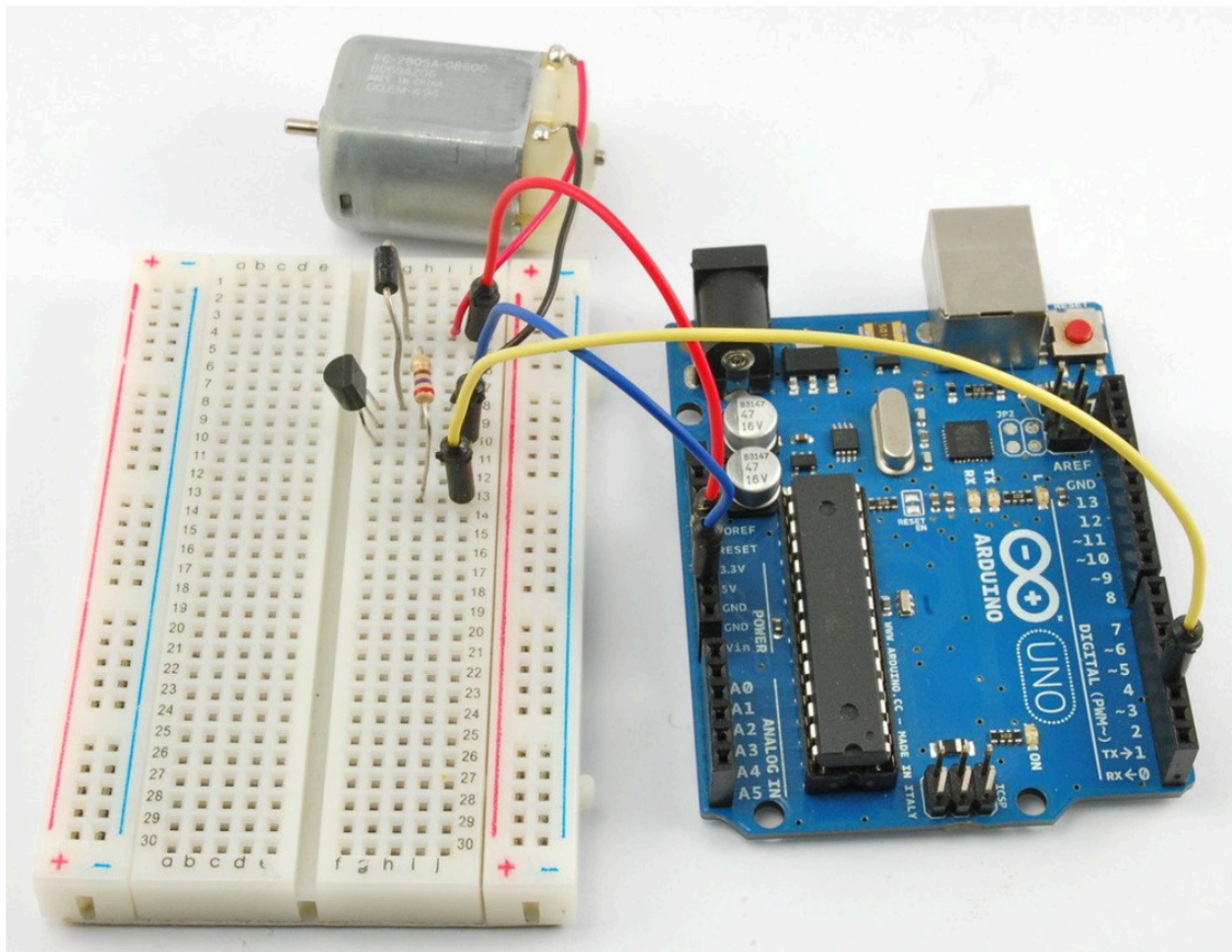


DC Motors- AKA SuperMOTORS Bro!!

Overview

In this lesson, you will learn how to control a small DC motor using an Arduino and a transistor.



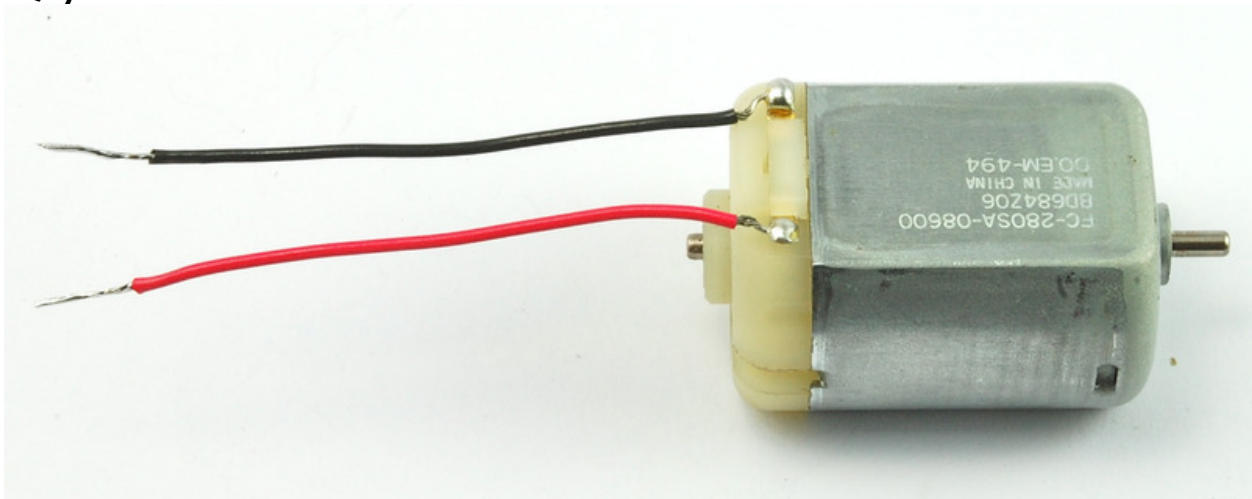
You will use an Arduino analog output (PWM) to control the speed of the motor by sending a number between 0 and 255 from the Serial Monitor.

Parts

To build the project described in this lesson, you will need the following parts.

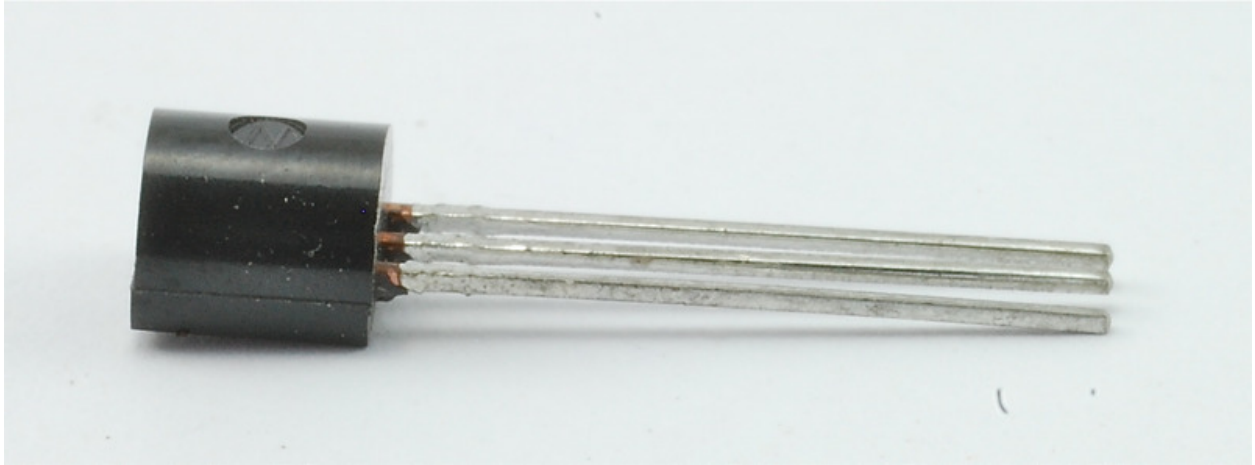
Part

Qty



Small 6V DC Motor

1



PN2222 Transistor

1



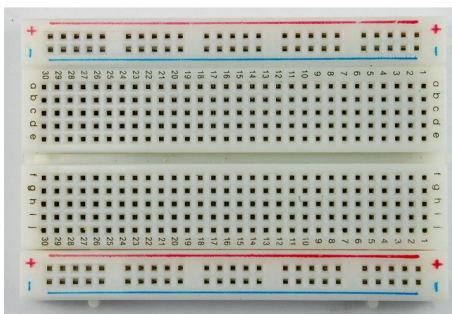
1N4001 diode

1



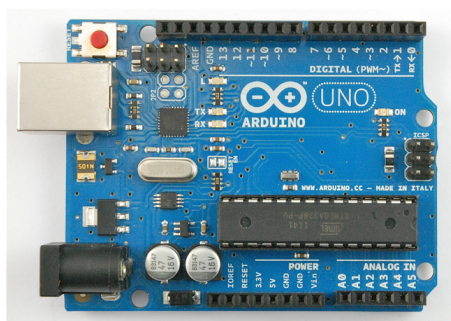
270 Ω Resistor (red, purple, brown stripes)

1



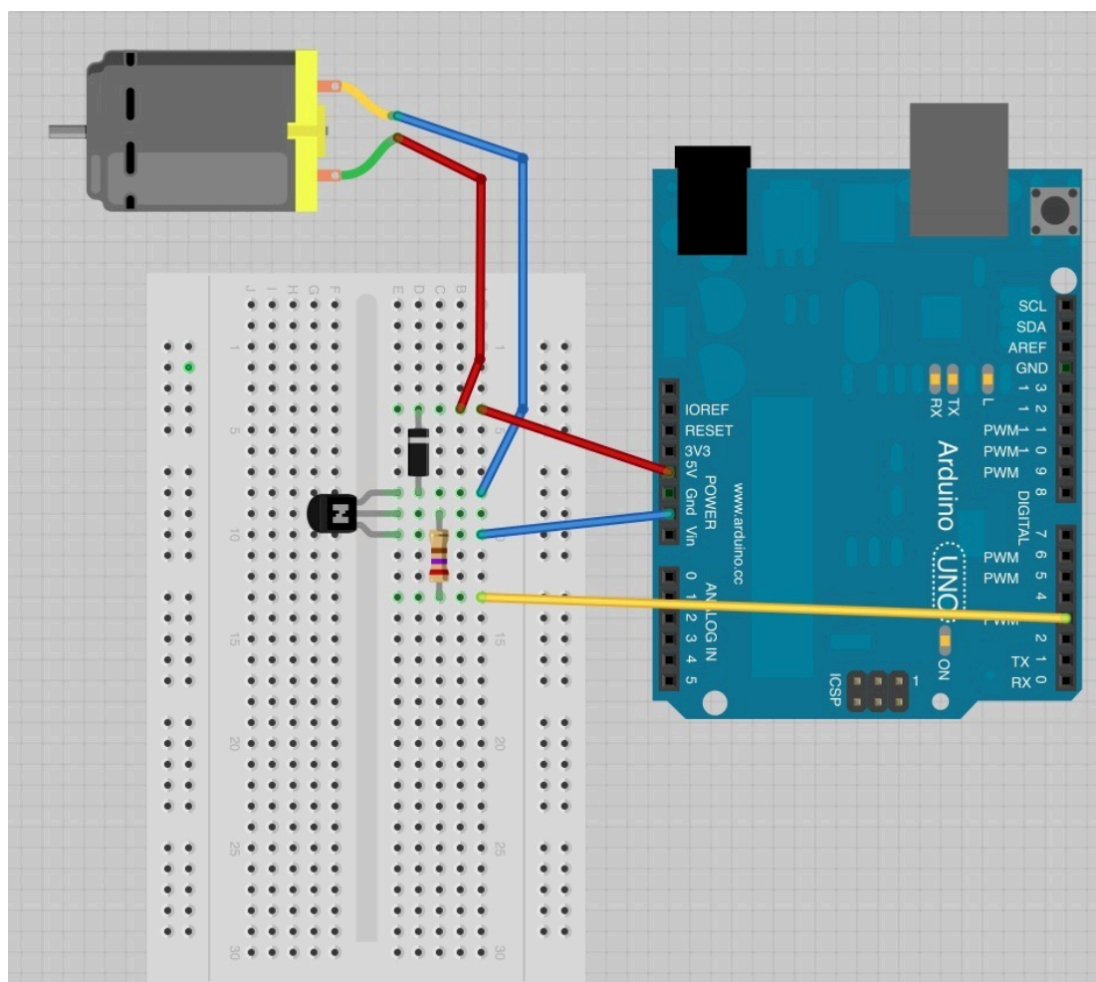
Half-size Breadboard

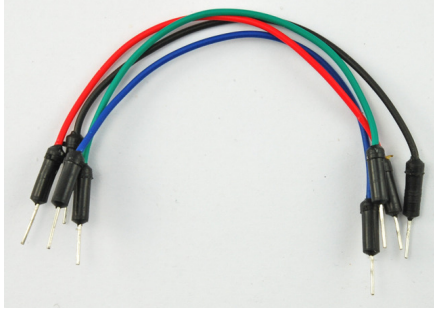
1



Arduino Uno R3

1





Jumper wire pack

1

Breadboard Layout

When you put together the breadboard, there are two things to look out for.

Firstly, make sure that the transistor is the right way around. The flat side of the transistor should be on the right-hand side of the breadboard.

Secondly the striped end of the diode should be towards the +5V power line – see the image above!

Arduino Code

Load up the following sketch onto your Arduino.

```
1.
2.
3.
4. int motorPin = 3;
5.
6. void setup()
7. {
8.   pinMode(motorPin, OUTPUT);
9.   Serial.begin(9600);
10.  while (! Serial);
11.  Serial.println("Speed 0 to 255");
12.}
13.
14.
15. void loop()
16. {
```

```
17.  if (Serial.available())
18.  {
19.      int speed = Serial.parseInt();
20.      if (speed >= 0 && speed <= 255)
21.      {
22.          analogWrite(motorPin, speed);
23.      }
24.  }
25. }
```

The transistor acts like a switch, controlling the power to the motor, Arduino pin 3 is used to turn the transistor on and off and is given the name 'motorPin' in the sketch.

When the sketch starts, it prompts you, to remind you that to control the speed of the motor you need to enter a value between 0 and 255 in the Serial Monitor.

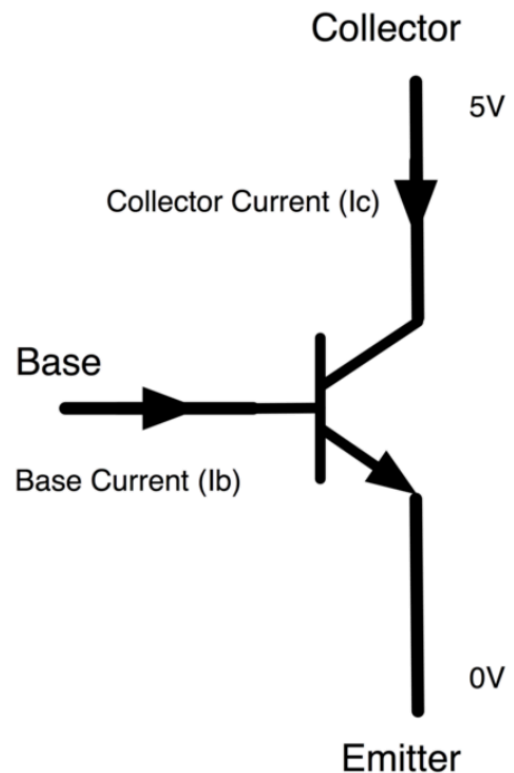
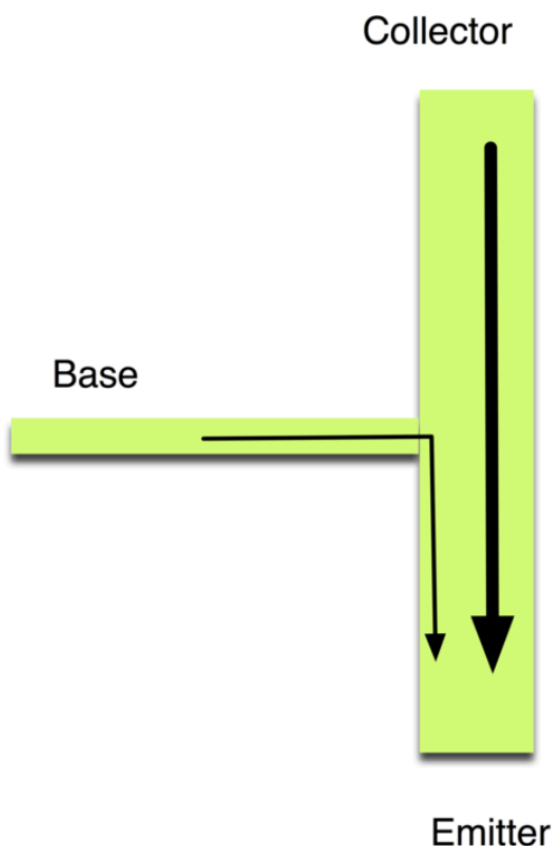
In the 'loop' function, the command 'Serial.parseInt' is used to read the number entered as text in the Serial Monitor and convert it into an 'int'.

You could type any number here, so the 'if' statement on the next line only does an analog write with this number if the number is between 0 and 255.

Transistors

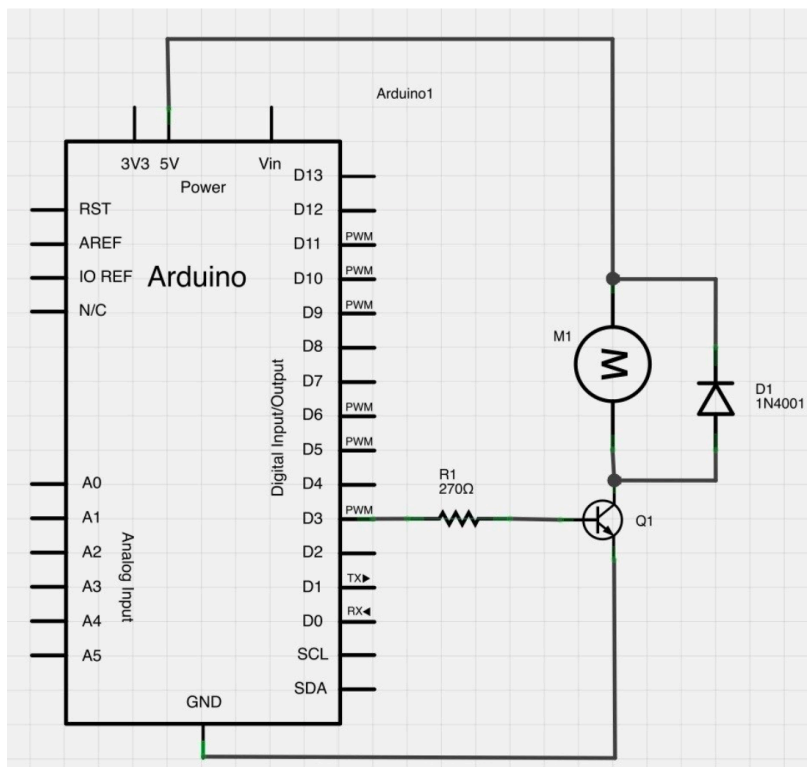
The small DC motor, is likely to use more power than an Arduino digital output can handle directly. If we tried to connect the motor straight to an Arduino pin, there is a good chance that it could damage the Arduino.

A small transistor like the PN2222 can be used as a switch that uses just a little current from the Arduino digital output to control the much bigger current of the motor.



The transistor has three leads. Most of the electricity flows from the Collector to the Emitter, but this will only happen if a small amount is flowing into the Base connection. This small current is supplied by the Arduino digital output.

The diagram below is called a schematic diagram. Like a breadboard layout, it is a way of showing how the parts of an electronic project are connected together.



The pin D3 of the Arduino is connected to the resistor. Just like when using an LED, this limits the current flowing into the transistor through the base.

There is a diode connected across the connections of the motor. Diodes only allow electricity to flow in one direction (the direction of their arrow).

When you turn the power off to a motor, you get a negative spike of voltage, that can damage your Arduino or the transistor. The diode protects against this, by shorting out any such reverse current from the motor.

The motor can be connected either way around.