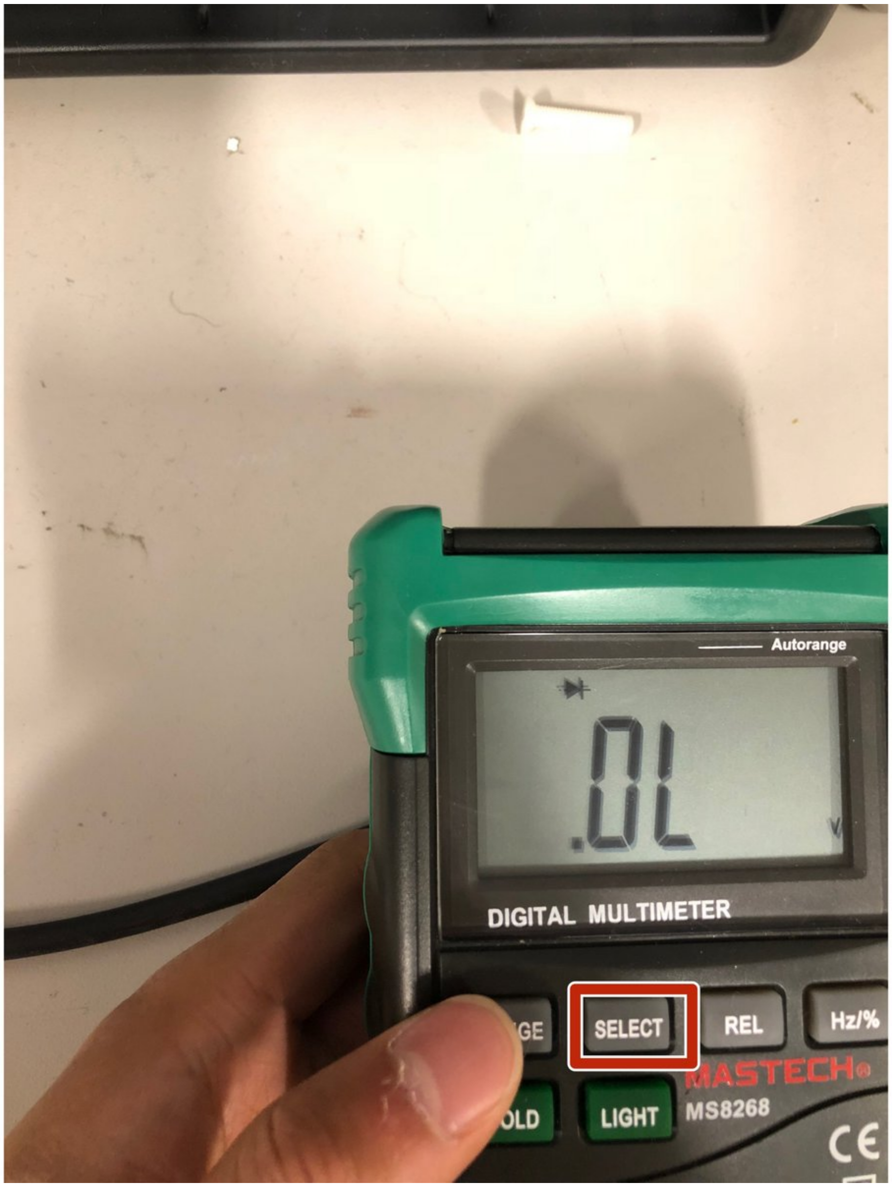
brandeismakerlab

How to Control a Vibrator

In this guide, I will teach you how to control a vibrator and make it vibrate at different strengths and turn it on an off.

Written By: Jonathan Lam



This document was generated on 2021-12-26 08:50:09 AM (MST).

PARTS:

- vibrator (1)
- Arduino UNO (1)
- transistor (1)
- Breadboard (1)
- USB 2.0 A-Male to B-Male Cable (1)
- jumper wire (5)

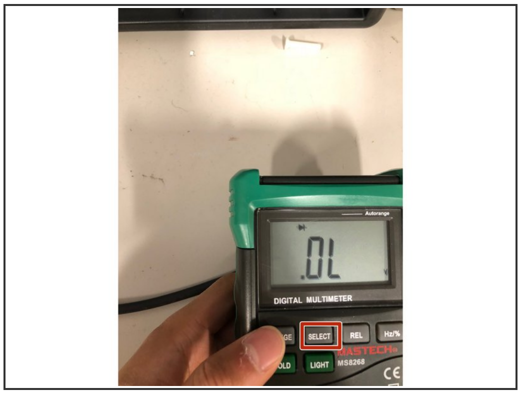
male to male

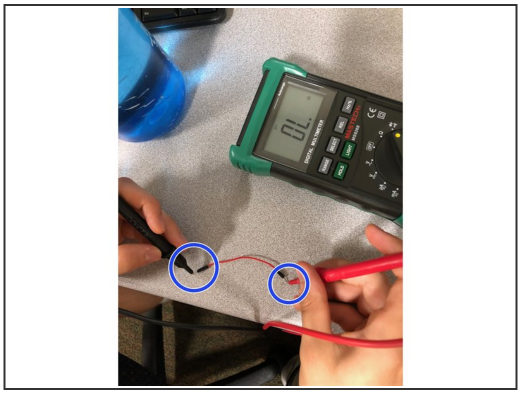
- multimeter (1)
- spools of solid core wire (2)

Spools of differing colors

- heat gun (1)
- Pliers (1)
- Heat Shrink Casing (1)

Step 1 — Preparing the Wires



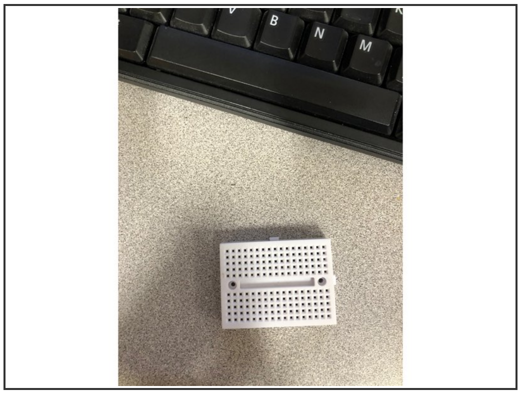


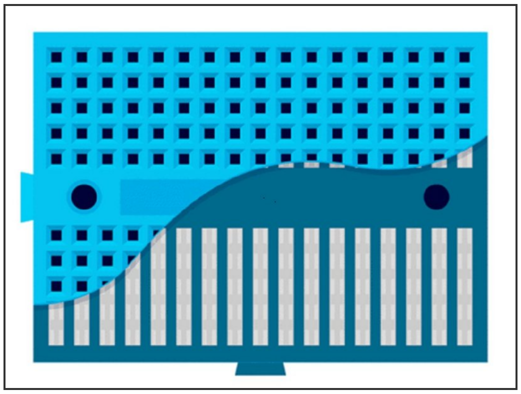
- Before starting assembly, make sure the jumper wires you plan on using operate properly.
- Put the multimeter into the continuity mode, which looks like the picture. Also, you need to click the select button once after turning the knob to that setting. You will be able to tell when the solid arrow with a line changes to an open circle and pulses.
- Then, press one of the prongs onto each side of the wire. If it beeps, that means a signal can travel through and the wire works properly. If no sound is made, get another wire. Keep doing this step until all five jumper wires work properly.

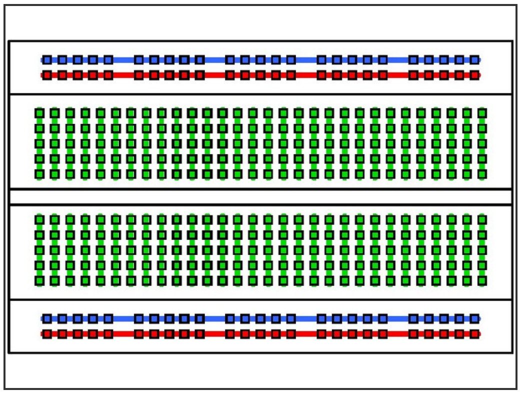
Step 2 — Preparing the Vibrator

- The first thing that needs to be addressed is the vibrator. The wires it comes with aren't compatible with any breadboard and will be useless unless we change something. Leads of solid wire need to be attached to the vibrators leads.
- The addition of these leads is done by soldering solid core wire to the wires coming off of the vibrator. The amount of solid core wire is based on what your needs are, because the length doesn't affect how the vibrator will operate.
- After that, heat shrink a case around the two recently soldered wires to protect them. This can be achieved by placing a casing over the wires, and applying heat from a heat gun until the casing shrinks around the wiring.
- This change will make the vibrator easier to connect with the breadboard. The red lead on the vibrator is the positive side where power should flow into, and the black/blue wire is the negative end, where power should flow back to the Ground.

Step 3 — Understanding a breadboard

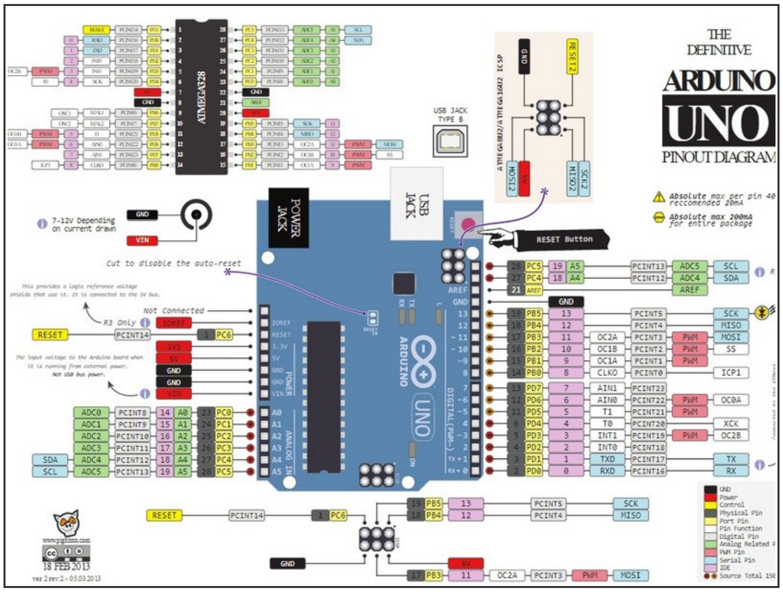


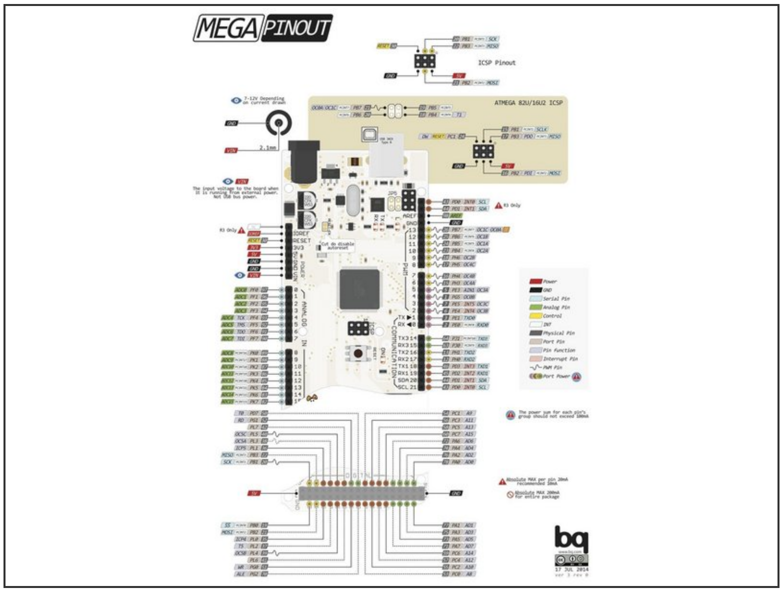




- Breadboards are arranged like in the picture. Note the connections like demonstrated in the picture. They are organized in columns when it is set up long ways horizontally, and don't cross over the gap in between.
- The larger looking breadboard is an actual breadboard that comes with a power line for positive and negative leads. The difference between a mini breadboard and a normal one is the number of columns and the power line.

Step 4 — Understanding the Arduino

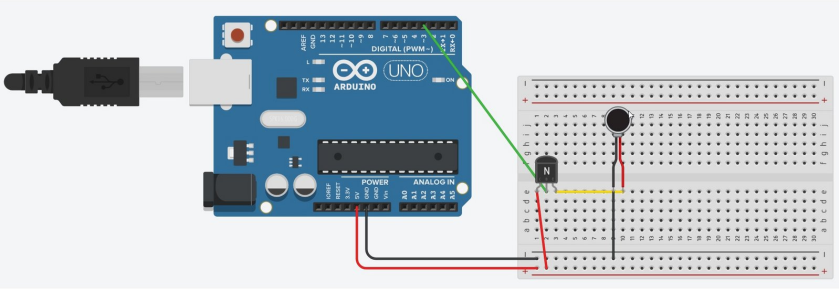

THE DEFINITIVE
ARDUINO
UNO
PINOUT DIAGRAM
ATMEGA328
USB JACK TYPE B
POWER JACK
RESET Button
7-12V Depending on current drawn
Cut to disable the auto-reset
Absolute max per pin 40mA reccomended 20mA
Absolute max 200mA for entire package
18 FEB 2013



- Take the Arduino and plug in a male to male jumper wire in a PWM supported digital pin. A PWM pin means that the Arduino can output information beside s a 1 or 0, or a yes/ no. This means that there can be a type of granularity with the outputted information.
- For an UNO, it is pins : 3,5,6,9-11
- For a Mega 2560, it is pins : 2-13

This document was generated on 2021-12-26 08:50:09 AM (MST).

Step 5 — Assembly Pt 1

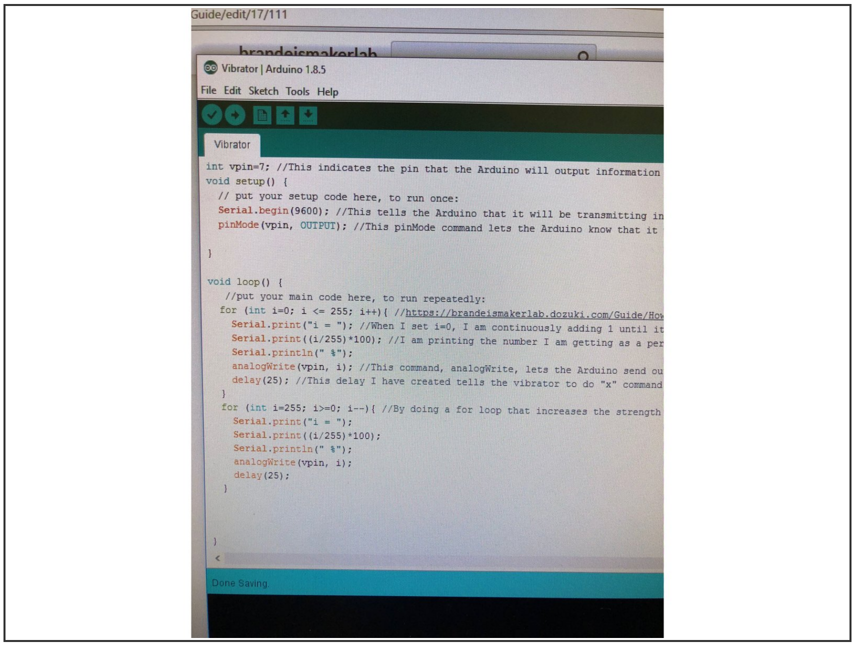


- Take a male to male jumper(j.) wire and attach it to a digital pin with PWM capabilities. Put the other end of that j. wire to column in the breadboard. In this column of the breadboard, the transistor's base pin will go here too.
- Secondly, take another male to male j. wire and plug it into the 5V power pin on the Arduino. This j. wire will lead to a separate column from the digital pin and transistor's base, attaching to the collector of the transistor.

Step 6 — Assembly Pt 2

- Attach the vibrator's positive end, the red wire from the vibrator, to the emitter pin of the transistor. This is achieved by putting both wires into the same column.
- Attach the black/blue wiring of the vibrator to another column, and have that go to the ground of the arduino via a j. wire.

Step 7 — Coding the Vibrator



```
int vpin=7; //This indicates the pin that the Arduino will output information
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600); //This tells the Arduino that it will be transmitting in
  pinMode(vpin, OUTPUT); //This pinMode command lets the Arduino know that it

}

void loop() {
   //put your main code here, to run repeatedly:
  for (int i=0; i <= 255; i++){ //https://brandeismakerlab.dozuki.com/Guide/How
    Serial.print("i = "); //When I set i=0, I am continuously adding 1 until it
    Serial.print((i/255)*100); //I am printing the number I am getting as a per
    Serial.println(" %");
    analogWrite(vpin, i); //This command, analogWrite, lets the Arduino send ou
    delay(25); //This delay I have created tells the vibrator to do "x" command
  }
  for (int i=255; i>=0; i--){ //By doing a for loop that increases the strength
    Serial.print("i = ");
    Serial.print((i/255)*100);
    Serial.println(" %");
    analogWrite(vpin, i);
    delay(25);
  }


}
```

- https://github.com/AidenKunkler-Peck/Vib...
- The link above will lead you to the sample code created to demonstrate how to operate a vibrator.
- The code is commented to make it easier to understand what each command's purpose is. Open the code and modify it to learn how to make your own code and have it do something else like turn off and on in certain intervals.

This document was generated on 2021-12-26 08:50:09 AM (MST).