



DC Motor and Encoder for Position and Speed Control



by tzeqianghun

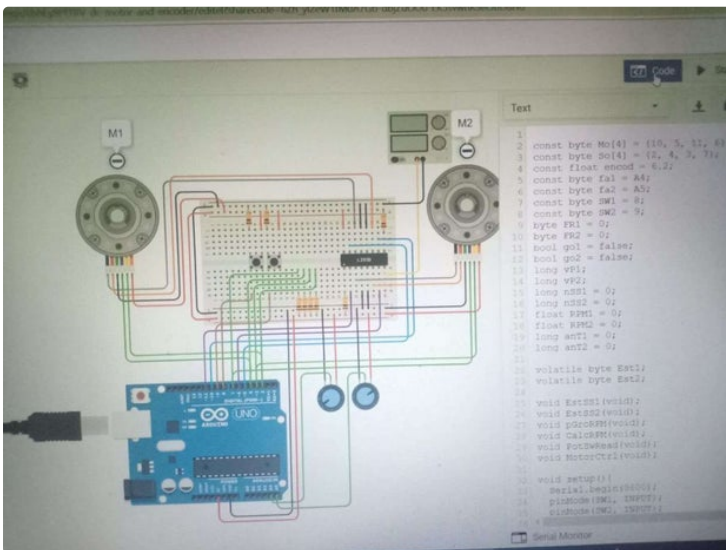
Introduction

We are a group of UQD10801(Robocon I) student from Universiti Tun Hussein Onn Malaysia(UTHM).We have 9 group in this course.My group is group 2.Our group's activity is DC motor and encoder for position and speed control.Our group's objective is control the DC motor rotate with the speed we needed.

Description

Driving electromotors needs a high current. In addition, spinning direction and speed are two important parameters to be controlled. These

requirements can be handled by using a microcontroller (or a development board like Arduino). But there is a problem; Microcontrollers cannot provide enough current to run the motor and if you connect the motor to the microcontroller directly, you may damage the microcontroller.For example, Arduino UNO pins are limited to 40mA of current which is far less than the 100-200mA current necessary to control a small hobby motor. To solve this, we should use a motor driver. Motor drivers can be connected to the microcontroller to receive commands and run the motor with a high current.

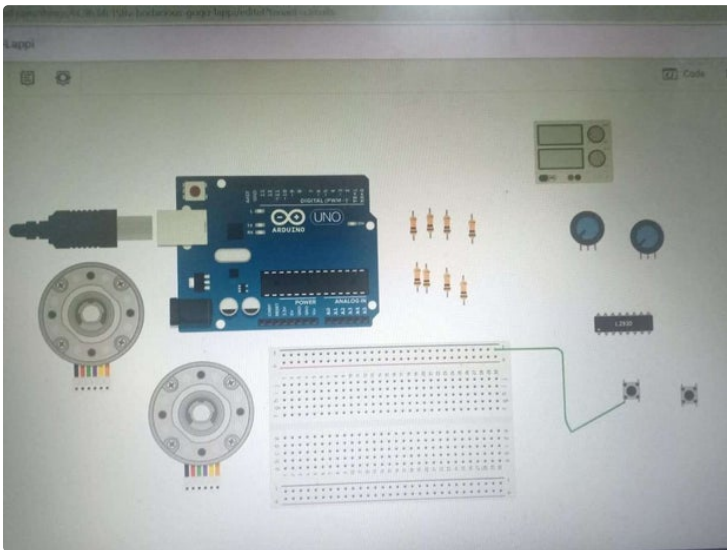


Step 1: Material Preparation

Required Material

For do this activity,we need to prepare:

- Arduino UNO R3
- 2 Potentiometer with 10kOhm
- 2 DC motor with encoder
- Power supply with 12V and 5A
- H-bridge motor driver
- 2 push button
- 8 resistor with 10kOhm
- Jumper wires
- Breadvroad small



Step 2: Pin Connection

1.For the left side motor connect to Arduino UNO 3:

-Channel A to pin 2

-Channel B to pin 4

2.For the right motor connect to Arduino UNO 3:

-Channel A to pin 3

5.For push button 1 connect to Arduino UNO 3:

-Terminal 1a to pin 8

6.For push button 2 connect to Arduino UNO 3:

-Terminal 1a to pin 9

-Channel B to pin 7

3.For the potentiometer 1 connect to Arduino UNO 3:

-Wiper to A4 analog

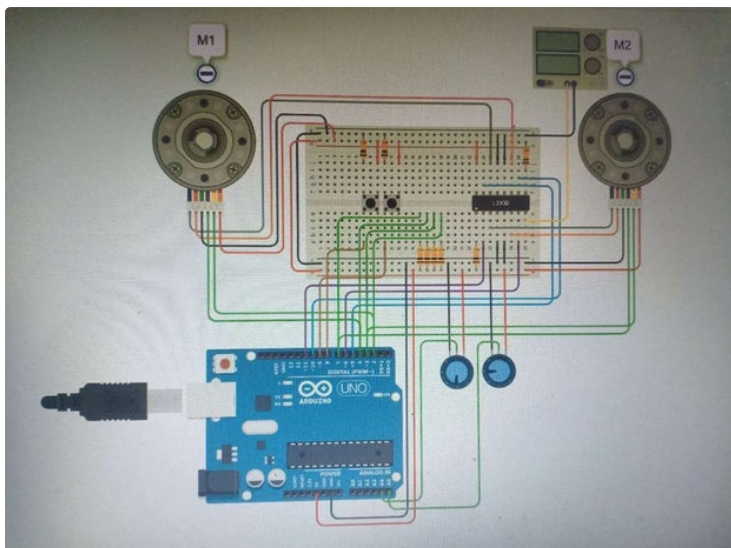
4.For potentiometer 2 connect to Arduino UNO 3:

-Wiper to A5 analog

7.For H-Bridge Motor Drive connect to Arduino UNO 3:

-Input 1 to pin 11


-Input 2 to pin 6



Step 3: Coding

You can download the coding for test the DC motor that can rotate.This coding can help you to make the DC motor rotate and work.You must download this coding to your PC for the next step.

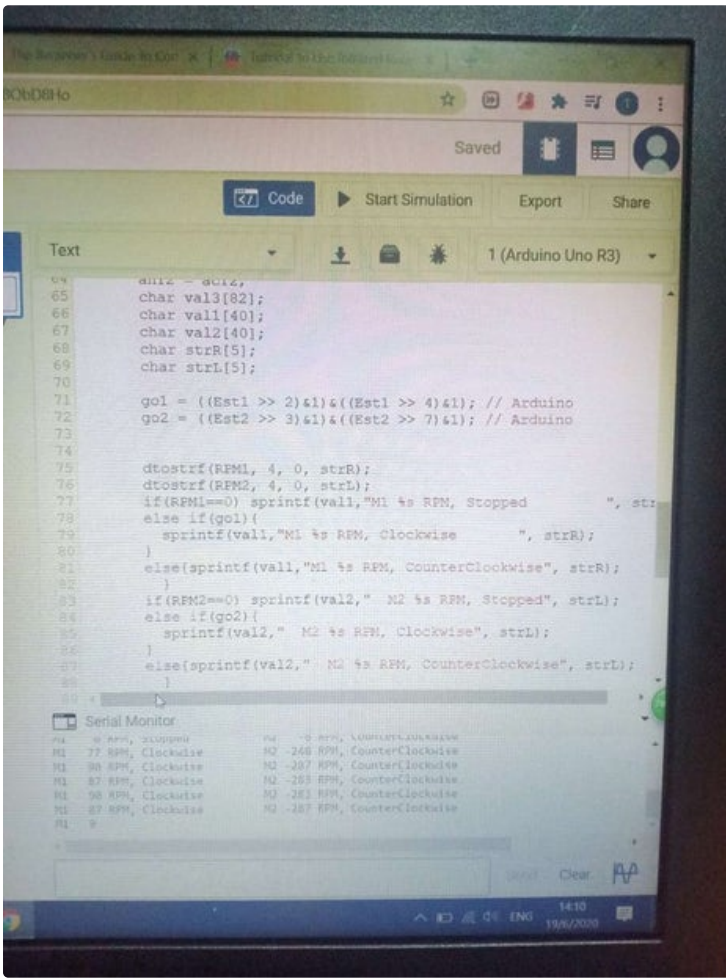
```
1
2 const byte Mo[4] = {10, 5, 11, 6};
3 const byte So[4] = {2, 4, 3, 7};
4 const float encod = 6.2;
5 const byte fal = A4;
6 const byte fa2 = A5;
7 const byte SW1 = 8;
8 const byte SW2 = 9;
9 byte FR1 = 0;
10 byte FR2 = 0;
11 bool go1 = false;
12 bool go2 = false;
13 long vP1;
14 long vP2;
15 long nSS1 = 0;
16 long nSS2 = 0;
17 float RPM1 = 0;
18 float RPM2 = 0;
19 long anT1 = 0;
20 long anT2 = 0;
21
22 volatile byte Est1;
23 volatile byte Est2;
24
25 void EstSS1(void);
26 void EstSS2(void);
27 void pGroRPM(void);
28 void CalcRPM(void);
29 void PotsWRead(void);
30 void MotorCtrl(void);
31
32 void setup() {
33
```

 <https://www.instructabl...> Download

Step 4: Testing the DC Motor

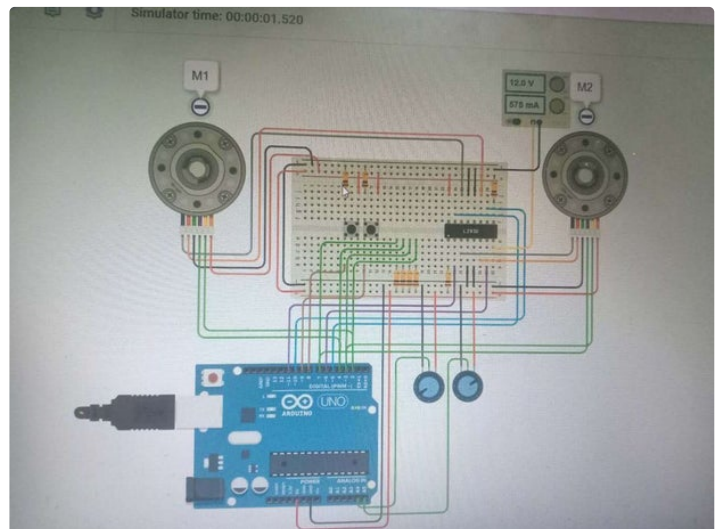
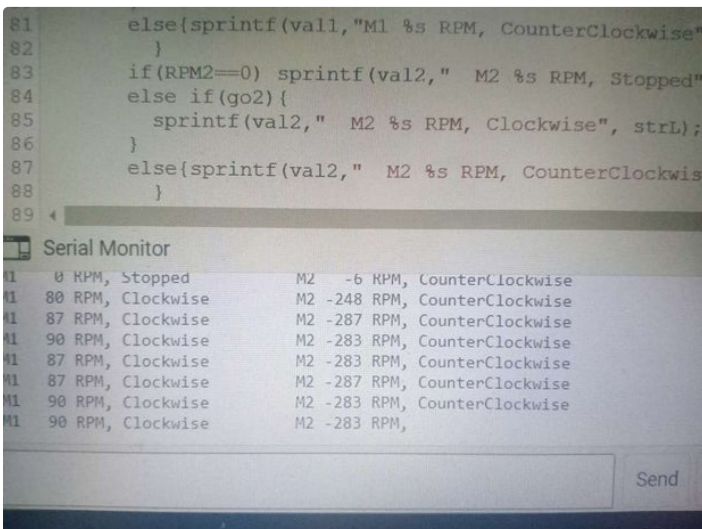
So, after you download the coding from the previous step, you must open it in your Arduino IDE that is already installed in your PC or use Tinkercad online. And then, upload this coding to your Arduino board via USB cable. If you use Tinkercad online, you just

upload this coding to the "Code" that is shown in the photo. After you upload the coding source, you can run the DC motor. If you use Tinkercad, you must press "Start Simulation" to start this system.



Step 5: Result

After we start the simulation, we can see both of the DC motor rotate but different direction. When we see the "Serial Monitor", M1's direction is clockwise and M2's direction is anti-clockwise.



Step 6: Video

<https://youtu.be/0SyDBgH04po>