Lab Manual for Embedded System Design

Lab No. 7

Interfacing DC Motor with Arduino & PWM

Objectives

In this lab students are introduced to Pulse width modulation (PWM), motor driver and interfacing of DC motor with Arduino Uno

LAB # 7
PWM & DC motor interfacing with Arduino Uno

Introduction

Pulse width modulation (PWM):

Pulse width modulation (PWM) is an important feature in Arduino Uno, it is used to control the brightness of LED or speed of a DC motor by varying the duty cycle of PWM. Figure 7.1 shows PWM signals with different duty cycles.
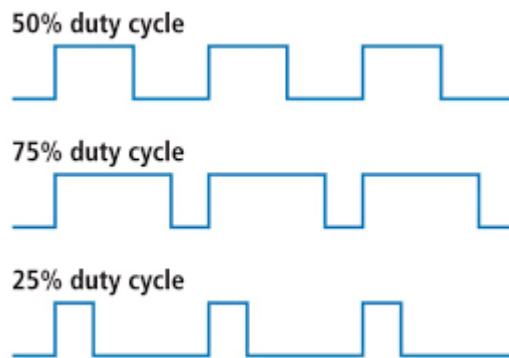

Fig 7.1: PWM signals

Arduino Uno has pin no 3, 5, 6 9,10 & 11 that can be used to generate a PWM signal , these pins are labelled with (~) sign on Arduino Uno board. Arduino IDE has a built in function analogWrite () that is used to generate PWM signal.

analogWrite (0) means a signal of 0% duty cycle.

analogWrite (127) means a signal of 50% duty cycle.

analogWrite(255) means a signal of 100% duty cycle.

DC Motor Driver:

DC motor can be interfaced to Arduino Uno through L298N motor driver. L298N is a dual channel H-bridge IC capable of controlling two Dc motors. Simplified circuit diagram of H-bridge is shown in figure 7.2:
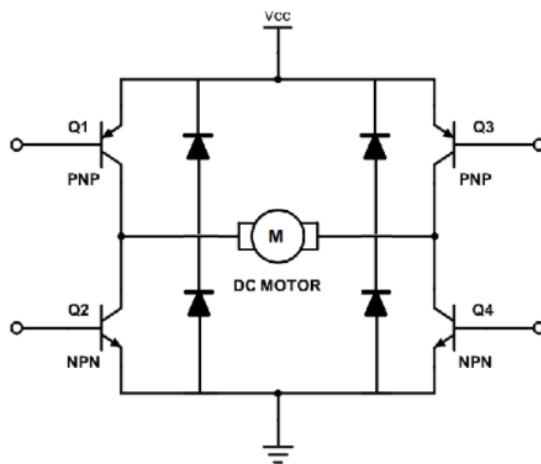
Fig 7.2: Single channel H-bridge.

When transistors Q1 & Q4 are on and Q2 & Q3 are off, motor runs in clock wise direction, similarly when Q1 & Q4 are off and Q2 & Q3 are on, motor runs in anti-clock wise direction. Diode across the transistors are to prevent back EMF from the motor.

Time Boxing

Activity Name	Activity Time	Total Time
Login Systems + Setting up Proteus & Arduino Environment	3 mints + 5 mints	8 mints
Walk through Theory & Tasks	60 mints	60 mints
Implement Tasks	80 mints	80 mints
Evaluation Time	30 mints	30 mints
	Total Duration	178 mints

Objectives

To Study the PWM generation & DC motor Interfacing with Arduino Uno.

Lab Tasks/Practical Work

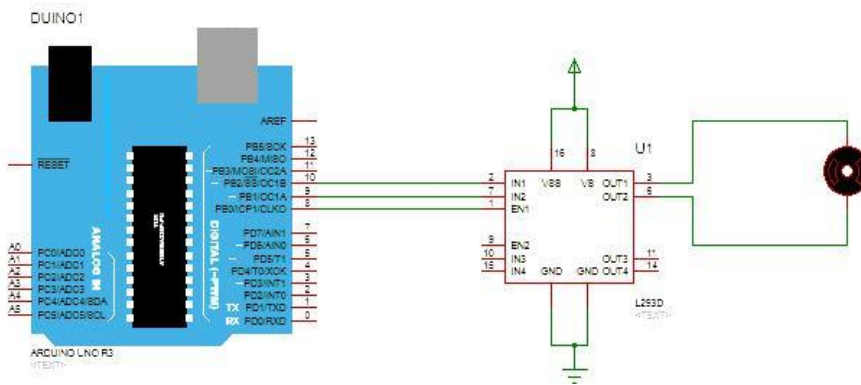**1. Write a program to generate a PWM signal with duty cycles (25%, 50%, 75% & 100%)**
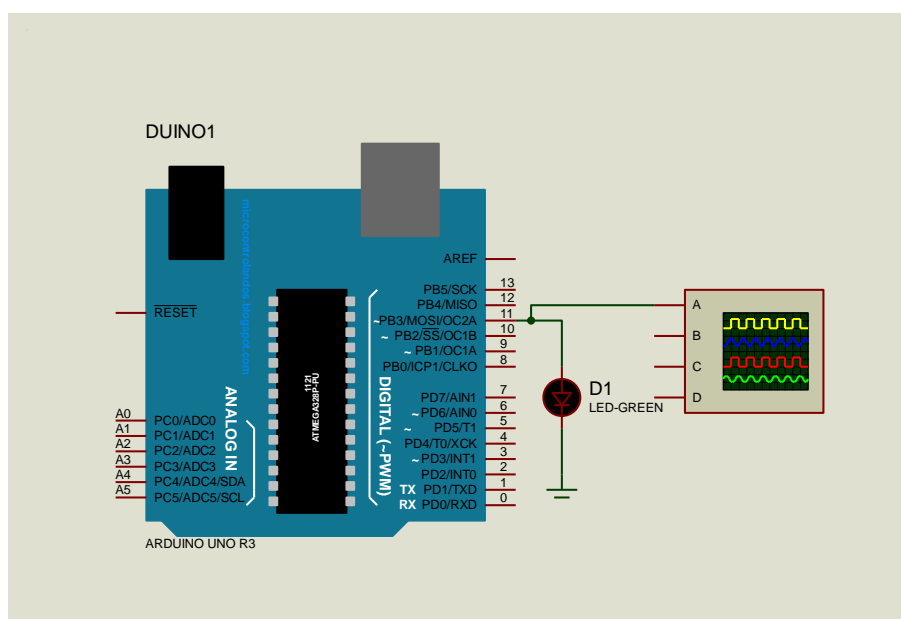
Code:
void setup() {
pinMode(11,OUTPUT);
}

void loop() {

analogWrite(11,63.75);   / 25 % duty cycle
delay(200);
analogWrite(11,127);   // 50 % duty cycle
delay(200);
analogWrite(11,191);   // 75 % duty cycle
delay(200);
analogWrite(11,255);   // 100 % duty cycle
delay(200);

}

2. Write a program to interface DC motor with Arduino Uno.

Code:
int in1=10;
int in2=9;
int en=8;

void setup()
{
        pinMode(in1, OUTPUT);
        pinMode(in2, OUTPUT);

        . pinMode(en, OUTPUT);

}

43

```
void loop()
{
        digitalWrite(en, HIGH);                  // Activating the Channel 1 of L293D.
        digitalWrite(in1, HIGH);                 // Making the in1 at HIGH Logic Level.
        digitalWrite(in2, LOW);                  // Making the in2 at LOW Logic Level.
        delay(5000);                             // Wait for 5 seconds.
        digitalWrite(en, LOW);                   // De-Activating the Channel 1 of L293D.
        delay(1000);                             // Wait for 1 seconds.
        digitalWrite(en, HIGH);                  // Activating the Channel 1 of L293D.
        digitalWrite(in2, HIGH);                 // Making the in2 at HIGH Logic Level.
        digitalWrite(in1, LOW);                  // Making the in1 at LOW Logic Level.

        delay(5000);

        digitalWrite(en, LOW);     // De-Activating the Channel 1 of L293D.

        delay(1000);
}
```

3. Write a program in Arduino Uno to control the speed of DC motor using PWM.