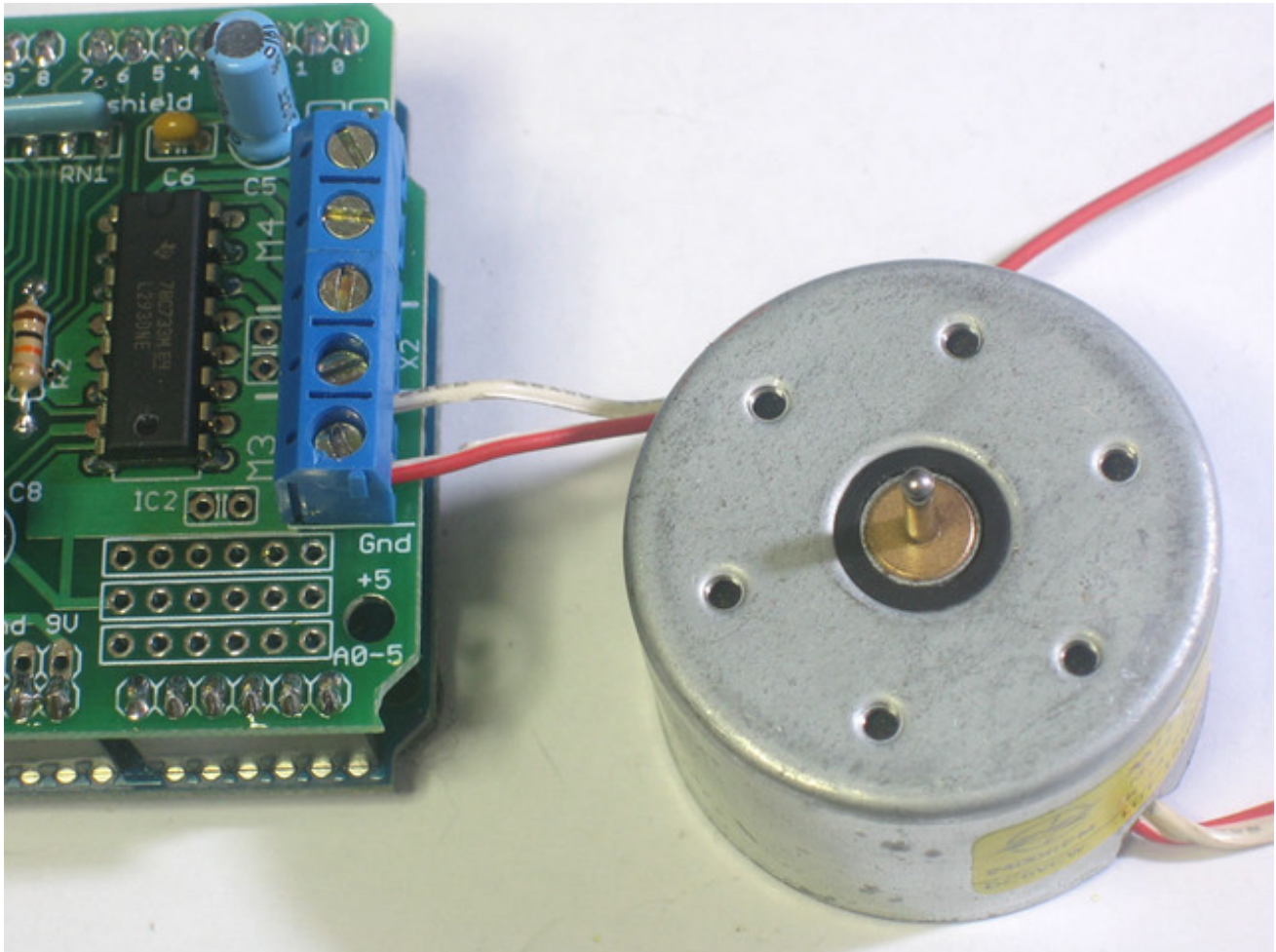


Using DC Motors

This tutorial is for the now ancient V1 Motor shield. Chances are you have a V2, check out the tutorial <https://learn.adafruit.com/adafruit-motor-shield-v2-for-arduino> This tutorial is for historical reference and previous customers only!



DC motors are used for all sort of robotic projects.

The motor shield can drive up to 4 DC motors bi-directionally. That means they can be driven forwards and backwards. The speed can also be varied at 0.5% increments using the high-quality built in PWM. This means the speed is very smooth and won't vary!

Note that the H-bridge chip is not meant for driving loads over 0.6A or that peak over 1.2A so this is for *small* motors. Check the datasheet for information about the motor to verify its OK.

To connect a motor, simply solder two wires to the terminals and then connect them to either the **M1**, **M2**, **M3**, or **M4**. Then follow these steps in your sketch

1. Make sure you `#include <AFMotor.h>`
2. Create the `AF_DCMotor` object with **`AF_DCMotor(motor#, frequency)`**, to setup the motor H-bridge and latches. The constructor takes two arguments.
The first is which port the motor is connected to, **1, 2, 3 or 4**.
`frequency` is how fast the speed controlling signal is.
For motors 1 and 2 you can choose **`MOTOR12_64KHZ`**, **`MOTOR12_8KHZ`**, **`MOTOR12_2KHZ`**, or **`MOTOR12_1KHZ`**. A high speed like 64KHz won't be audible but a low speed like 1KHz will use less power. Motors 3 & 4 are only possible to run at 1KHz and will ignore any setting given
3. Then you can set the speed of the motor using **`setSpeed(speed)`** where the **`speed`** ranges from 0 (stopped) to 255 (full speed). You can set the speed whenever you want.
4. To run the motor, call **`run(direction)`** where **`direction`** is **`FORWARD`**, **`BACKWARD`** or **`RELEASE`**. Of course, the Arduino doesn't actually know if the motor is 'forward' or 'backward', so if you want to change which way it thinks is forward, simply swap the two wires from the motor to the shield.

```
#include <AFMotor.h>

AF_DCMotor motor(2, MOTOR12_64KHZ); // create motor #2, 64KHz pwm

void setup() {
  Serial.begin(9600);      // set up Serial library at 9600 bps
  Serial.println("Motor test!");

  motor.setSpeed(200);    // set the speed to 200/255
}

void loop() {
  Serial.print("tick");

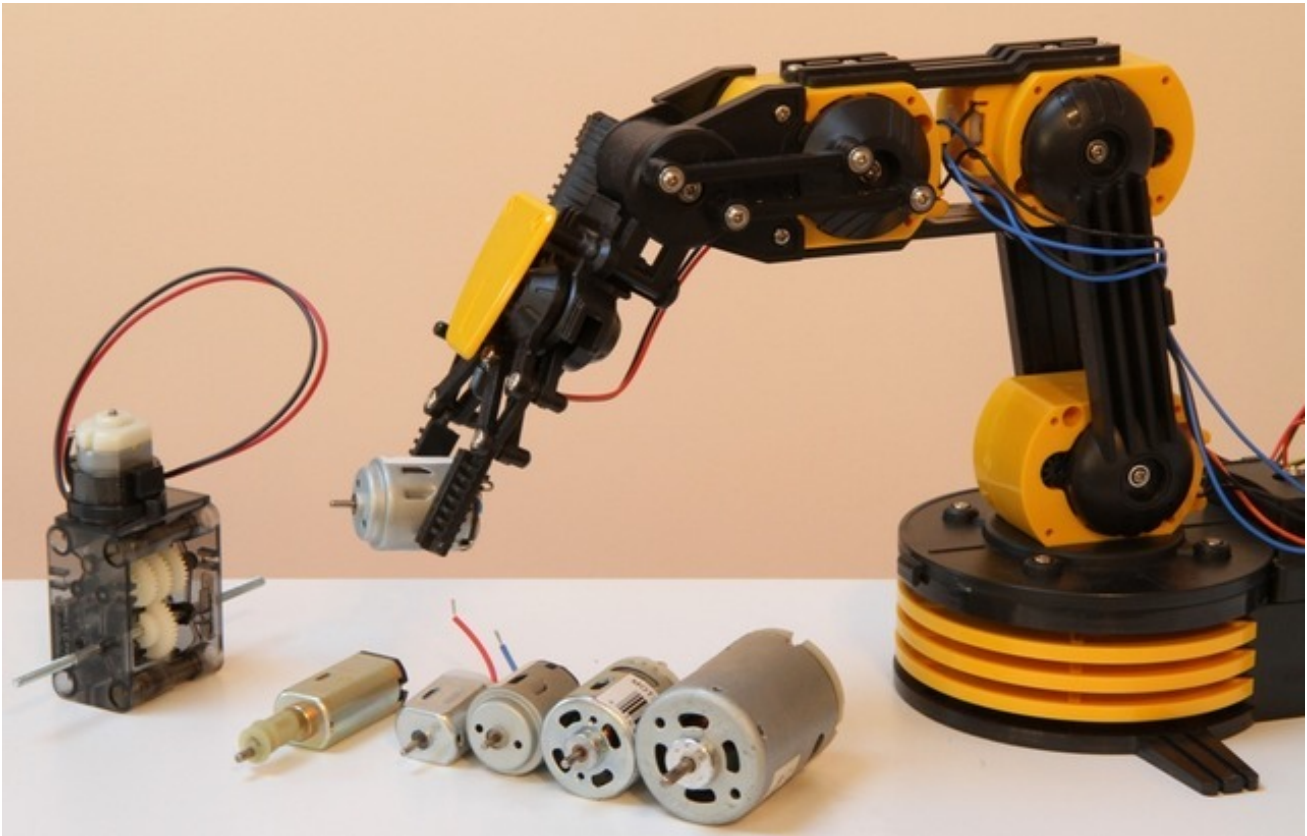
  motor.run(FORWARD);    // turn it on going forward
  delay(1000);

  Serial.print("tock");
  motor.run(BACKWARD);   // the other way
  delay(1000);

  Serial.print("tack");
  motor.run(RELEASE);    // stopped
  delay(1000);
}
```

AF_DCMotor Class

This tutorial is for the now ancient V1 Motor shield. Chances are you have a V2, check out the tutorial <https://learn.adafruit.com/adafruit-motor-shield-v2-for-arduino> This tutorial is for historical reference and previous customers only!



The AF_DCMotor class provides speed and direction control for up to four DC motors when used with the Adafruit Motor Shield. To use this in a sketch you must first add the following line at the beginning of your sketch:

```
#include <AFMotor.h>
```

AF_DCMotor *motorname*(*portnum*, *freq*)

This is the constructor for a DC motor. Call this constructor once for each motor in your sketch. Each motor instance must have a different name as in the example below.

Parameters:

- **port num** - selects which channel (1-4) of the motor controller the motor will be connected to

- **freq** - selects the PWM frequency. If no frequency is specified, 1KHz is used by default.

Frequencies for channel 1 & 2 are:

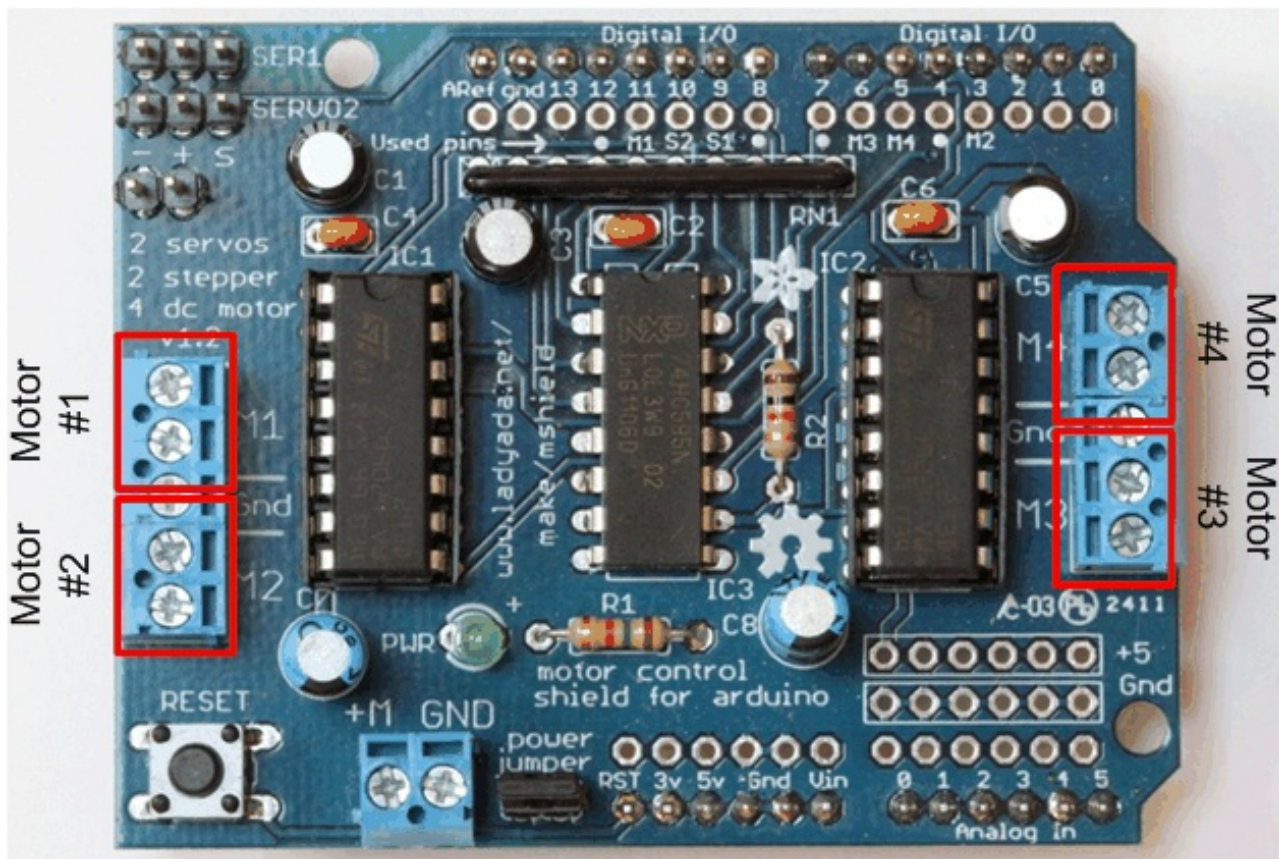
- MOTOR12_64KHZ
- MOTOR12_8KHZ
- MOTOR12_2KHZ
- MOTOR12_1KHZ

Frequencies for channel 3 & 4 are:

- MOTOR34_64KHZ
- MOTOR34_8KHZ
- MOTOR34_1KHZ

Example:

```
AF_DCMotor motor4(4); // define motor on channel 4 with 1KHz default PWM
AF_DCMotor left_motor(1, MOTOR12_64KHZ); // define motor on channel 1 with 64KHz PWM
```



Note: Higher frequencies will produce less audible hum in operation, but may result in lower torque with some motors.

setSpeed(*speed*)

Sets the speed of the motor.

Parameters:

- **speed** - Valid values for 'speed' are between 0 and 255 with 0 being off and 255 as full throttle.

Example:

Note: DC Motor response is not typically linear, and so the actual RPM will not necessarily be proportional to the programmed speed.

run(*cmd*)

Sets the run-mode of the motor.

Parameters:

- **cmd** - the desired run mode for the motor

Valid values for cmd are:

- **FORWARD** - run forward (actual direction of rotation will depend on motor wiring)
- **BACKWARD** - run backwards (rotation will be in the opposite direction from FORWARD)
- **RELEASE** - Stop the motor. This removes power from the motor and is equivalent to setSpeed(0). The motor shield does not implement dynamic breaking, so the motor may take some time to spin down

Example:

```
motor.run(FORWARD);  
delay(1000); // run forward for 1 second  
motor.run(RELEASE);  
delay(100); // 'coast' for 1/10 second  
motor.run(BACKWARDS); // run in reverse
```