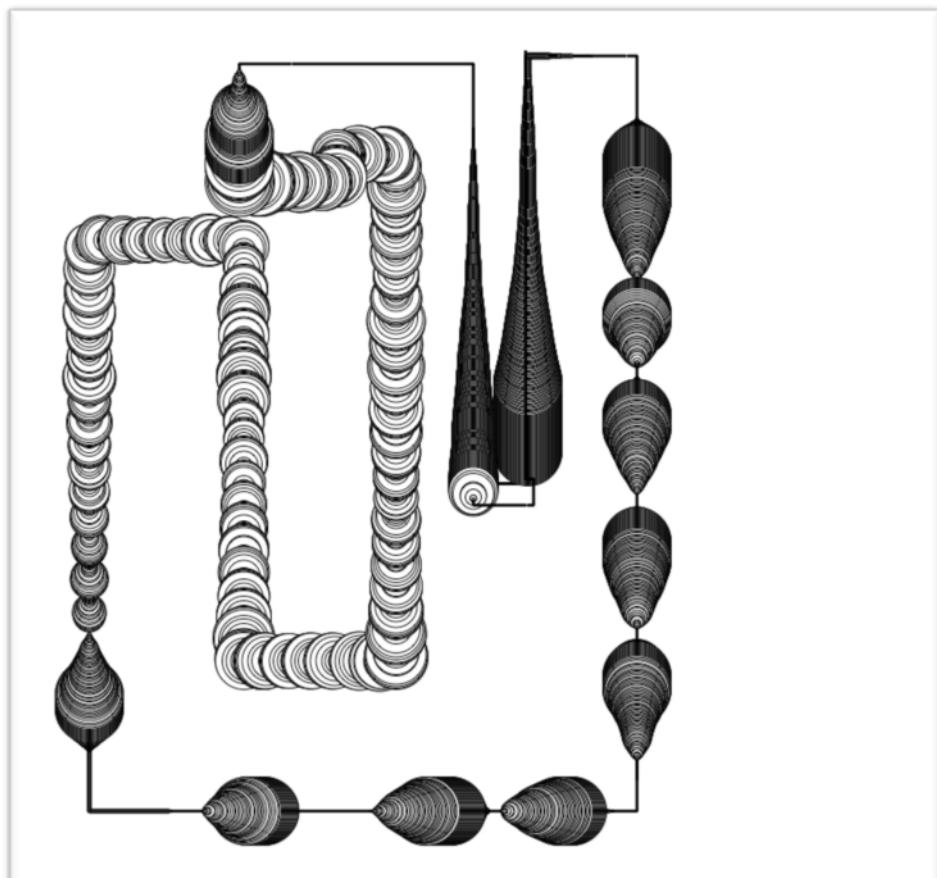
Etch A Sketch

My Project is a cooler more advanced etch a sketch. You use the accelerometer to move draw with the etch a sketch. You tilt it down to draw down, you tilt it up to draw up, you tilt it right to draw right and you tilt it left to draw left. Then in your other hand or in the hands of a second person is the Slide Potentiometer that will adjusts the size of the circle that you are drawing with.

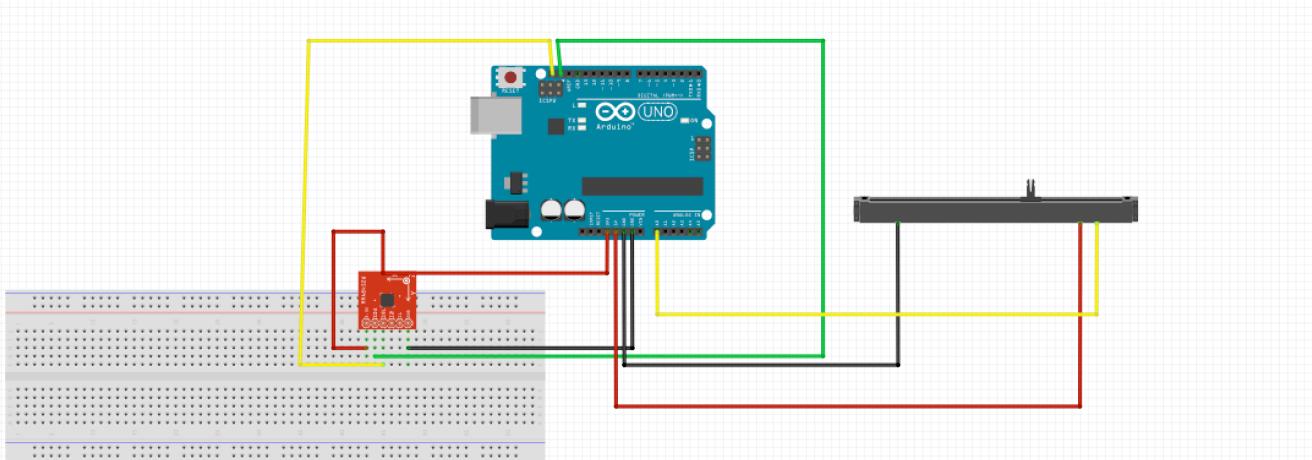
Sensors:



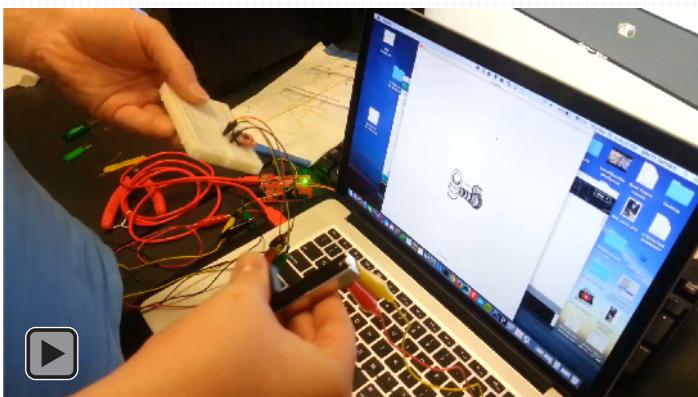
Example Drawing:



Fritzing Diagram:



fritzing



Arduino Code:

```
#include <Wire.h>
#include <SFE_MMA8452Q.h>

MMA8452Q accelerometer;
void setup() {
  Serial.begin(9600);
  accelerometer.init(SCALE_8G, ODR_100);
}
void loop() {
  int sensorValue = analogRead(A0);
  sensorValue = sensorValue/17;
  Serial.println(sensorValue);
  if (accelerometer.available())
```

```

{
  accelerometer.read();
//  printaccelerometer();
  printOrientation();
  Serial.println();
}
}

// This will print out the x,y, and z axis information for the accelerometer.
void printaccelerometer()
{
  Serial.print("x: ");
  Serial.print(accelerometer.cx, 3);
  Serial.print("\t");
  Serial.print("y: ");
  Serial.print(accelerometer.cy, 3);
  Serial.print("\t");
  Serial.print("z: ");
  Serial.print(accelerometer.cz, 3);
  Serial.print("\t");
}
void printOrientation()
{
if(accelerometer.cy <= 0.150 && accelerometer.cy >= -0.150 && accelerometer.cx <=
0.150 && accelerometer.cx >= -0.150){
//  Serial.print("Flat");
  Serial.write("F");
}
else if(accelerometer.cy < -0.150){
//  Serial.print("Left");
  Serial.write("L");
}
else if(accelerometer.cy > 0.150){
//  Serial.print("Right");
  Serial.write("R");
}
else if(accelerometer.cx > 0.150){
//  Serial.print("backward");
  Serial.write("D");
}
else if(accelerometer.cx < 0.150){
//  Serial.print("foward");
  Serial.write("U");
}
}

```

Processing Code:

```
import processing.serial.*;
Serial myPort;
int val = 1;
int directionX =1;
int directionY =0;
float x=400;
float y=400;
float speed =1;
int incomingByte;

void setup(){
size(800,800);
background(255);
smooth();
frameRate(60);
String portName = Serial.list()[10];
myPort = new Serial(this, portName, 9600);

}

void draw()
{
  x=x+speed*directionX;
  y=y+speed*directionY;
  move();

// check boundaries
if ((x>width) || (x<0))
{
  directionX=0;
}
if ((y>height) || (y<0))
{
  directionY=-0;
}

if( val >=1){

  stroke(0);
  fill(255);
  ellipse(x, y, val, val);
```

```

}

//will make the lowest value 1 instead of 0
else if (val == 0){
    val=1;
    stroke(0);
    fill(255);
    ellipse(x, y, val, val);
}

noStroke();
fill(255,255,255);
rect(375,height-40,60,20);
// Prints inches
fill(0);
text(val + " inches", 375, height - 25);

}

```

```

void move(){
incomingByte = myPort.read();
if (incomingByte == 'L') {
    directionX=-1;
    directionY=0;
}
else if (incomingByte == 'R') {
    directionX=1;
    directionY=0;
}

else if(incomingByte == 'U') {
    directionY=-1;
    directionX=0;
}

else if( incomingByte == 'D') {
    directionY=1;
    directionX=0;
}

else if( incomingByte == 'F') {
    directionY=0;
    directionX=0;
}

}

```

```
}

void keyPressed()
{
    if (key == CODED)
    {

        if (keyCode == LEFT )
        {
            directionX=-1;
            directionY=0;
        }
        else if (keyCode == RIGHT)
        {
            directionX=1;
            directionY=0;
        }
        else if (keyCode == UP)
        {
            directionY=-1;
            directionX=0;

        }
        else if (keyCode == DOWN)
        {
            directionY=1;
            directionX=0;
        }
    }
    erase();
}

void erase(){

if(key == ' ')
{
background(255,255,255);

}
// When we hear something from the serial port
void serialEvent(Serial myPort)
{
```

```
// Read the data until the new line character
String data = myPort.readStringUntil(10);

// If data exists
if(data != null)
{
    // Convert String into an integer
    val = int(trim(data));
    println(data);
}

}
```