Exp. 04 - Motor Driver and Interfacing

Part 1 – Auto Transmission using an L293D motor driver

Equipment List:

1. 1x Arduino Nano
2. 1x USB 2.0 to USB type A connecting cable
3. 1x Breadboard
4. 2x/4x small DC Motor for tires
5. 1x L293D motor driver
6. 1x 3/4 wheel motor chassis

Schematic Diagram:

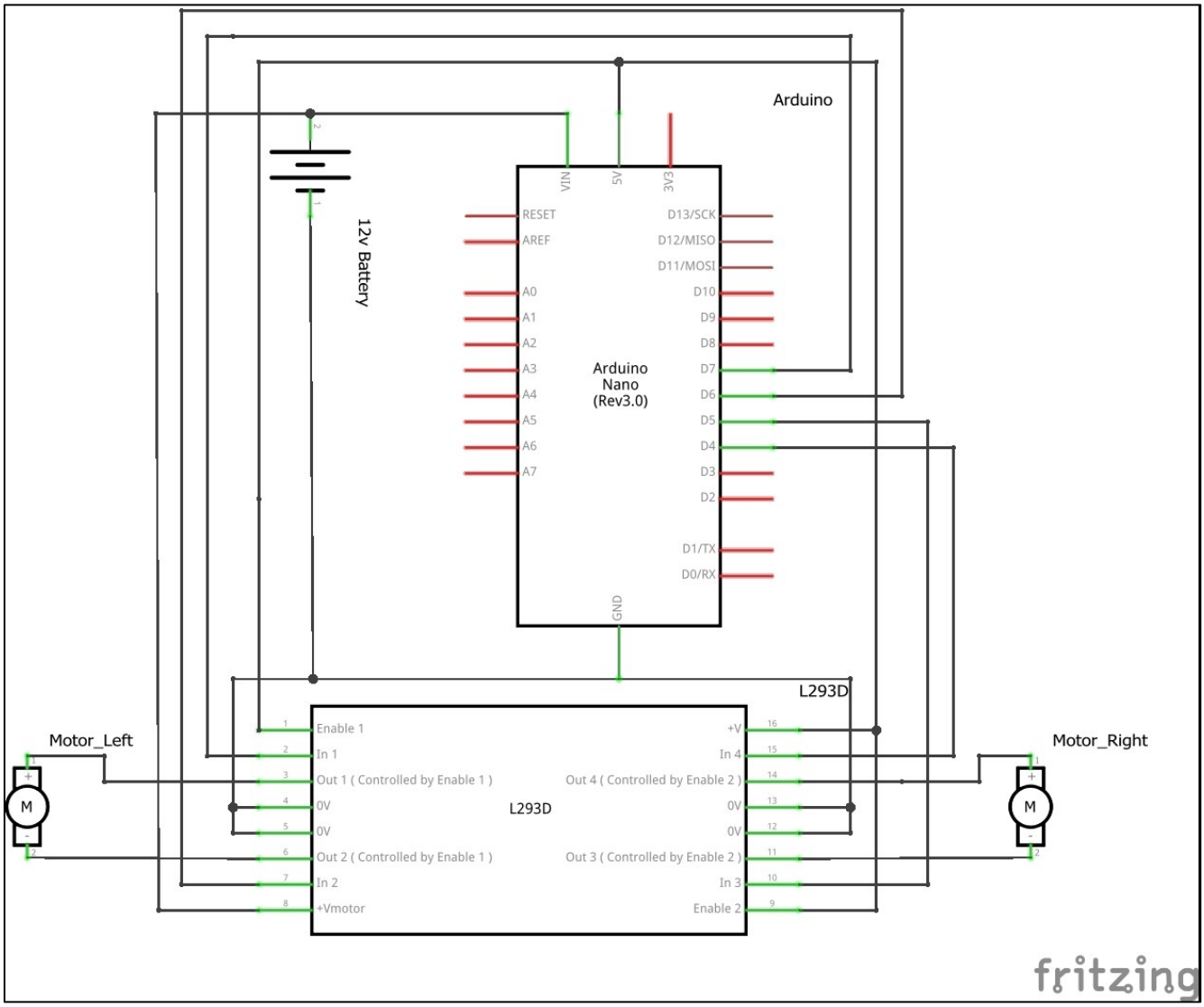

Figure 1 - Exp. 04 - Part 1 – L293D Motor driver interfacing

Experiment Setup: Connect the PINS as per the following –

Connection of L293D motor driver

- PIN 1,9 and 16 with 5V output of Arduino
- PIN 4,5 and 12,13 with GROUND PIN of battery
- PIN 3,6 with LEFT_MOTOR
- PIN 11,14 with RIGHT_MOTOR
- PIN 2,7 with Arduino Digital PIN 7,6
- PIN 10,15 with Arduino Digital PIN 5,4
- PIN 8 to 12V of battery.

Connection of Arduino

- PIN 2,7 of L293D with Arduino Digital PIN 7,6
- PIN 10,15 of L293D with Arduino Digital PIN 5,4
- PIN 1, 9 and 16 of L293D with 5V output of Arduino.
- VIN of Arduino to 12V Battery and GROUND of Arduino to GROUND of Battery.
- Upload the code and check the Output

Expected Output:

- The car should move forward, backward, left and right and repeat the sequence.

Code:

```
int motor_left[] = {7, 6};
int motor_right[] = {5, 4};

void setup() {
      int i;
      for(i = 0; i < 2; i++){
            pinMode(motor_left[i], OUTPUT);
            pinMode(motor_right[i], OUTPUT);
      }
}
void loop() {
      drive_forward();
      delay(1000);
      motor_stop();
      drive_backward();
      delay(1000);
      motor_stop();
      turn_left();
      delay(1000);
      motor_stop();
      turn_right();
      delay(1000);
      motor_stop();
      motor_stop();
      delay(1000);
      motor_stop();
}
```

```
void motor_stop(){
      digitalWrite(motor_left[0], LOW);
      digitalWrite(motor_left[1], LOW);
      digitalWrite(motor_right[0], LOW);
      digitalWrite(motor_right[1], LOW);
      delay(25);
}
void drive_forward(){
      digitalWrite(motor_left[0], HIGH);
      digitalWrite(motor_left[1], LOW);
      digitalWrite(motor_right[0], HIGH);
      digitalWrite(motor_right[1], LOW);
}
void drive_backward(){
      digitalWrite(motor_left[0], LOW);
      digitalWrite(motor_left[1], HIGH);
      digitalWrite(motor_right[0], LOW);
      digitalWrite(motor_right[1], HIGH);
}
void turn_left(){
      digitalWrite(motor_left[0], LOW);
      digitalWrite(motor_left[1], HIGH);
      digitalWrite(motor_right[0], HIGH);
      digitalWrite(motor_right[1], LOW);
}
void turn_right(){
      digitalWrite(motor_left[0], HIGH);
      digitalWrite(motor_left[1], LOW);
      digitalWrite(motor_right[0], LOW);
      digitalWrite(motor_right[1], HIGH);
}
```

Code Listing 1 - Code for interfacing DC motors with L293D driver

Appendix:

The L293 and L293D are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications. All inputs are TTL compatible. Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1, 2 EN and drivers 3 and 4 enabled by 3, 4 EN. When an enable input is high, the associated drivers are enabled, and their outputs are active and in phase with their inputs. When the enable input is low, those drivers are disabled, and their outputs are off and in the high-impedance state. With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

Reference: "L293D Motor Driver IC | L293D Datasheet", Engineersgarage.com, 2019. [Online]. Available: https://www.engineersgarage.com/electronic-components/l293d-motor-driver-ic. [Accessed: 17- Jun- 2019].