

EMT246 I

§ 3. Arduino I/O – Motor Drive & PWM

Dr. Xiaohai Li

xhli@citytech.cuny.edu

Dept. of Computer Eng. Tech
New York City College of Technology

Last update: October 2013

Copyright Notice

- *The slides here include pictures, figures, diagrams, tables and other contents from a variety of sources. Use them for educational purpose ONLY. Copyrights are reserved by their original authors and/or publishers.*
- *Copyright of all other contents is reserved.*

Lecture Objectives

- Learn how to use Arduino to generate an output: to drive a motor
- Prepare the electrical and programming components for the course project

Outline of Contents

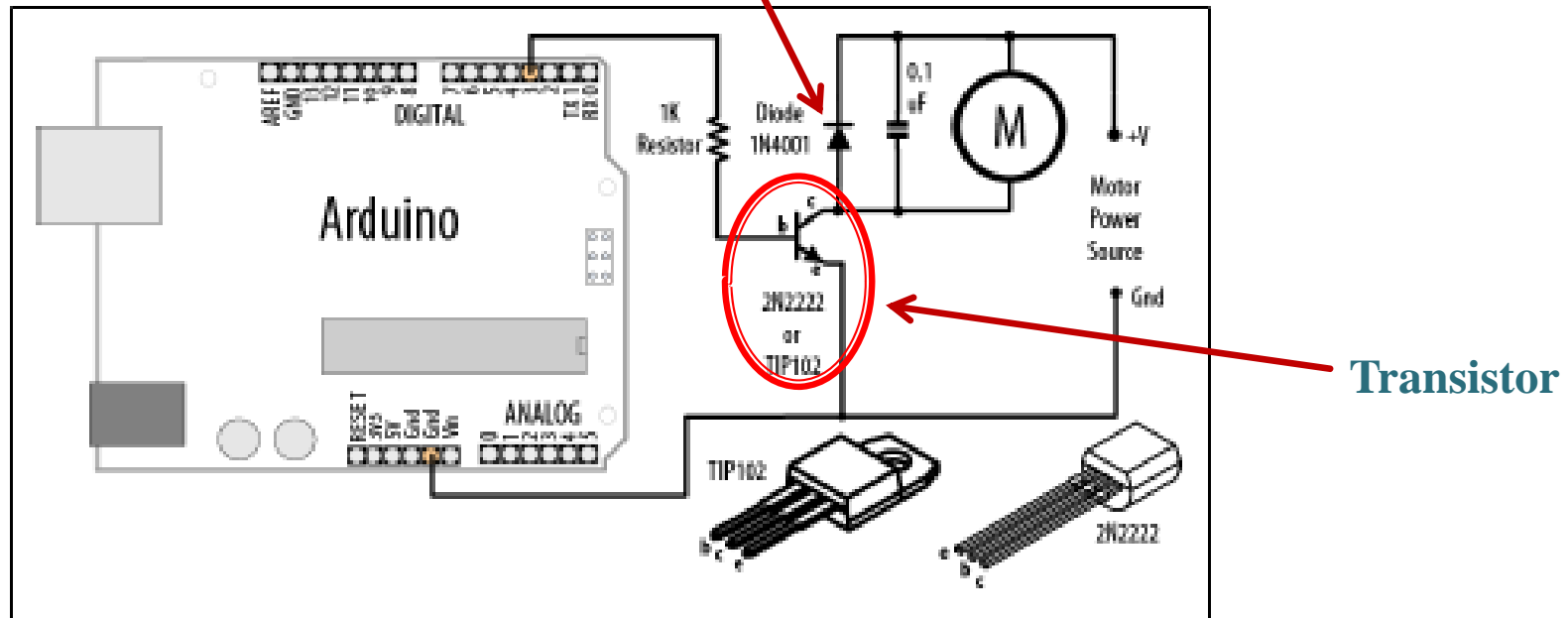
- How to connect a DC Brush Motor to Arduino
- How to use a Transistor to drive a DC Brush Motor
- How to use Arduino to control a DC Brush Motor's Speed.

Outline of Contents

- How to connect a DC Brush Motor to Arduino?
 - ❖ Can you directly connect a DC motor to the 5V or 3.3V pin on Arduino board? Yes or No?
 - ❖ Why Not?
 - ❖ **Do NOT blow your Arduino!!**
 - **Max DC Current per I/O Pin : 40mA**
 - **Max DC Current VCC and GND Pins: 200mA**

Use a Transistor to Drive a Brush Motor

Flyback Diode (also called “freewheel diode”)



- **NEVER** ever connect a load directly with Arduino, especially a load like motor that may draw a large current!
- **Max current** that the 5V/3.3V on Arduino UNO can provide: **200mA**

Example Sketch to Drive a DC Brush Motor

```
* SimpleBrushed sketch
* commands from serial port control motor speed
* digits '0' through '9' are valid where '0' is off, '9' is max speed
*/

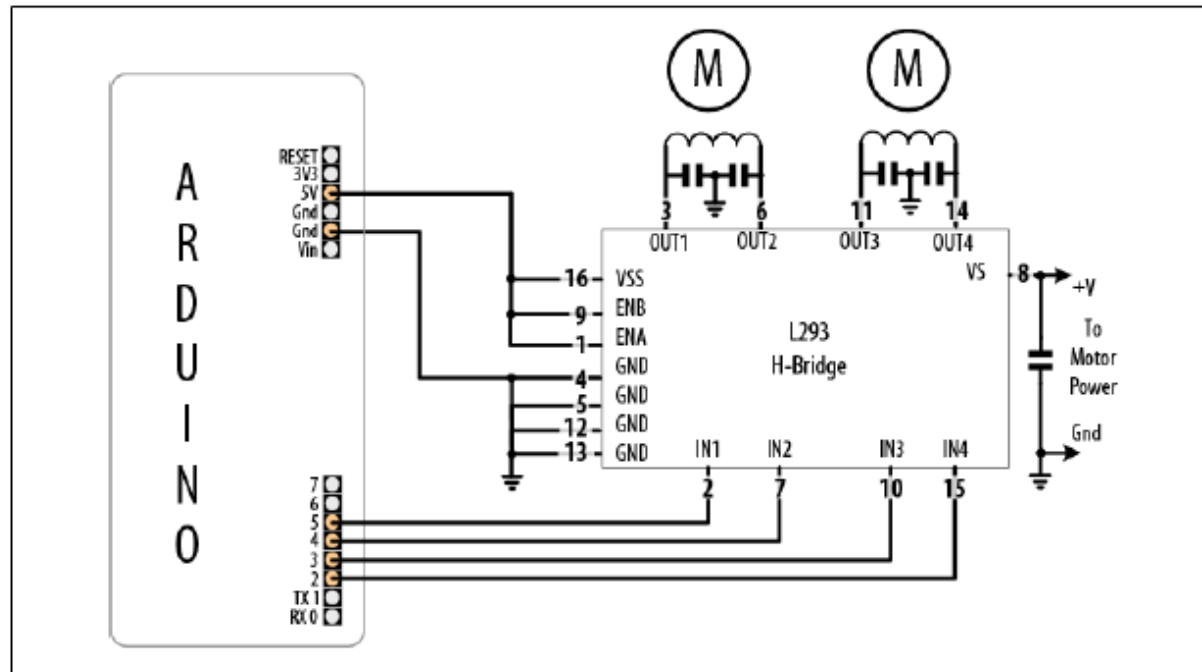
const int motorPins = 3; // motor driver is connected to pin 3

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  if ( Serial.available() ) {
    char ch = Serial.read();

    if(ch >= '0' && ch <= '9')          // is ch a number?
    {
      int speed = map(ch, '0', '9', 0, 255);
      analogWrite(3, speed);
      Serial.println(speed);
    }
    else
    {
      Serial.print("Unexpected character ");
      Serial.println(ch);
    }
  }
}
```

Drive a DC Brush Motor with H-Bridge IC *(will be discussed in another week)*

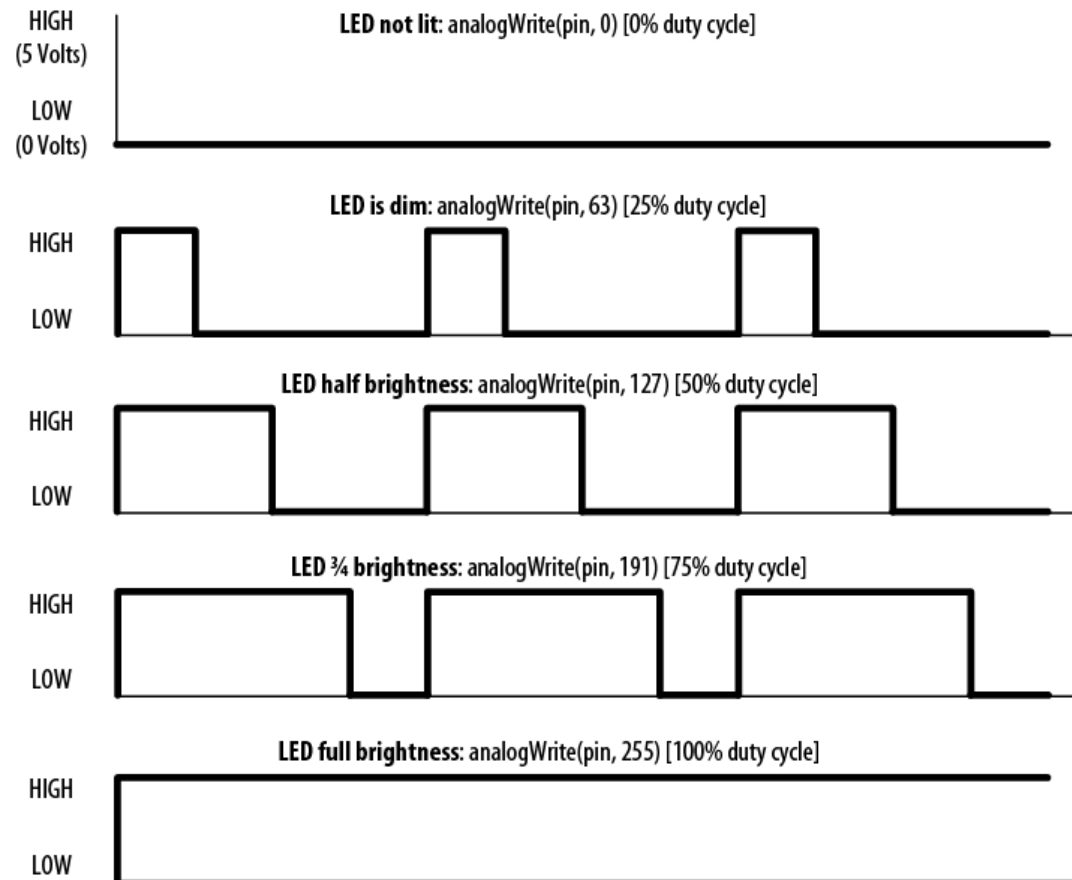


How to Control the Speed of DC Brush Motors

- Ans: **PWM (Pulse Width Modulation)**, the key technique to control motor speed, and many other applications such as power control in your cellphone.
- What is PWM?
- Why PWM works?
- How PWM works? i.e., how to change motor speed by using PWM signal?

Control the Speed of DC Brush Motors

- PWM: **Pulse Width Modulation**



Control the Speed of DC Brush Motors

- **Generate a PWM signal from Arduino:**

```
analogWrite(port#, value); // Write a value (duty cycle of a  
// PWM wave) to an Arduino port.
```

Parameters:

port#: The Arduino port from that the PWM will be produced.

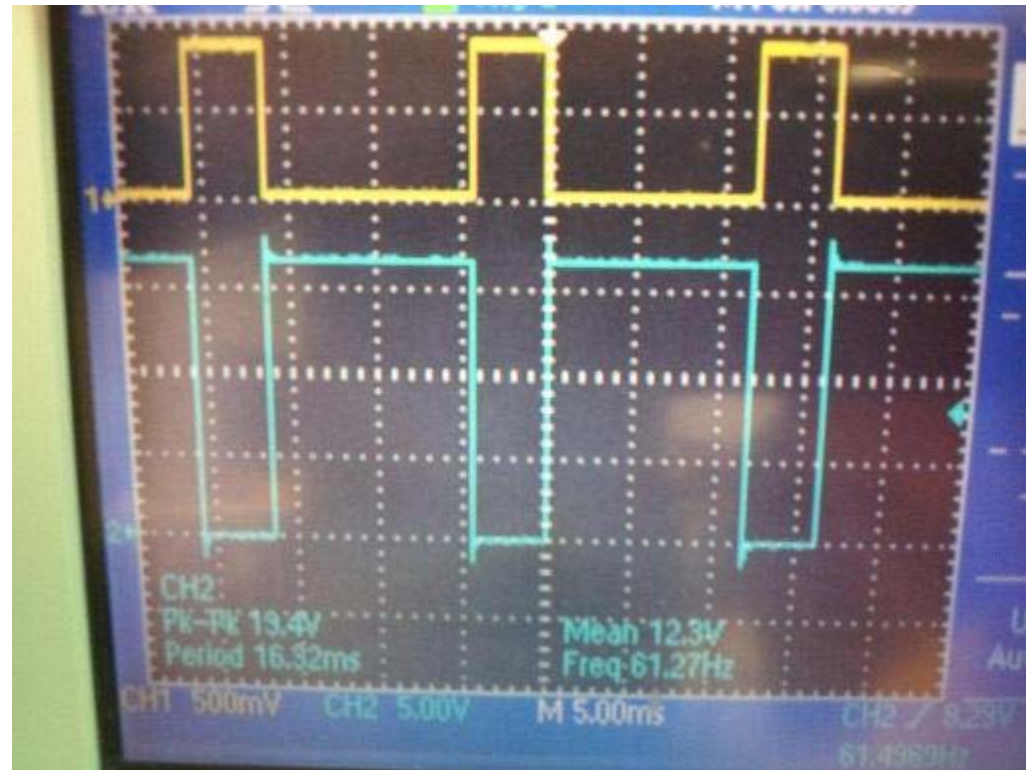
value: Duty cycle: in 0~255! 0: always off and 255: always on.

Returns: nothing

- Example: `analogWrite(3, 127);` // Generate a PWM signal with a Duty cycle of 50% on port3

PWM Waveforms

- Display PWM waveform by oscilloscope



Generate PWM from Arduino (Contd):

- After calling **analogWrite(port#, value)**, the port will generate a steady PWM wave of the specified duty cycle until the next call to **analogWrite()** (or a call to **digitalRead()** or **digitalWrite()** on the same port).

Generate PWM from Arduino (Contd):

- On most Arduino boards (those using the ATmega168 or ATmega328 MCU), like UNO, PWM ports are 3, 5, 6, 9, 10, and 11 by default. On Arduino Mega, PWM works on ports 2 through 13.
- The *analogWrite* function has nothing whatsoever to do with the analog pins or the *analogRead* function.
- The default frequency of PWM signal generated by Arduino UNO is approximately 490 Hz.
- Recommended PWM frequency for industrial applications: 4K or 5KHz ~20KHz.

Example Sketch

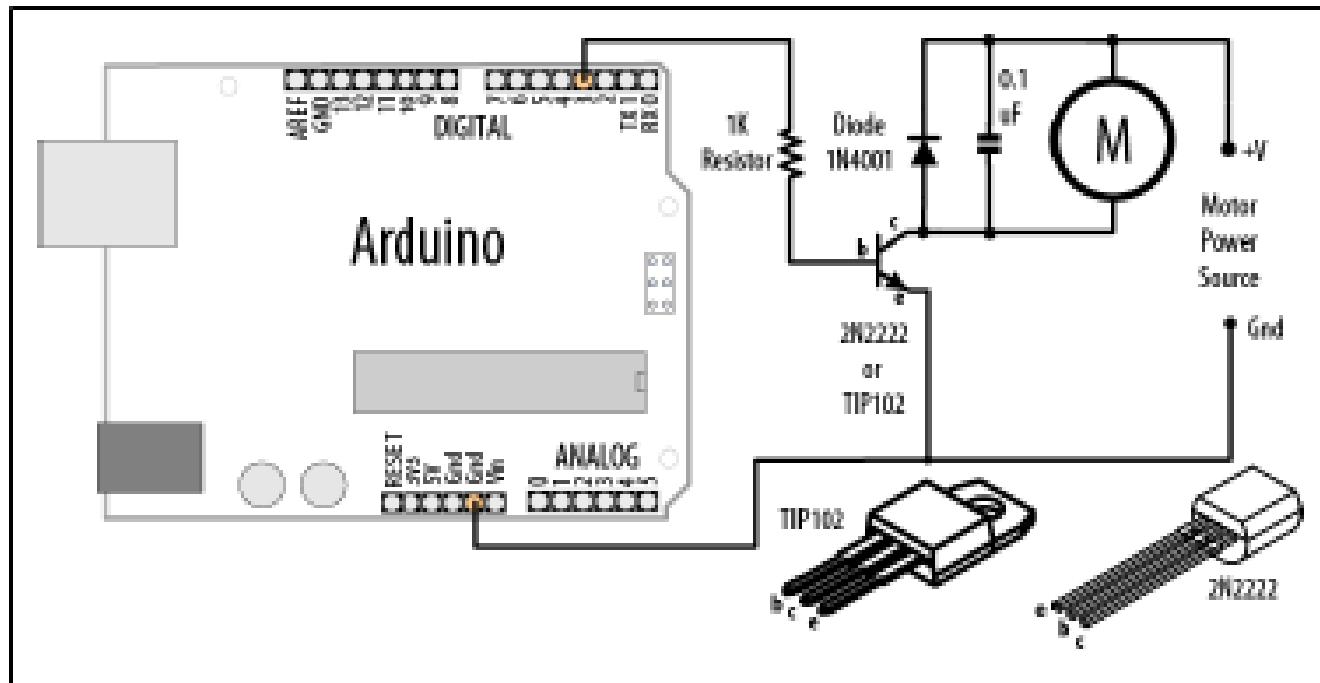
//Sets the output to the LED proportional to the value read from a potentiometer.

```
int ledPin = 9;    // a LED connected to digital port 9
int analogPin = 3; // potentiometer connected to analog port 3
int val = 0;      // variable to store the read value

void setup()
{
  pinMode(ledPin, OUTPUT); // sets the port 9 as output
}

void loop()
{
  val = analogRead(analogPin); // read in inputs from port3
  analogWrite(ledPin, val / 4); // analogRead values go from 0 to 1023, analogWrite
  values from 0 to 255
}
```

Use a Transistor with Arduino to Control a DC Brush Motor's Speed



Note: Only speed control now, no any direction control yet.

Example Sketch to Control a Brush Motor's Speed

```
* SimpleBrushed sketch
* commands from serial port control motor speed
* digits '0' through '9' are valid where '0' is off, '9' is max speed
*/

const int motorPins = 3; // motor driver is connected to pin 3

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  if ( Serial.available() ) {
    char ch = Serial.read();

    if(ch >= '0' && ch <= '9')           // is ch a number?
    {
      int speed = map(ch, '0', '9', 0, 255);
      analogWrite(3, speed);
      Serial.println(speed);
    }
    else
    {
      Serial.print("Unexpected character ");
      Serial.println(ch);
    }
  }
}
```

Lab Experiment

- Lab: PWM & Speed Control of a DC Brush Motor