American International University- Bangladesh

Department of Electrical and Electronic Engineering

EEE4103: Microprocessor and Embedded Systems Laboratory

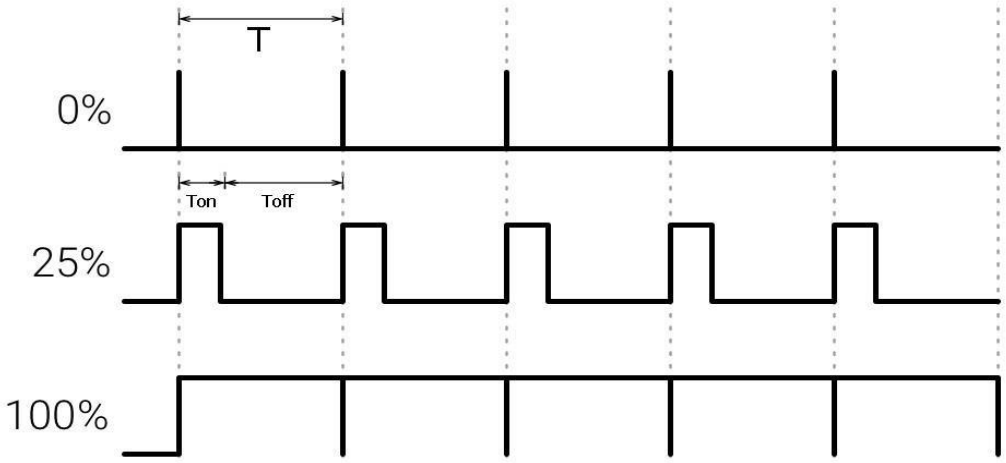
Title: Implementation of a motor control system using Arduino: Digital input, outputs, and PWM

Introduction: The objective of this experiment is to get familiarized with Microcontroller based motor speed control.

Theory and Methodology:

Microcontrollers and Arduino are digital devices; they cannot give analog output. Microcontroller gives Zero and ONE as output, where ZERO is logical LOW and ONE is logical HIGH. In our case, we are using a 5-volt version of the Arduino. So its logical ZERO is zero voltage, and logical HIGH is 5 voltage.

The digital output is good for digital devices but sometimes we need analog output. In such a case the PWM is very useful. In the PWM, the output signal switches between zero and one, on a high and fixed frequency, as shown in the figure below.



Output Signal Of PWM

As shown in the above figure the ON time is T_{on} and the OFF time is T_{off}. **T is the sum of the T_{on} and T_{off}, which is called the Period.** In the concept of PWM, T is not varying and the T_{on} and the T_{off} can vary, in this way when T_{on} increase T_{off} will decrease, and T_{off} increase when T_{on} decrease proportionally.

The duty cycle is a fraction of one Time period. The duty cycle is commonly expressed as a percentage or a ratio. A period is a time it takes for a signal to complete an on-and-off cycle. As a formula, a duty cycle may be expressed as:

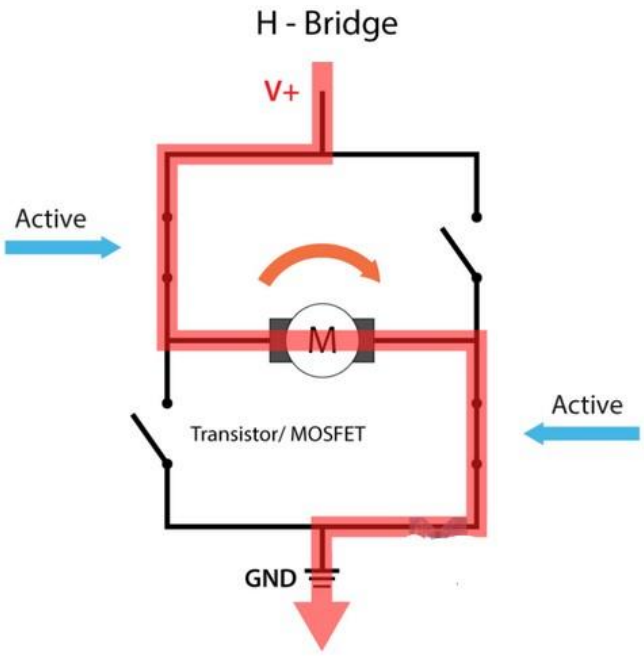
DUTY CYCLE = (Ton / T) x100 %

Now the motor speed varies according to the duty cycle. Suppose the duty is zero, the motor does not run, and when the duty cycle is 100 % the motor moves on maximum RPM**. But this concept is not always right because the motor starts running after giving some fixed voltage that is called threshold voltage.**

Microcontroller and the Arduino can process signals and **consumes almost 20 to 40mA current but motors need high current and voltage, so we are using the transistor for driving the motor.** Transistor is connected in ***series*** with the motor and the ***transistor's base is connected to Arduino's PWM pin through a resistance.*** PWM signal is coming from Arduino and the transistor works as a switch and it **short circuits the Emitter (E) and Collector (C) when the PWM signal is in a High state** and ***normally opens when the PWM signal is in a LOW state***. This process works continuously and the motors run at the desired speed.

H-Bridge DC Motor Control

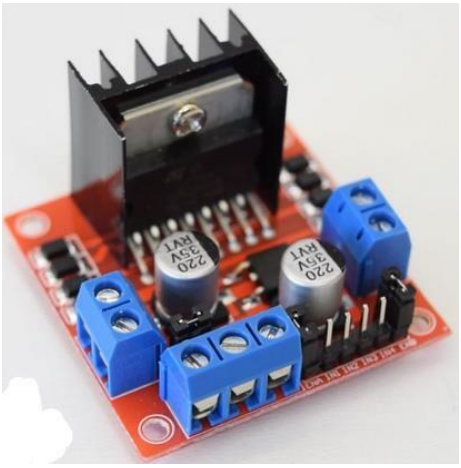
On the other hand, for controlling the rotation direction, we just need to inverse the direction of the current flow through the motor, and the most common method of doing that is by using an H-Bridge. An H-Bridge circuit contains four switching elements, transistors or MOSFETs, with the motor at the center forming an H-like configuration. **By activating two particular switches at the same time we can change the direction of the current flow, thus changing the rotation direction of the motor.**



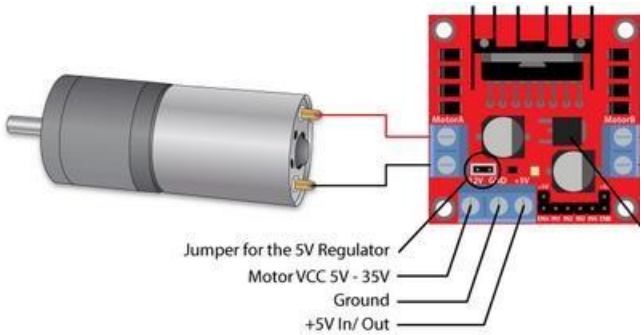
So if we combine these two methods, the PWM and the H-Bridge, we can have complete control over the DC motor. Many DC motor drivers have these features and the L298N is one of them.

L298N Driver

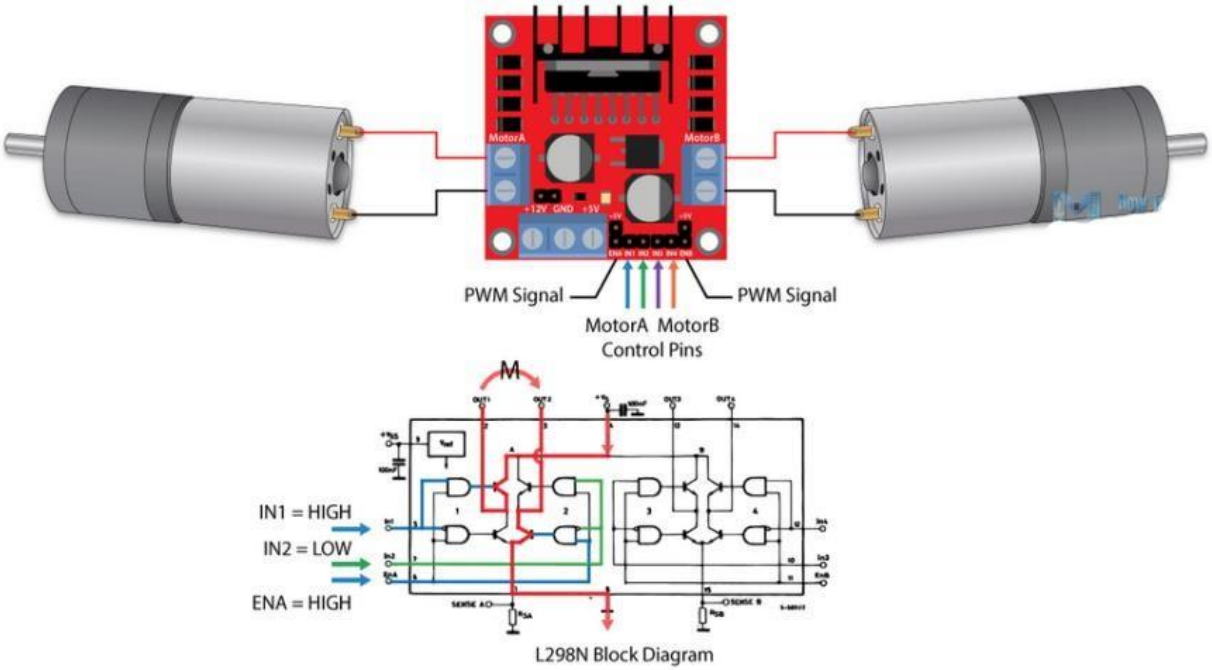
The L298N is a ***dual H-Bridge motor driver which allows speed and direction control of two DC motors at*** the same time. The module can drive DC motors that have voltages between 5 and 35V, with a peak current up to 2A.



Let's take a closer look at the pinout of the L298N module and explain how it works. The module has two screw terminal blocks for motor A and B, another screw terminal block for the Ground pin, the VCC for the motor, and a 5V pin which can either be an input or output.



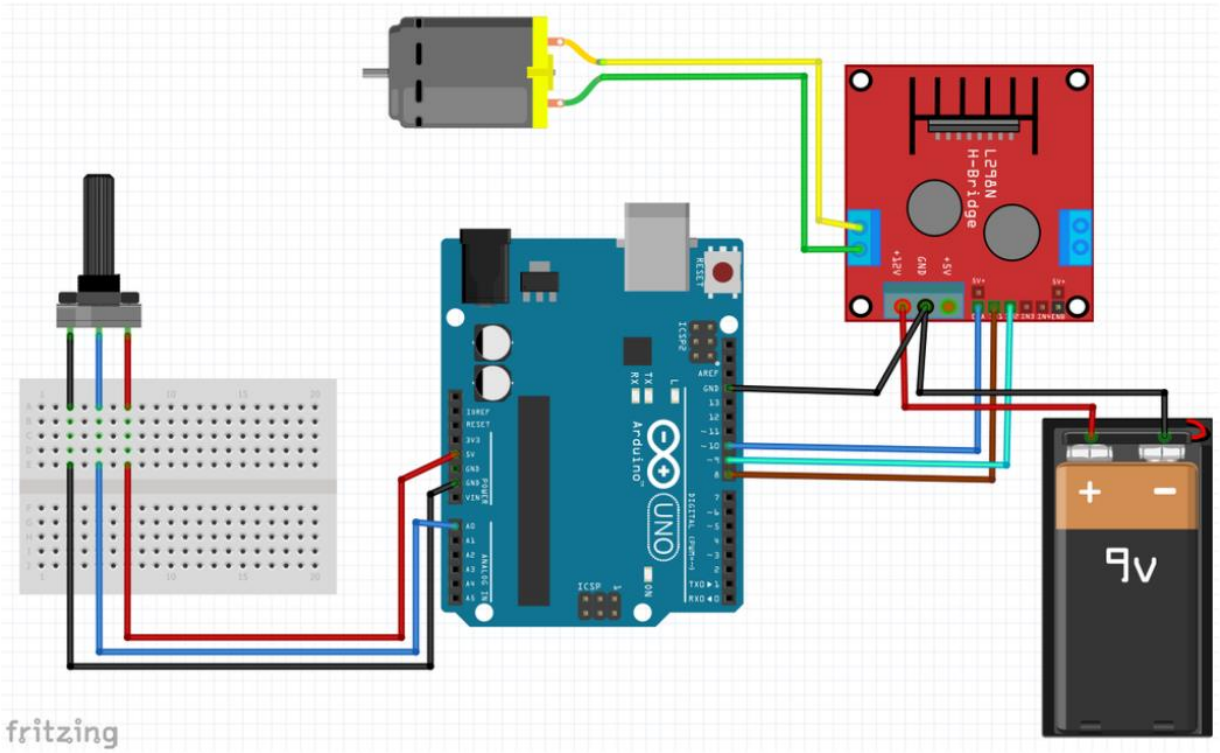
Next are the logic control inputs. The Enable A and Enable B pins are used for enabling and controlling the speed of the motor. ***If a jumper is present on this pin, the motor will be enabled and work at maximum speed, and if we remove the jumper we can connect a PWM input to this pin and in that way control the speed of the motor. If we connect this pin to a Ground the motor will be disabled.***



Next, the Input 1 and Input 2 pins are used for controlling the rotation direction of motor A, and inputs 3 and 4 for motor B. Using these pins we control the switches of the H-Bridge inside the L298N IC. If input 1 is LOW and input 2 is HIGH the motor will move forward, and vice versa, if input 1 is HIGH and input 2 is LOW the motor will move backward. In case both inputs are the same, either LOW or HIGH the motor will stop. The same applies to inputs 3 and 4 and motor B.

Arduino and L298N

Now let's make some practical applications. In the first example, we will ***control the speed of the motor using a potentiometer*** and ***change the rotation direction using a push button***. Here's the circuit schematic diagram.



So we need an L298N driver, a DC motor, a potentiometer, and an Arduino board.

Components List

- L298N Driver
- 12V High Torque DC Motor
- Arduino Board
- Potentiometer
- A power supply
- Breadboard and Jump Wires

Program to be written in ArduinoIDE:

int in1 = 8; //*Declaring where our module is wired*

```
int in2 = 9;
int ConA = 10;// Don't forget this is a PWM DI/DO
int speed1;

void setup() {

Serial.begin(9600);
pinMode(8, OUTPUT);
pinMode(9, OUTPUT);
pinMode(10, OUTPUT);
}

void TurnMotorA(){ //A function to control the direction and speed
digitalWrite(in1, LOW); //Switch between this HIGH and LOW states to change direction
digitalWrite(in2, HIGH);
speed1 = analogRead(A0);
speed1 = speed1*0.2492668622; //Analog value is read from the potentiometer to calibrate it
analogWrite(ConA,speed1);// To activate the motor
}
void loop() {
 int value = analogRead(A0); //declaring and reading value from the pin
value = value*0.2492668622; // doing calibration to change range from 0-1023 to 0-255 the number and is obtained by 255/1023

Serial.println(value);
TurnMotorA(); //one function that keeps looping you can add another one with a different direction or stop

}
```

Questions for report writing:

1) Include all codes and scripts into the lab report following the writing template mentioned in appendix A of Laboratory Sheet Experiment 8.
2) Implement the system in the proteus simulation tool by following the link: https://youtu.be/q_B3yAM4PH0