

Practical 4: Controlling the motor

What is it about?

For the last week on this DC motor lab, we want to properly control the speed of our home-made DC motor with the help of an encoder, and H-bridge, and some control logic.

Aims

At the end of the lab, you should:

- have built a proper, closed-loop, motor and controller,
- understand what PWM signals and a H-bridge are,
- know how to use the Arduino's motor shield.

Specific Challenges

- This last lab is really about integration: you'll need a working DC motor and a working encoder.
- Besides wiring these components together, this lab mainly requires you to program the Arduino.



Note

In your lab journal, **describe the basic design of each component, how it was constructed and how it was tested**. Add **pictures** and link to **videos** as needed. Describe as well **the overall system** and how it performs.

And do not forget: **write your lab journal as a text file using the Markdown syntax and push your journal and the pictures on GitHub**.

Preliminary steps

Step 1 – Sign-out an Arduino + motor shield kit

Sign-out and collect from SMB310 an Arduino Uno Kit (Arduino Uno, power supply, motor shield).

You can keep it for as long as you need to finish all the laboratory sessions.

Return the kit (before the end of term!) when you are done.

Step 2 – Finish your DC motor and encoder



Important

This last 'DC motor' lab is about integrating the motor and the encoder together with the Arduino's motor shield to program a closed-loop motor controller.

You need a working DC motor and encoder. Finish first the previous labs and ask for help if you are really late.

Part I

Controlling a small hobby motor

Step 1 – Install the motor shield

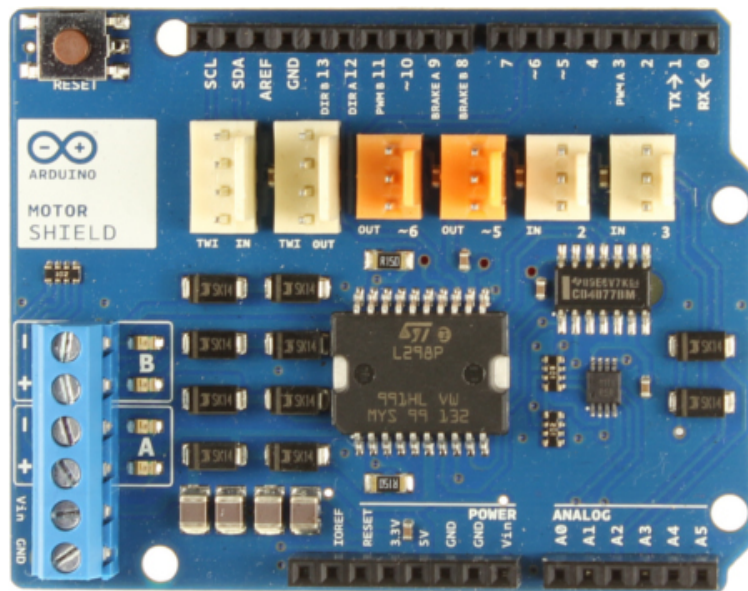


Figure 1: Motor shield. The motor screw terminal connection block is found on the left

This part of this lab is to get you to program an Arduino Uno in conjunction with a motor shield (Figure 1) to control a DC motor.

First, carefully install the motor shield into the Arduino Uno as shown in Figure 2. **Please ask for help if you aren't sure what to do!**

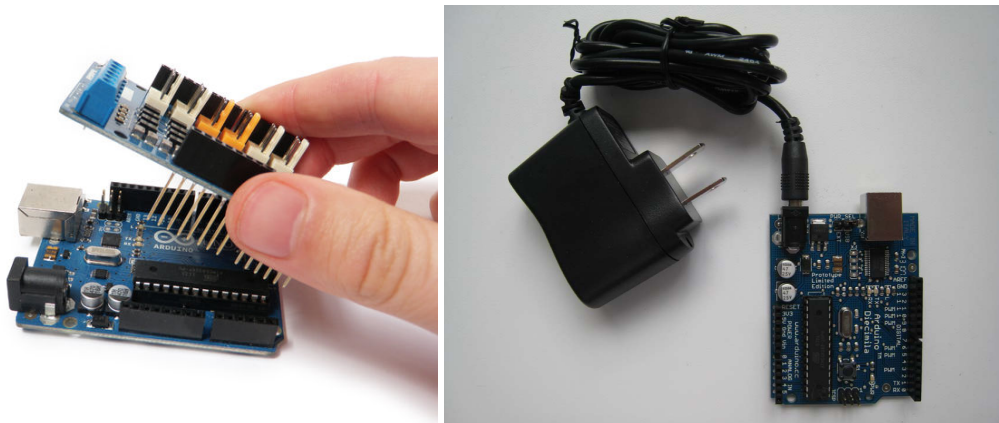


Figure 2: Assembly of the motor shield on top of the Arduino. On the left, the Arduino power supply.

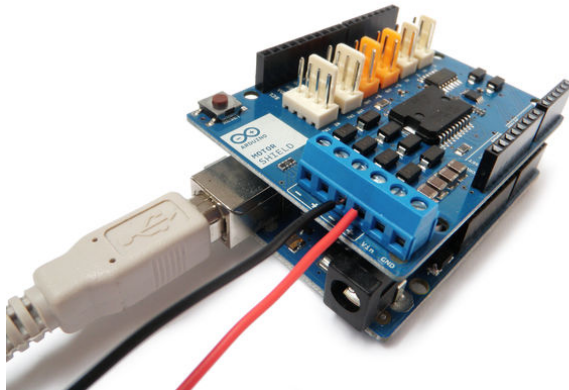


Figure 3: Motor connected to channel A

Then, plug in the external power supply. **Please do not use any other sources of power yet!** We don't want to risk damaging to the boards. We need to use the mains DC power adaptor as the USB power will be insufficient to run the DC motor.

Step 2 – Control a small hobby DC motor

Connect the provided DC motor to the channel A of the motor shield (see Figure 3). If needed, solder first wires to the motor.

Open the Arduino IDE, plug the provided Arduino, and make sure the IDE is correctly configured for your card (check the card type – Arduino UNO – and the port – likely `/dev/ttyACM0`).

Use the following program to get your hobby motor to turn:

```
/******  
Motor Shield 1-Channel DC Motor  
by Randy Sarafan  
  
For more information see:  
https://www.instructables.com/id/Arduino-Motor-Shield-Tutorial/  
*****/  
  
void setup() {  
  
    //Setup Channel A  
    pinMode(12, OUTPUT); //Initiates Motor Channel A pin  
    pinMode(9, OUTPUT); //Initiates Brake Channel A pin  
  
}  
  
void loop(){
```

```
//forward @ full speed
digitalWrite(12, HIGH); //Establishes forward direction of Channel A
digitalWrite(9, LOW);    //Disengage the Brake for Channel A
analogWrite(3, 255);    //Spins the motor on Channel A at full speed

delay(3000);

digitalWrite(9, HIGH); //Engage the Brake for Channel A

delay(1000);

//backward @ half speed
digitalWrite(12, LOW); //Establishes backward direction of Channel A
digitalWrite(9, LOW);  //Disengage the Brake for Channel A
analogWrite(3, 123);   //Spins the motor on Channel A at half speed

delay(3000);

digitalWrite(9, HIGH); //Engage the Brake for Channel A

delay(1000);

}
```



Note

In your lab journal, describe how one configure and use the Arduino motor shield: what is the required setup? How to change the rotation direction? How to change the speed?

Part II

Controlling your DC motor in closed-loop

Step 1 – Get your DC motor to rotate

Now, replace the small provided DC motor by your own motor. You might need to provide more current: use the lab power supply to power the motor shield, but **set the maximum current to 2A** and **do not exceed 12V**, otherwise you will damage the motor shield.

Step 2 – Close the loop

Using your encoder and the Arduino program you wrote last week to read the speed, write a new program that allow to set the desired speed of the motor in RPM, and accordingly control the motor.



Note

Do not forget to document the process!
Insert the code samples (including initial versions which did not work as expected) in your journal.
You can even get syntax highlighting by putting your the code inside a pre-formatted Markdown block like this one:

```
```c
// ...
void loop()
// ...

```
```