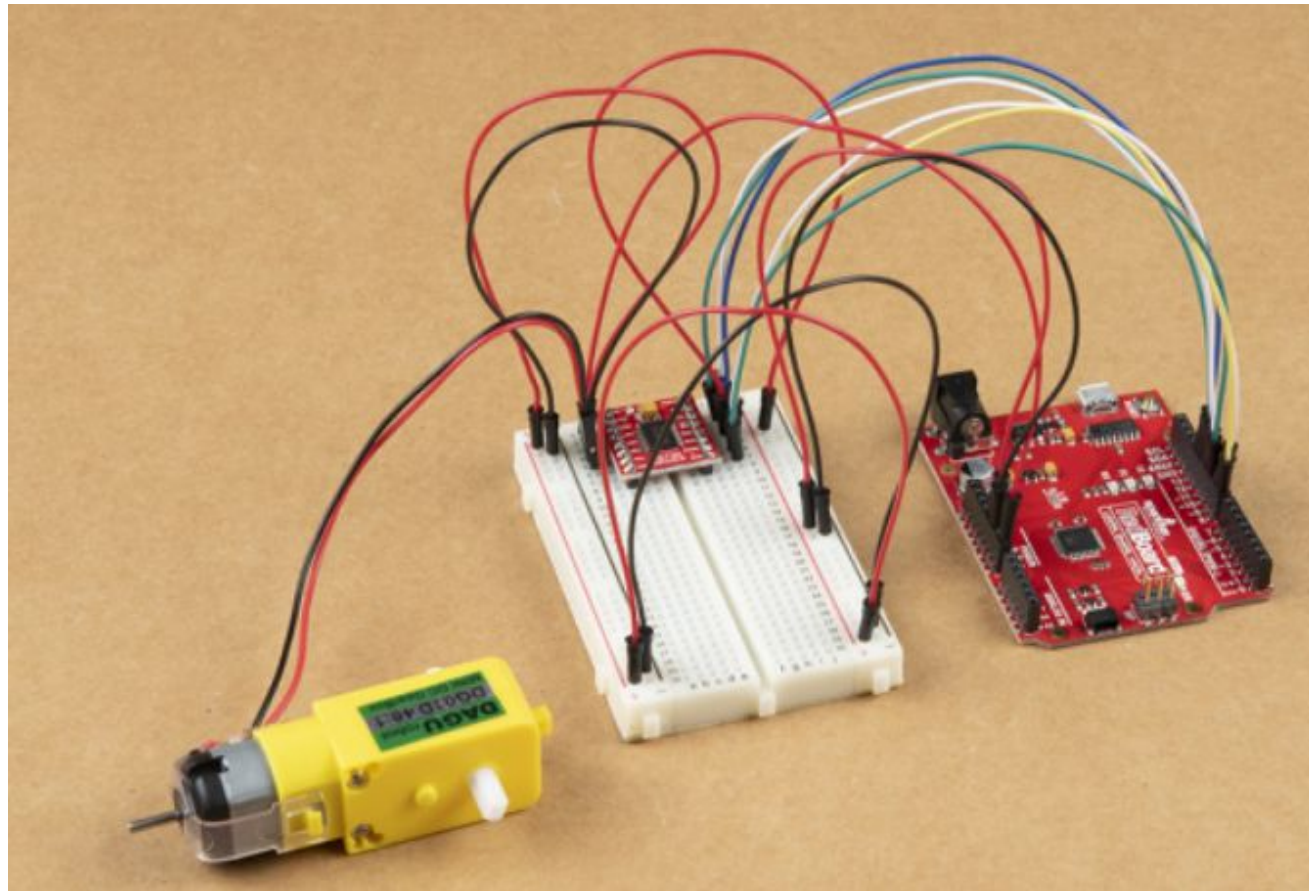
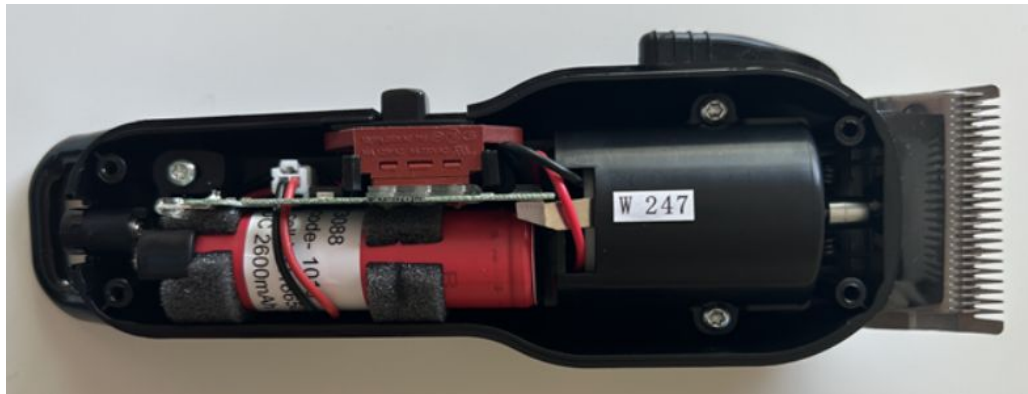


Assemble the Rotary Motor



Main Concept: Rotary Motors

- Rotary motors are used in many barbering clippers and trimmers. They are a great way to balance power with blade speed. They offer multiple speeds, making them highly adaptable for heavy-duty cutting.
- These motors are commonly found in the most powerful hair clippers available and can operate on either alternating current (AC) from home outlets or direct current (DC) from batteries.
- For this activity you are going to be building and programming a rotary motor to simulate one you would find in a clipper or trimmer.

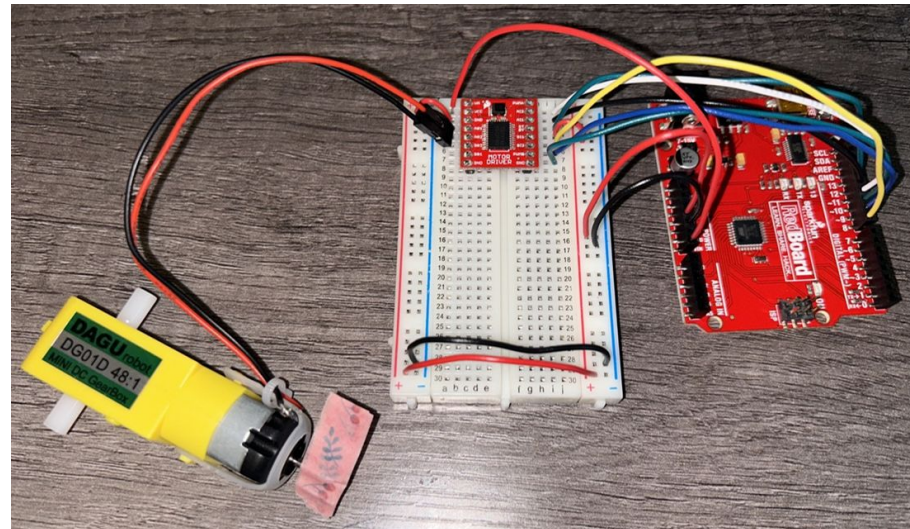


Your **goal** is to learn how to correctly connect each wire to the appropriate location to make the rotary motor spin.

It's also important to understand how the coding designed for its operation can be adjusted using specific instructions.

Today, we are going to connect a motor to some electronic components to make it spin. You will receive instructions on how to connect the motor by using the provided materials.

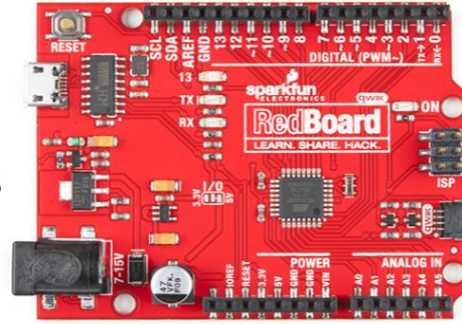
Then, you will have the opportunity to take on some challenges to play with the motor's speed.



Materials

1 SparkFun RedBoard -

An innovative Arduino-compatible development board, designed for easy programming and interfacing with various electronic devices.



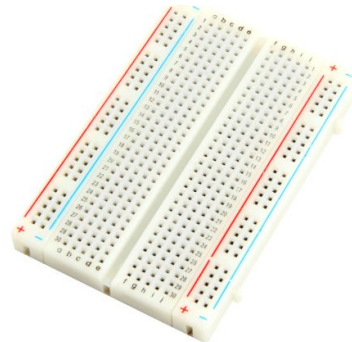
1 Reversible USB A to Reversible Micro-B Cable – 0.8 m

It connects the RedBoard to a computer.



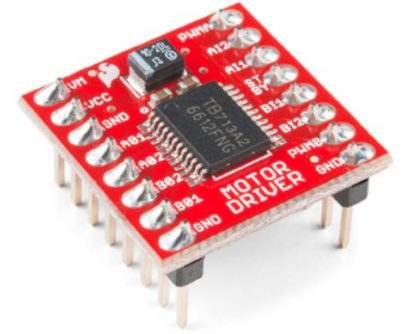
1 Breadboard -

A breadboard is a circuit building platform that allows you to connect multiple components without using a soldering iron.



1 SparkFun Motor Drive-

It takes commands for motors over three wires (two control direction, one controls speed), it uses these signals to control the current through two wires attached to your motor.



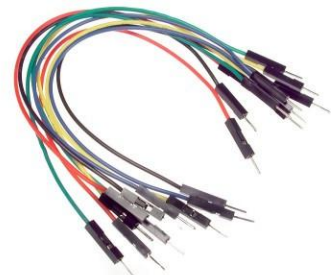
1 Hobby DC Gearmotor -

The motor consists of two parts: a small, fast-spinning DC motor and a plastic gearbox that slows down the motor's speed but makes stronger. Inside, there are coils of wire that create magnetic fields when electricity flows through them. When you provide power to these electromagnets, they turn the motor's drive shaft.



Jumper Wires -

It is used to create electrical connection.



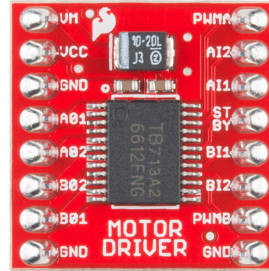
Switch -

It controls the open-ness or closed-ness of an electric circuit.



Place the motor driver and wires on the breadboard

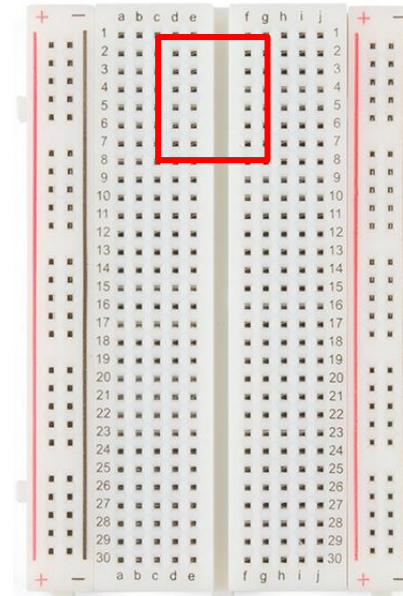
- Take your motor driver





and put it in your breadboard


- Put the left top leg of the motor driver into **C1**, and the right top leg of the motor driver into **G1**

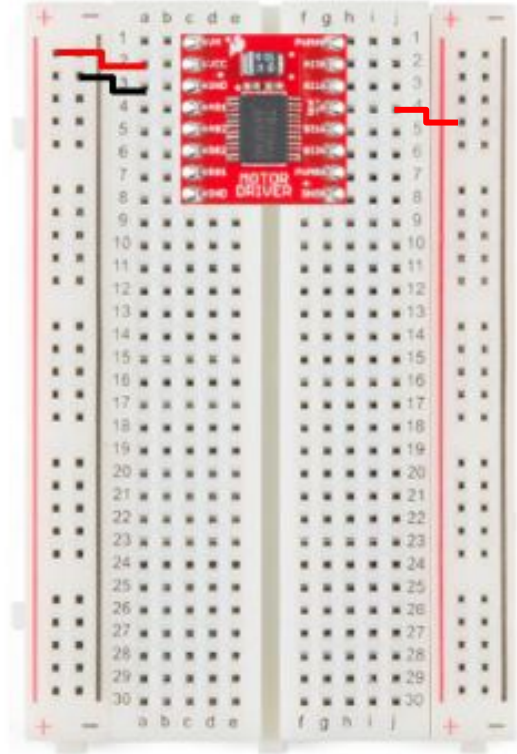
- Put the remaining legs in order
(**C1 ~ C8**, **G1 ~G8**)




- Take a jumper wire  and use it to ground your circuit
 - Put one end of the jumper wire into **A2** on your breadboard
 - Put the other end of the jumper wire into the **left side of row 1** in the **positive (+)** column of your breadboard


- Take a jumper wire  and use it to ground your circuit.
 - Put one end of the jumper wire into **A3** on your breadboard
 - Put the other end of the jumper wire into the **left side of row 2** in the **negative (-)** column of your breadboard

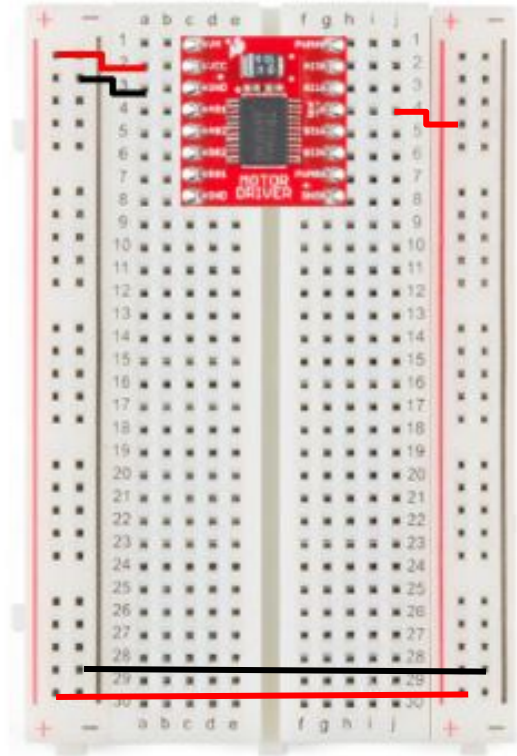
- Take a jumper wire  and use it to ground your circuit
 - Put one end of the jumper wire into **J4** on your breadboard
 - Put the other end of the jumper wire into the **right side of row 4** in the **positive (+)** column of your breadboard



* The color of a jumper wire doesn't matter, as long as both ends are inserted into the correct locations following the instructions


- Take a jumper wire  and use it to ground your circuit
 - Put one end of the jumper wire into the **left side of the last row** in the **positive (+)** column of your breadboard
 - Put the other end of the jumper wire into the **right side of the last row** in the **positive (+)** column of your breadboard

- Take a jumper wire  and use it to ground your circuit
 - Put one end of the jumper wire into the **left side of the 2nd last row** in the **negative (-)** column of your breadboard
 - Put the other end of the jumper wire into the **left side of the 2nd last row** in the **negative (-)** column of your breadboard



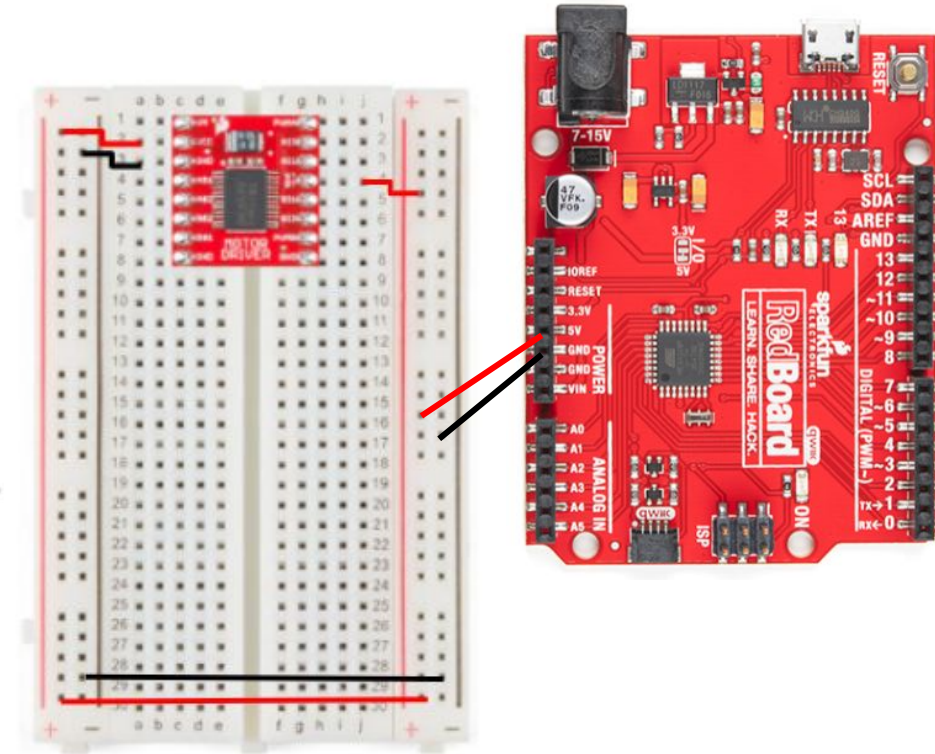
* The color of a jumper wire doesn't matter, as long as both ends are inserted into the correct locations following the instructions

Connect your Redboard to your breadboard

- Take a jumper wire  and use it to ground your circuit
 - Put one end of the jumper wire into **5V pin** on your RedBoard
 - Put the other end of the jumper wire into the **right side of row 16** in the **positive (+)** column of your breadboard



- Take another jumper wire and use it to ground your circuit
 - Put one end of the jumper wire into **GND pin** on your RedBoard
 - Put the other end of the jumper wire into the **right side of row 17** in the **negative (-)** column of your breadboard



* The color of a jumper wire doesn't matter, as long as both ends are inserted into the correct locations following the instructions

- Take a jumper wire  and use it to ground your circuit.

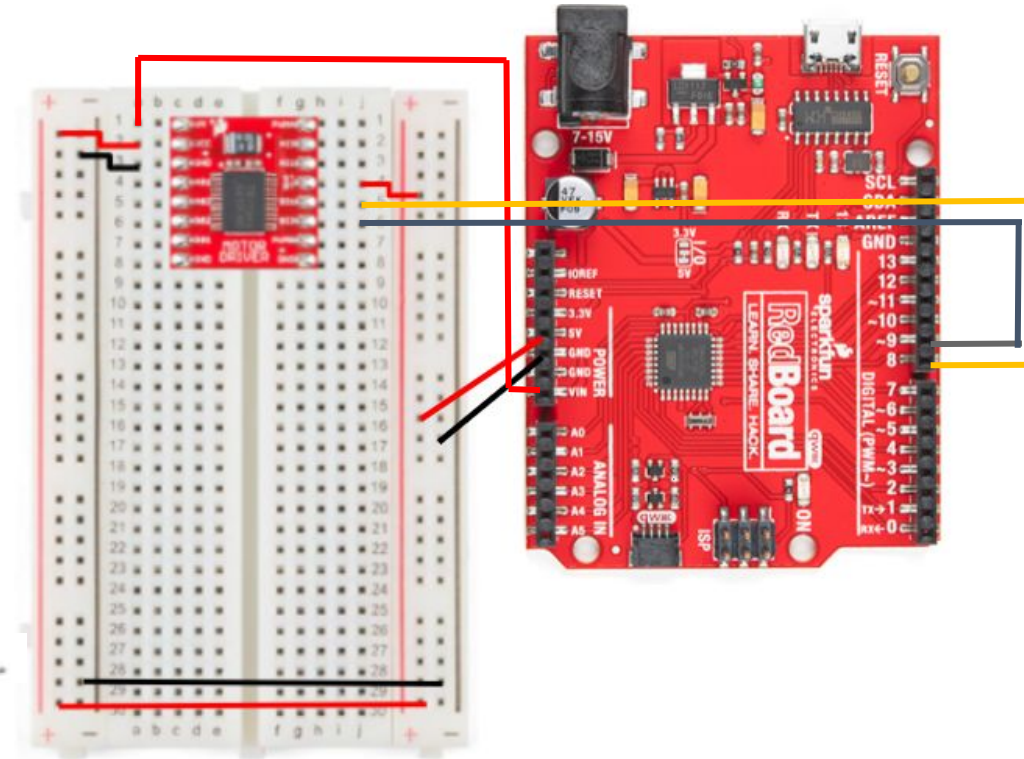
- Put one end of the jumper wire into **VIN** on your RedBoard
- Put the other end of the jumper wire into **A1** on your breadboard

- Take a jumper wire  and use it to ground your circuit

- Put one end of the jumper wire into **Digital 8** on your RedBoard
- Put the other end of the jumper wire into **J5** on your breadboard

- Take a jumper wire  and use it to ground your circuit

- Put one end of the jumper wire into **Digital 9** on your RedBoard
- Put the other end of the jumper wire into **J6** on your breadboard



* The color of a jumper wire doesn't matter, as long as both ends are inserted into the correct locations following the instructions

- Take a jumper wire  and use it to ground your circuit

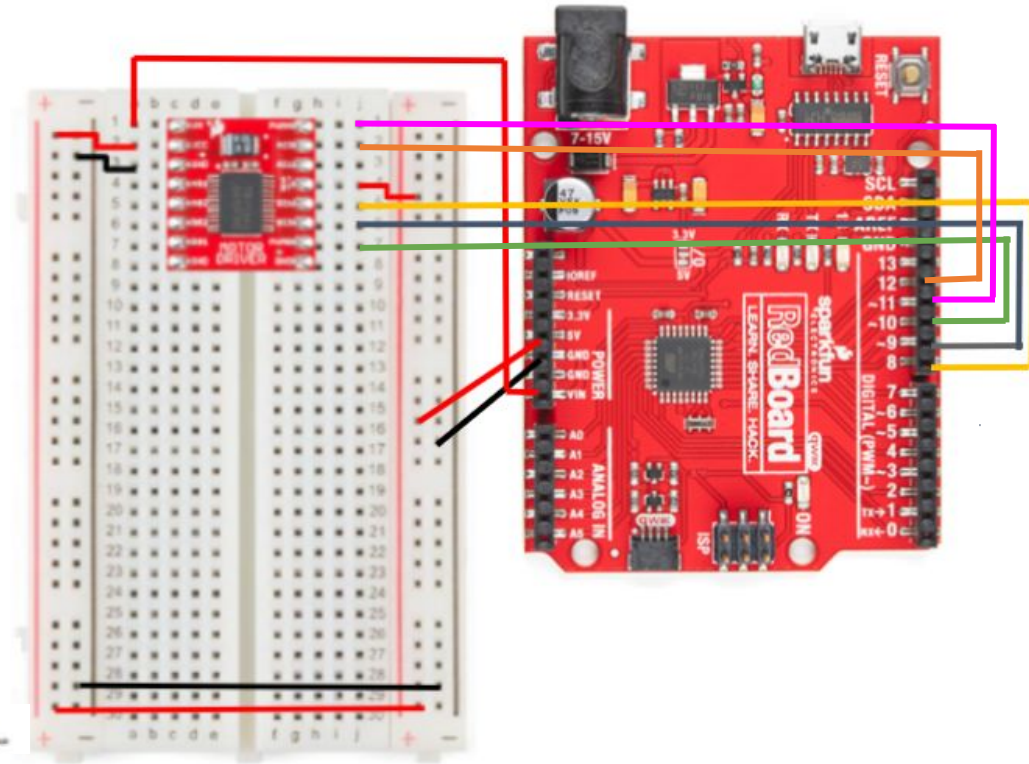
- Put one end of the jumper wire into **Digital 10** on your RedBoard
- Put the other end of the jumper wire into **J7** on your breadboard

- Take a jumper wire  and use it to ground your circuit

- Put one end of the jumper wire into **Digital 11** on your RedBoard
- Put the other end of the jumper wire into **J1** on your breadboard

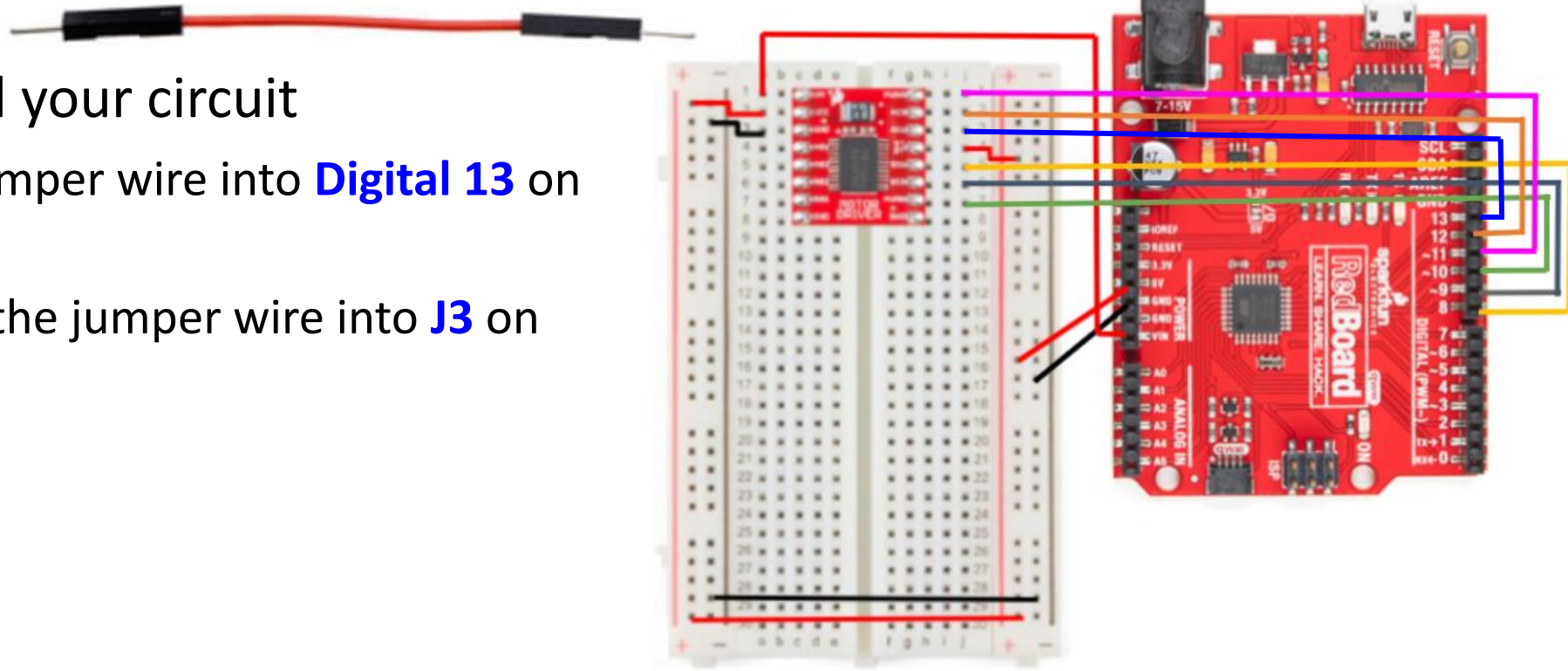
- Take a jumper wire  and use it to ground your circuit

- Put one end of the jumper wire into **Digital 12** on your RedBoard
- Put the other end of the jumper wire into **J2** on your breadboard



** The color of a jumper wire doesn't matter, as long as both ends are inserted into the correct locations following the instructions*

- Take a jumper wire and use it to ground your circuit
 - Put one end of the jumper wire into **Digital 13** on your RedBoard
 - Put the other end of the jumper wire into **J3** on your breadboard



* The color of a jumper wire doesn't matter, as long as both ends are inserted into the correct locations following the instructions

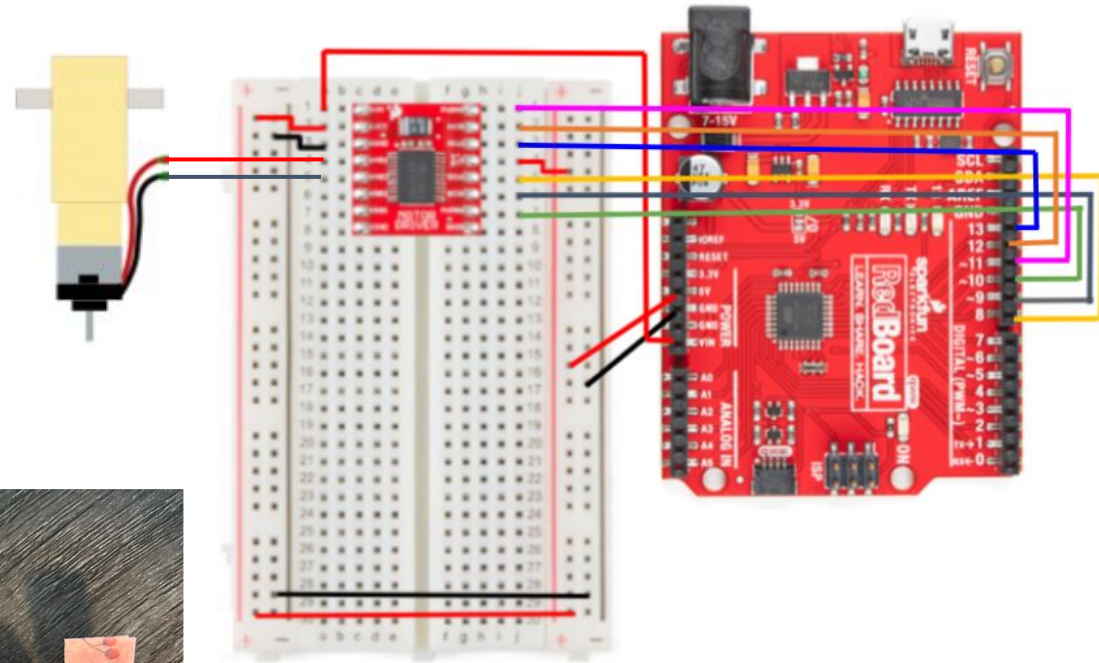
Connect your motor to your breadboard

- Take a Motor



and use it to connect to your circuit

- Put **Red +** wire of the motor into **A4** on your breadboard
- Put **Black -** wire of the motor into **A5** on your breadboard
- Attach a piece of tape to the motor shaft so that you can see it spinning



Software

This next section will describe how to setup the software for this project

- Take a USB cable



to connect your RedBoard to the computer

- Plug the Micro-B cable end of the USB cable



into the RedBoard

- Plug Reversible USB A end



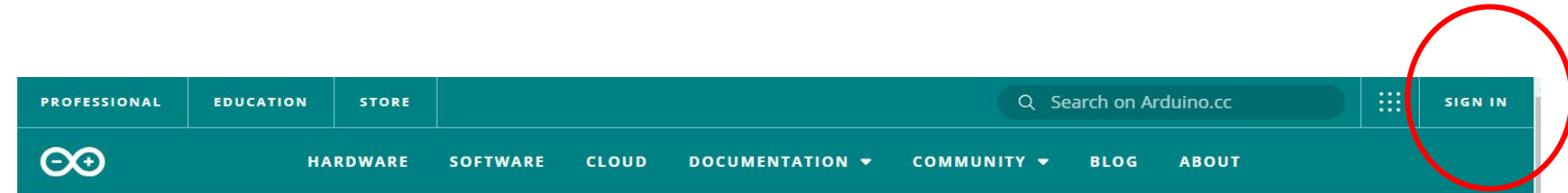
into the computer

PLUG YOUR MICROCONTROLLER INTO THE COMPUTER



Let's use Arduino Web Editor

➔ <https://www.arduino.cc/>



- Click "SIGN IN"



BUY AN ARDUINO



LEARN ARDUINO



DONATE



ARDUINO IN THE CLOUD



That's I(o)T!

Discover the easy way to build IoT projects with the power of the ESP32-S3.

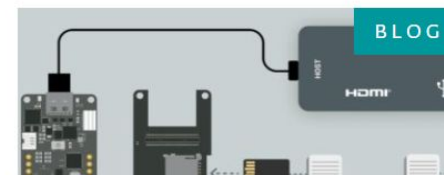
Buy now!



Arduino OPTA

Secure, easy-to-use micro PLC with Industrial IoT capabilities

Check it out now!



BLOG




BLOG


Help


- Sign up with your email address. Once you complete, you may receive the verification email

Welcome to Arduino
Learn and create, with the ever-growing platform

 **SIGN IN**


Don't have an account yet? [Create one.](#)

Hey Kids!
To sign in, click here if you are under 14 



Sign up to Arduino

Enter your email *

Choose a username * 

Choose a password *
Minimum 8 characters long

I have read the [Privacy Policy](#) and accept the [Terms of Service](#) *

I would like to receive the Arduino newsletter.

I agree to see personalised commercial offers from Arduino based on my browsing and purchasing behaviour.

I would like to get email updates about special deals and commercial offers from Arduino.

SIGN UP

- Install Arduino Create Plugin:

Arduino Create Plugin - Getting Started

(1)

WELCOME TO THE ARDUINO CREATE AGENT INSTALLATION!



You're about to begin the process of downloading and installing the Arduino Create Agent. The agent will provide you with several features:

- Recognize Arduino boards and other supported devices connected to your computer via USB;
- Upload sketches from your web browser to your boards via USB or through a network;
- Read data from serial monitor, as well as write to it.

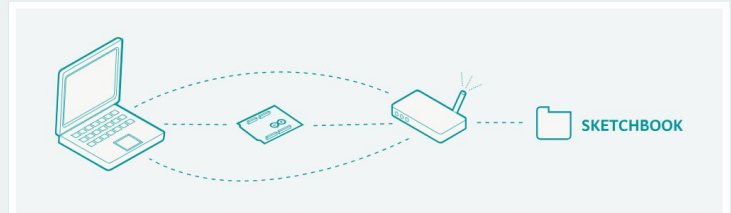
START

(2)

DOWNLOAD THE CREATE AGENT FOR WINDOWS

SETUP STEPS

1. DOWNLOAD AGENT
2. INSTALL AGENT
3. CONGRATULATIONS!



You need to download and install the Create Agent to be able to upload sketches from Arduino Cloud to your board. Please note that you have to be Administrator of your system to install the Agent. Administrative privileges aren't required for MacOS El Capitan or an earlier version.

Source code for the Create Agent is available on [GitHub](#).

DOWNLOAD FOR WIN32

DOWNLOAD FOR WIN64

(3)

AGENT CORRECTLY INSTALLED!

SETUP STEPS

- ✓ DOWNLOAD AGENT
2. INSTALL AGENT
3. CONGRATULATIONS!



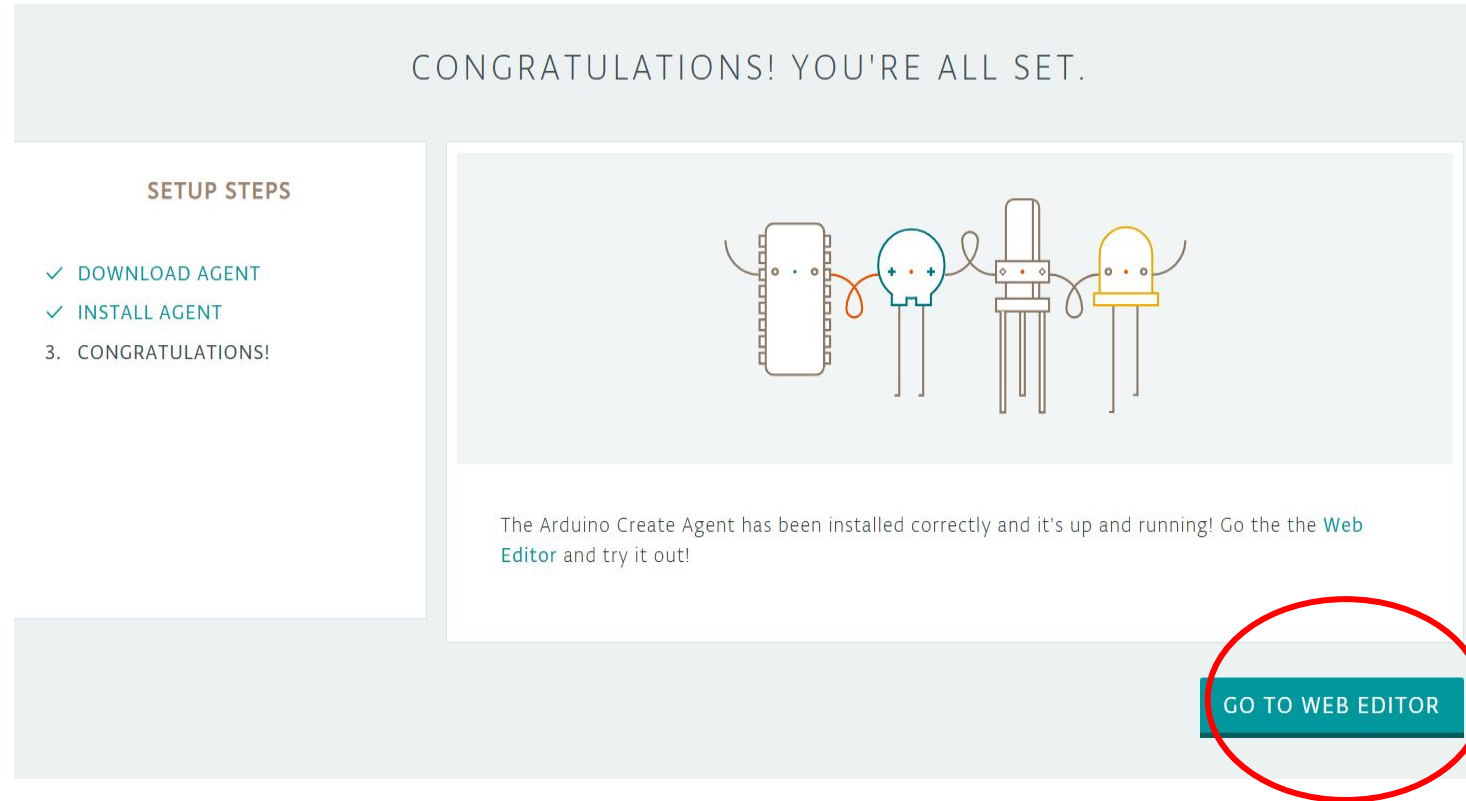
You should now see the Arduino icon on the bottom bar of your Desktop. Click on the tray icon to pause the agent or to visit the Arduino Cloud webpage.

If you happen to close the agent, you can relaunch the agent as you would any other application.

NEXT

- On the last page of installation, click “GO TO WEB EDITOR”

(4)

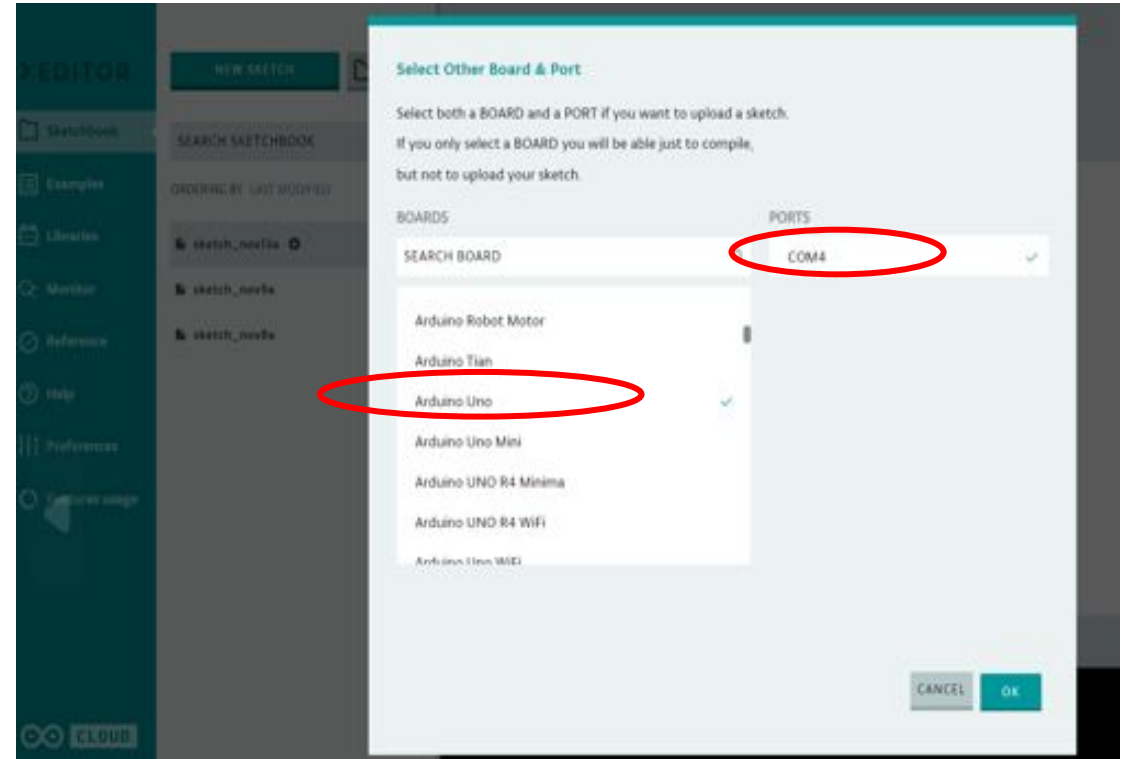
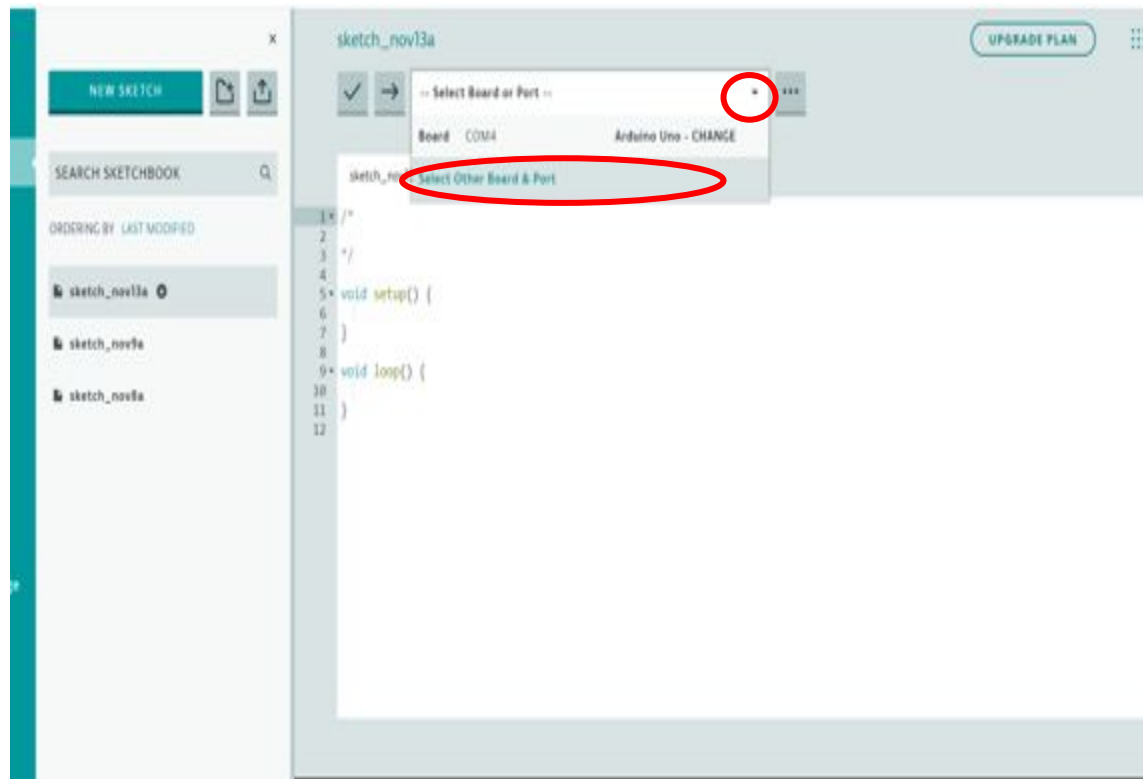


Or you can use this link: <https://create.arduino.cc/editor>

Once the webeditor is open, click on **“Select Board or Port”** on the computer that should be used to send data to the Arduino.

Choose **Arduino Uno** for Board and **then select** what is found in the port Port
-> This computer found the RedBoard attached to the COM4 communications port
(Yours may be a different number other than 4).

Click **“OK”**



Let's run some code.
Please copy this code

```
//PIN VARIABLES
//the motor will be controlled by the motor A pins on the motor driver
const int AIN1 = 13;           //control pin 1 on the motor driver for the right
motor
const int AIN2 = 12;           //control pin 2 on the motor driver for the right
motor
const int PWMA = 11;           //speed control pin on the motor driver for the right
motor

//VARIABLES
int motorSpeed = 0;           //starting speed for the motor

void setup() {
  //set the motor contro pins as outputs
  pinMode(AIN1, OUTPUT);
  pinMode(AIN2, OUTPUT);
  pinMode(PWMA, OUTPUT);
}

void loop() {
  //drive motor forward (positive speed)
  digitalWrite(AIN1, HIGH);           //set pin 1 to high
  digitalWrite(AIN2, LOW);           //set pin 2 to low
  analogWrite(PWMA, 255);           //now that the motor direction is set,
drive it at max speed
  delay(3000);

  //drive motor backward (negative speed)
  digitalWrite(AIN1, LOW);           //set pin 1 to low
  digitalWrite(AIN2, HIGH);           //set pin 2 to high
  analogWrite(PWMA, 255);           //now that the motor direction is set,
drive it at max speed
  delay(3000);

  //stop motor
  digitalWrite(AIN1, LOW);           //set pin 1 to low
  digitalWrite(AIN2, LOW);           //set pin 2 to low
  analogWrite(PWMA, 0);           //now that the motor direction is set, stop
motor
  delay(3000);
}
```

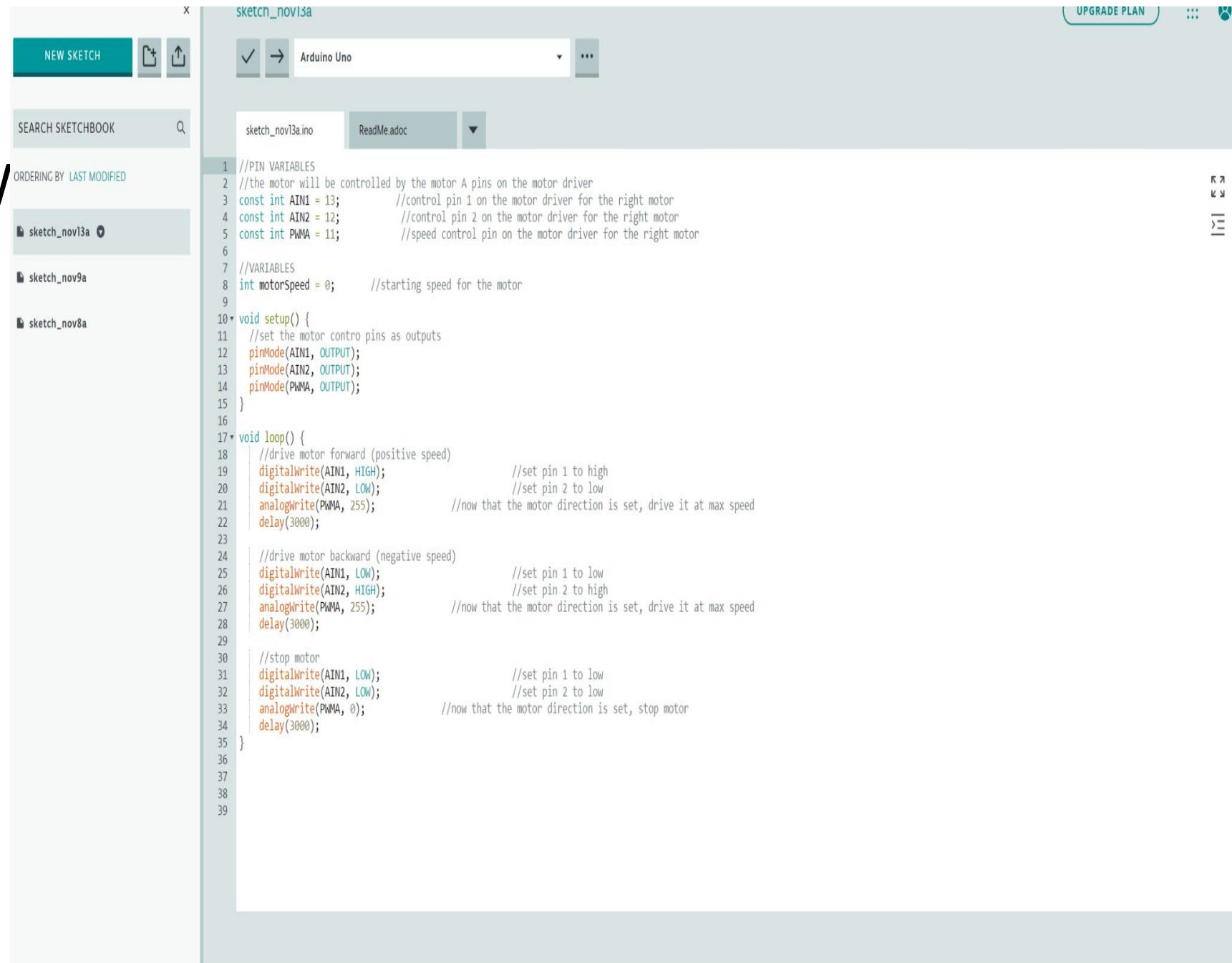
You can also find the code in this document:

https://docs.google.com/document/d/12PobvjYIU8oSyFdMxSXkQILk6IKA-tHi9E6_ID-yyc/edit

Paste the code into W

Click  (Verify)

Click  (Upload)



The screenshot shows the Arduino IDE interface. On the left, the 'Sketchbook' panel displays a list of sketches: 'sketch_nov13a', 'sketch_nov9a', and 'sketch_nov8a'. The main workspace shows the code for 'sketch_nov13a.ino'. The code is as follows:

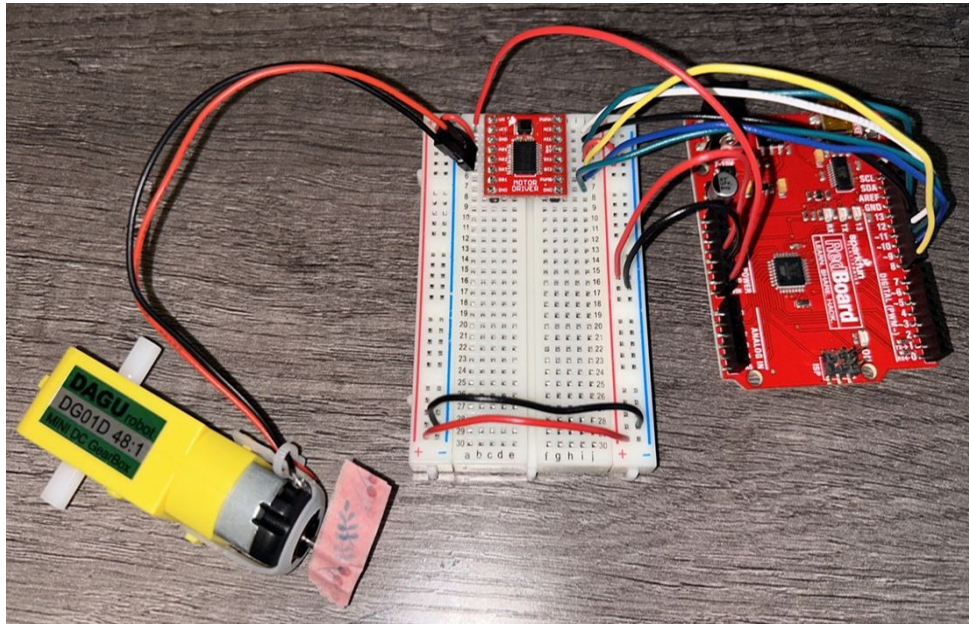
```
1 //PIN VARIABLES
2 //the motor will be controlled by the motor A pins on the motor driver
3 const int AIN1 = 13; //control pin 1 on the motor driver for the right motor
4 const int AIN2 = 12; //control pin 2 on the motor driver for the right motor
5 const int PWMA = 11; //speed control pin on the motor driver for the right motor
6
7 //VARIABLES
8 int motorSpeed = 0; //starting speed for the motor
9
10 void setup() {
11 //set the motor contro pins as outputs
12 pinMode(AIN1, OUTPUT);
13 pinMode(AIN2, OUTPUT);
14 pinMode(PWMA, OUTPUT);
15 }
16
17 void loop() {
18 //drive motor forward (positive speed)
19 digitalWrite(AIN1, HIGH); //set pin 1 to high
20 digitalWrite(AIN2, LOW); //set pin 2 to low
21 analogWrite(PWMA, 255); //now that the motor direction is set, drive it at max speed
22 delay(3000);
23
24 //drive motor backward (negative speed)
25 digitalWrite(AIN1, LOW); //set pin 1 to low
26 digitalWrite(AIN2, HIGH); //set pin 2 to high
27 analogWrite(PWMA, 255); //now that the motor direction is set, drive it at max speed
28 delay(3000);
29
30 //stop motor
31 digitalWrite(AIN1, LOW); //set pin 1 to low
32 digitalWrite(AIN2, LOW); //set pin 2 to low
33 analogWrite(PWMA, 0); //now that the motor direction is set, stop motor
34 delay(3000);
35 }
36
37
38
39
```


If you see "Success:~" at the bottom of your screen, then your program has successfully uploaded

```
Success: Saved on your online Sketchbook and done uploading sketch_nov13a
```

```
avidude: safemode: efuse reads as 0  
avidude: safemode: Fuses OK (E:00, H:00, L:00)  
avidude done. Thank you.
```

Your Motor will spin



What you should see

- The motor will spin in one direction at maximum speed (255) for 3 seconds
- Then the motor will spin the opposite direction at maximum speed (255) for 3 seconds
- The motor will stop for 3 seconds
- Repeat

Coding Challenges

1. Let's make the motor spin at a slower rate in one direction
2. Let's make the motor spin at a faster rate in one direction
3. Let's make the motor spin at a slower rate in one direction for 5 seconds, and then spin the motor in the opposite direction at a faster rate for 5 seconds

Coding Notes

digitalWrite() on pins AIN1 and AIN2 sets the direction for the motor to spin on motor connected to channel A. When one pin is **HIGH** and the other is **LOW**, the motor will spin in one direction. The motor will spin the other direction when the logic is reversed. Setting both pins to **LOW** will stop the motor.

```
void loop() {  
  //drive motor forward (positive speed)  
  digitalWrite(AIN1, HIGH); //set pin 1 to high  
  digitalWrite(AIN2, LOW); //set pin 2 to low  
  analogWrite(PWMA, ); //now that the motor direction is  
  set, drive it at max speed  
  delay(3000)  
  
  //drive motor backward (negative speed)  
  digitalWrite(AIN1, HIGH); //set pin 1 to high  
  digitalWrite(AIN2, LOW); //set pin 2 to low  
  analogWrite(PWMA, ); //now that the motor direction is  
  set, drive it at max speed  
  delay(3000)  
}
```


The number you input will determine the duration of the motor pause/stop. The example setting is in the picture is for **3** seconds

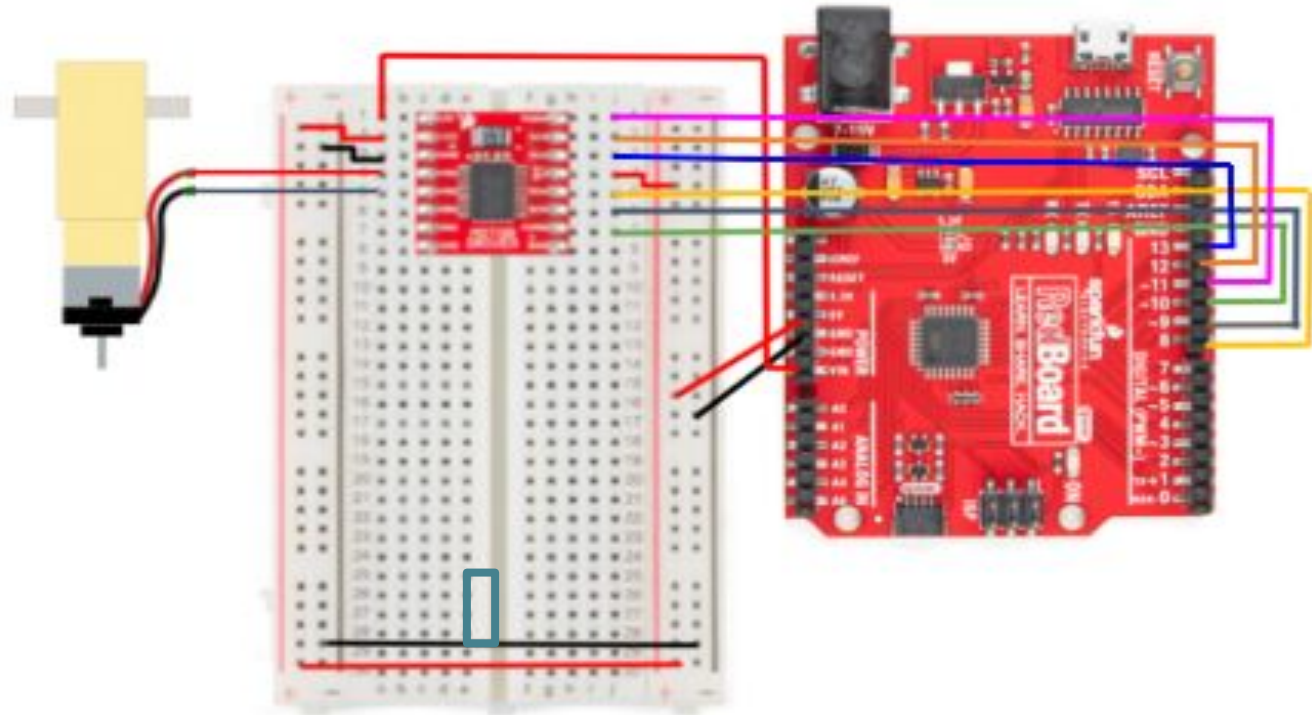
analogWrite() on the PWMA pin will tell the motor to move at a certain speed. The value must be a value between **0** and **255**.

Coding Challenges Solutions:

<https://docs.google.com/document/d/1RG8XurrUKqiSf7blmnWvOzL8pKCvBEQIkXkhBnPLiU0/edit?usp=sharing>

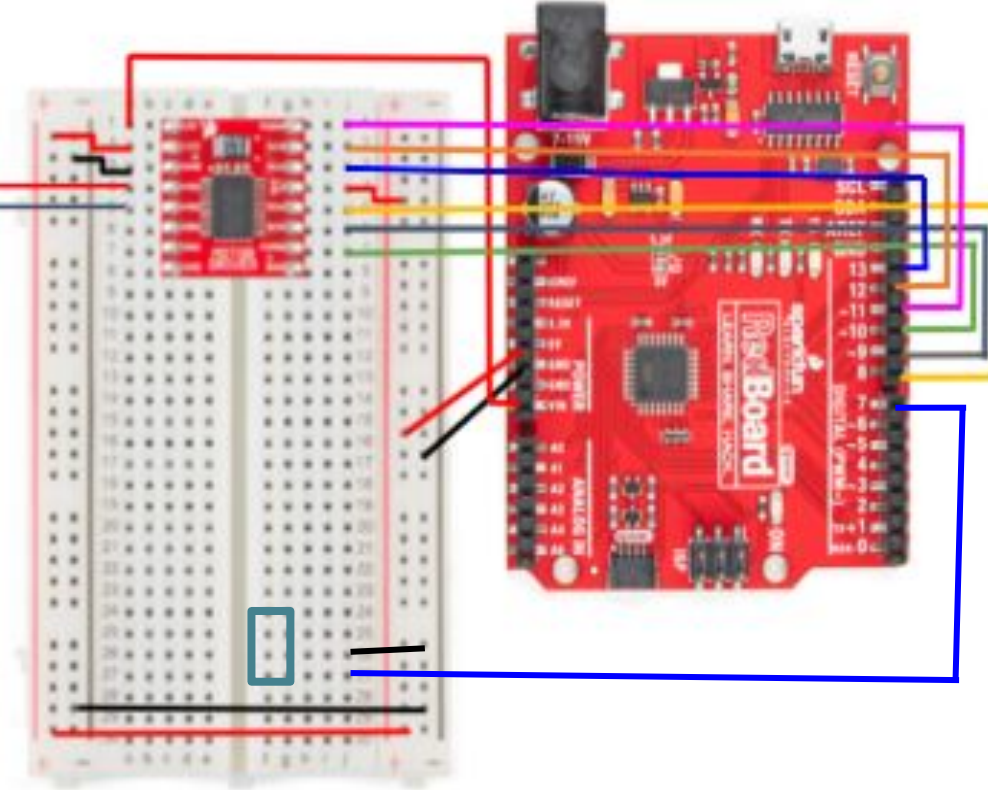
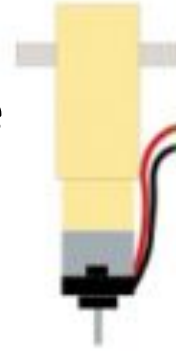
Disconnect the build from the computer (i.e., unplug the USB cord) and place the switch into a breadboard – like clippers we want to turn the rotary motor on and off

- Take a switch  and put in your breadboard
 - Put the legs into **F25**, **F26**, and **F27**



- Take a jumper wire 

- Put one end of the jumper wire into **J26** on your breadboard
- Put the other end of the jumper wire into the **right side of a parallel row** in the **negative (-)** column of your breadboard



- Take a jumper wire 

- Put one end of the jumper wire into **Digital 7** on your RedBoard
- Put the other end of the jumper wire into the **J27** of your breadboard

Please copy the code and paste it into your Arduino Web Editor by creating “NEW SKETCH” to add the switch function:

<https://docs.google.com/document/d/1YPv7U-COS8TwtUjB1NWhXH-5TNznSMmzDUquouUHYIs/edit?usp=sharing>

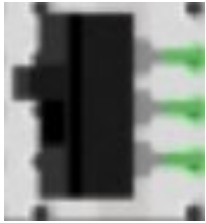
or you can remove the previous code and past this new code

PLUG YOUR MICROCONTROLLER INTO THE COMPUTER



Click  (Verify)

Click  (Upload)



- When the switch is OFF as a picture, the motor stops. When the switch is ON (located downward), the motor will spin after finishing the current loop
- Go study the code, how is it similar to the first code you uploaded, how is it different?