LAB 7: IMPLEMENTING ACTUATORS

Objective

Learn to supply a pulse width modulated (PWM) signal using a microcontroller, incorporate an H-bridge chip, and drive a DC motor.

Introduction

Prelab Assignment

1. Review the L298 spec sheet on the course website (under the "Lab Resources" heading).
2. Using the L298 truth table, what would be the result of applying a PWM signal to pin In1, while holding pin In2 LOW and pin EnA HIGH? What about applying a PWM signal to pin In2 while holding pin In1 LOW and pin EnA HIGH? Why do you think that the PWM signal is usually applied to the EnA pin, rather than pins In1 or In2?
3. Develop a state transition diagram for Step 3.
4. Develop code to implement Step 3.

H-Bridge Circuit

H-bridge circuits are often used to control the speed and direction of DC motors. A simplified H-bridge schematic is shown in Figure 1. Note that each transistor is paired with a *flyback diode*. These diodes protect the transistors when switching occurs. At the instant that a transistor switch opens, current passing through the transistor quickly goes to zero, causing $\frac{dI}{dt} < 0$. Motor windings act as inductors, so voltage across the motor becomes $L\frac{dI}{dt} < 0$. This results in large voltage spikes across the transistors. For the configuration shown in Figure 1, the diodes are forward biased, such that current passes through the diodes (and not the transistors) when voltage spikes occur. Some commercial H-bridge packages include these flyback diodes, and other *do not*, so always consult the spec sheets! The table below shows the corresponding motor direction for digital inputs C and D. *Note: This table is for the simplified operation shown in Figure 1. Always refer to a spec sheet for the integrated circuit you are using to determine an appropriate logic table.*

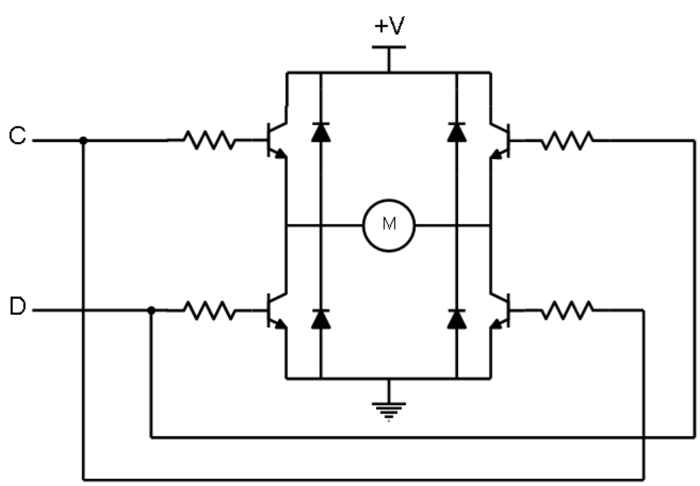

Figure 1: Simplified H-bridge schematic.

C	D	Dir
0	0	Stop
0	1	CCW
1	0	CW
1	1	N/A

In this lab, you will work with the L298 full bridge chip. Its internal wiring structure can be seen in Figure 2. Note the differences between the basic H-bridge truth table (shown above) and the L298 implementation. Each chip is comprised of two H-bridges, but lacks flyback diodes. The L298 driver can operate on voltages up to 50 V, which is more than adequate for actuators used in this lab. Note the 100 nF capacitors and 22 Ω resistors (R_{SA} and R_{SB}) shown in the diagram. The 22 Ω resistors are current limiting resistors and have been chosen very conservatively. If the motor draws current but does not spin well, it may be necessary to reduce the current limiting resistor. Consult the spec sheet to determine an appropriate value.

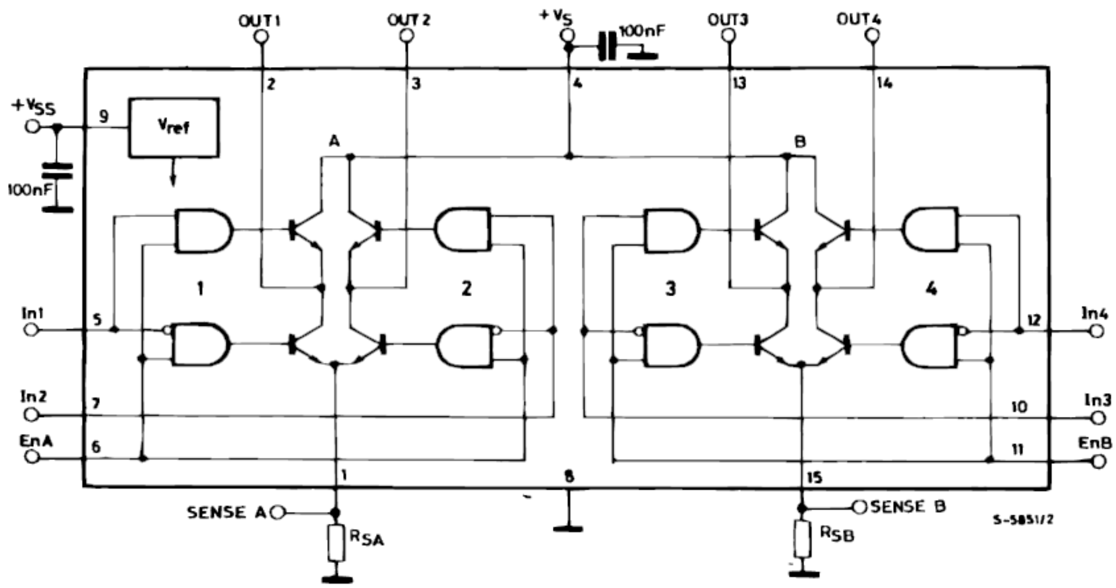

Figure 2: L298 Circuit.

Table 1: L298 Truth Table

	Inputs	Function
	In1 = H; In2 = L	Forward
EnA = H	In1 = L; In2 = H	Reverse
	In1 = In2	Fast Motor Stop
EnA = L	In1 = X; In2 = X	Free Running

Procedure

1. **Dimming an LED with a PWM Output**

   Appendix A provides sample code that takes an analog input (between 0 and 5 V) and converts it into a proportional PWM output. Load this code into your microcontroller. Use a potentiometer for the analog input, and an LED as an output.

   (a) Does your code work as expected?

   (b) What happens to the LED as the duty cycle of the PWM signal increases?

   (c) How might you use this in other projects?

2. **Implementing a Transistor to Drive a DC Motor**

   Implement a transistor circuit to drive a DC motor. The motor should be connected to the collector of the NPN transistor, with the PWM output from the microcontroller connected to the transistor base. Include a flyback diode as shown. Your circuit should look like the schematic in Figure 3.

   (a) Does your code work as expected?

   (b) What effect does the PWM duty cycle have on the motor?

   (c) Can you control the direction of the motor with this circuit?

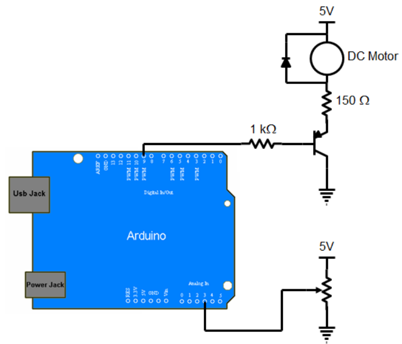

Figure 3: DC motor driven by a transistor using PWM output.

3. **Driving a Motor**

   Drive a motor using the L298 with one PWM output and two digital outputs from the Arduino microcontroller. Implement two pushbuttons such that the first push button starts the motor with CW rotation and the second pushbutton starts the motor with CCW rotation. If neither button is pressed or both buttons are pressed, the motor should be stopped. Use a potentiometer to control the duty cycle of the PWM signal. Be sure to incorporate flyback diodes, capacitors and current-limiting resistors with the L298 chip. Initially use 5V on pin 4 ($+V_S$) of the L298 to power your DC motor. *Be careful when wiring in lab, as large amounts of current can flow through the L298 chip, especially as the supply voltage is increased.*

   Hint: Verify your code using LEDs to represent the states in your state machine. Do this with the motor power supply turned off to avoid damage to the L298 or motor. Debug your code as necessary before powering up the motor.

   (a) Does your code work as expected?

   (b) What is the effect of changing the potentiometer setting?

   (c) How does this method compare to using only a single transistor to control the DC motor?

   (d) What is the effect of increasing the motor supply voltage from 5 to 24 volts? (Be sure to avoid increasing the potentiometer source voltage as you do this!)

   (e) How might you use this in other projects?

   (f) Demonstrate your device for the instructor.

   (g) Include a state transition diagram for your circuit in your report.

4. **Bonus: Controlling a Motor Based on Distance to an Object (continued from Lab 6)**

   Build a finite state machine to read an analog IR sensor and then output motor commands (go/stop, forward/backward, slow/fast) based on the distance to an object. Use the logic shown below. The "Emergency" input may be a push button, optical interrupter or other device to indicate a crisis condition. Implement your state machine with an analog IR sensor, H-bridge and a DC motor.

Sensing	Motor Response
No Object	Fast Forward
Object Visible	Slow Forward
Object Close	Slow Backward
Emergency	Stop

Table 2: Motor control logic

   (a) Does your code work as expected?

   (b) Demonstrate your device for the TA.

   (c) Include a state transition diagram in your report.

5. **Bonus: Stall Detection**

   What is the relationship between the current that the motor is drawing and the current sensing outputs (pins 1 and 15) of the L298? Use the current sensing outputs of the L298 to detect when the motor is stalled and trigger an Emergency Stop state from your finite state machine for Step 4.

Appendix A Sample PWM Code

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

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

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