Lab 6: Stepper Motor

Matthew Friedman 861151348
Souradeep Bhattacharya 861105938
EE128 Section: 021

November 06, 2017

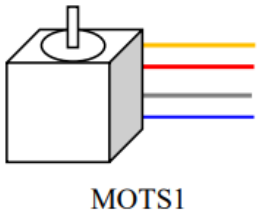
Abstract

The objective of this lab was to become familiar with the control of stepper motors. We create a program to control a stepper motor's direction and speed based off of two input switches.

Experimental System Specification

Create a system using the L298N that will control the direction (clock wise and counter-clockwise) and the speed of the motor (22.5 degrees per second and 90 degrees per second) based on the input of two switches.

Block Diagram



A1 (yellow)
B1 (red)
A2 (white)
B2 (blue)

Step	A1	A2	B1	B2
0	-	+	+	-
1	+	-	+	-
2	+	-	-	+
3	-	+	-	+

Figure 1: Stepper motor pin block diagram.

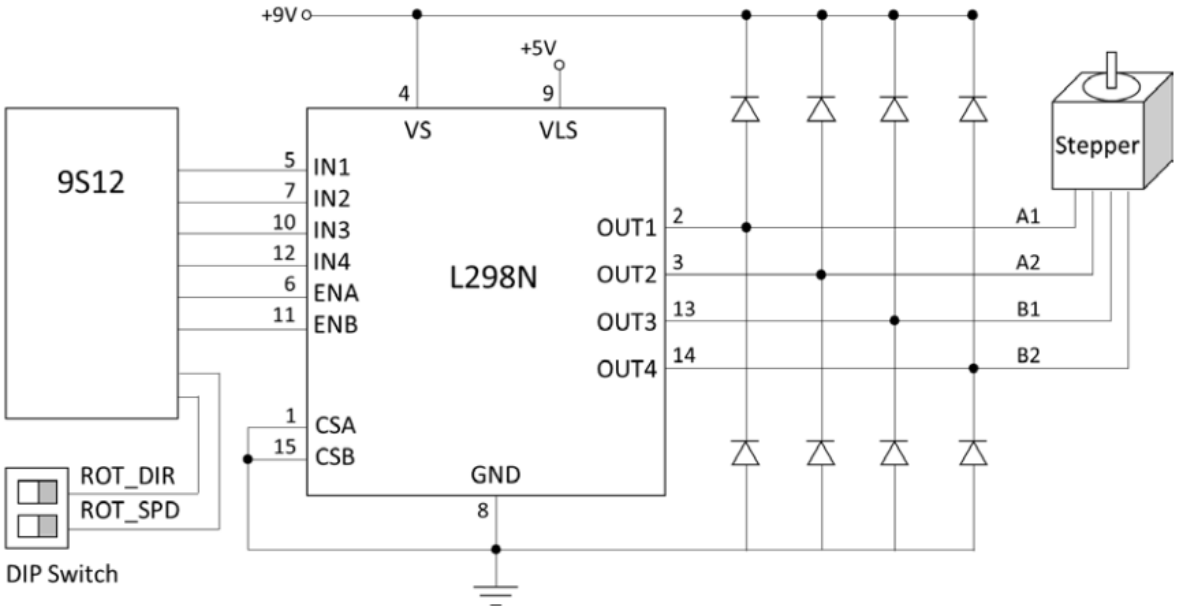


Figure 2: Overall System block diagram

Detailed Schematic

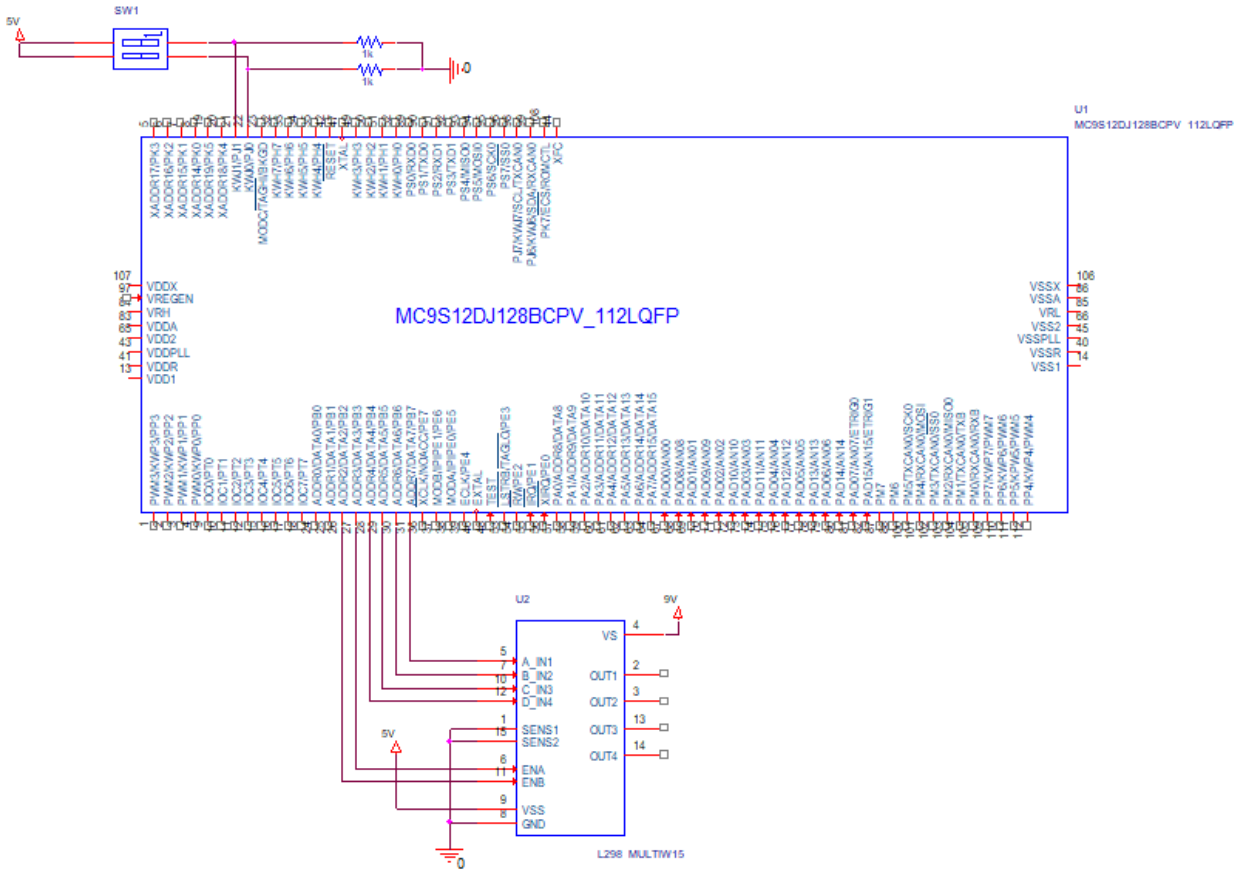


Figure 3: System Schematic

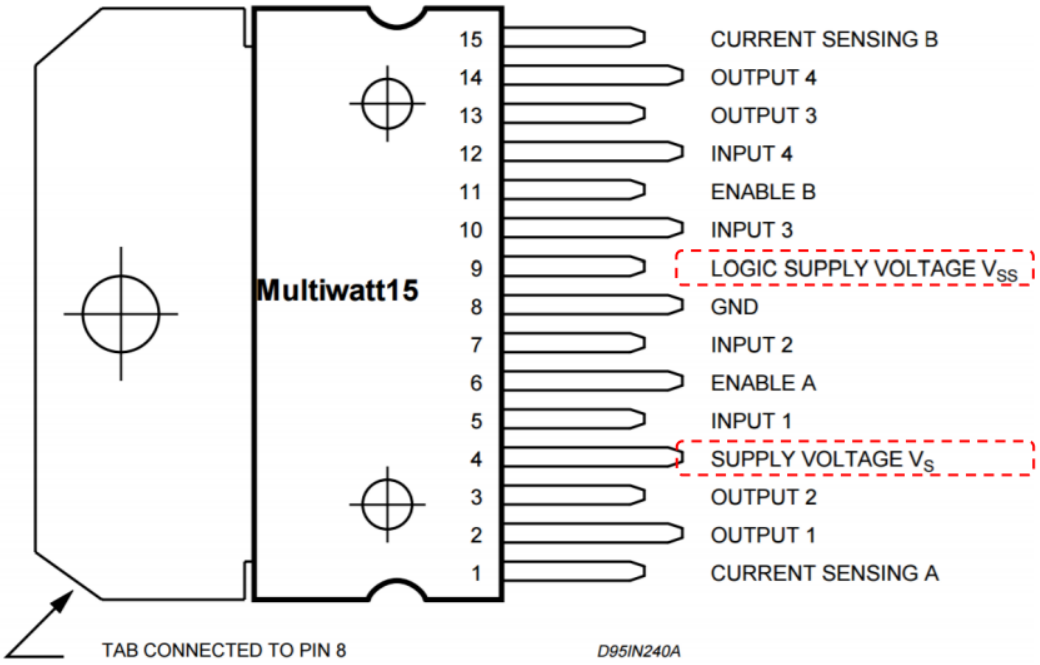


Figure 4: L298N Pinout

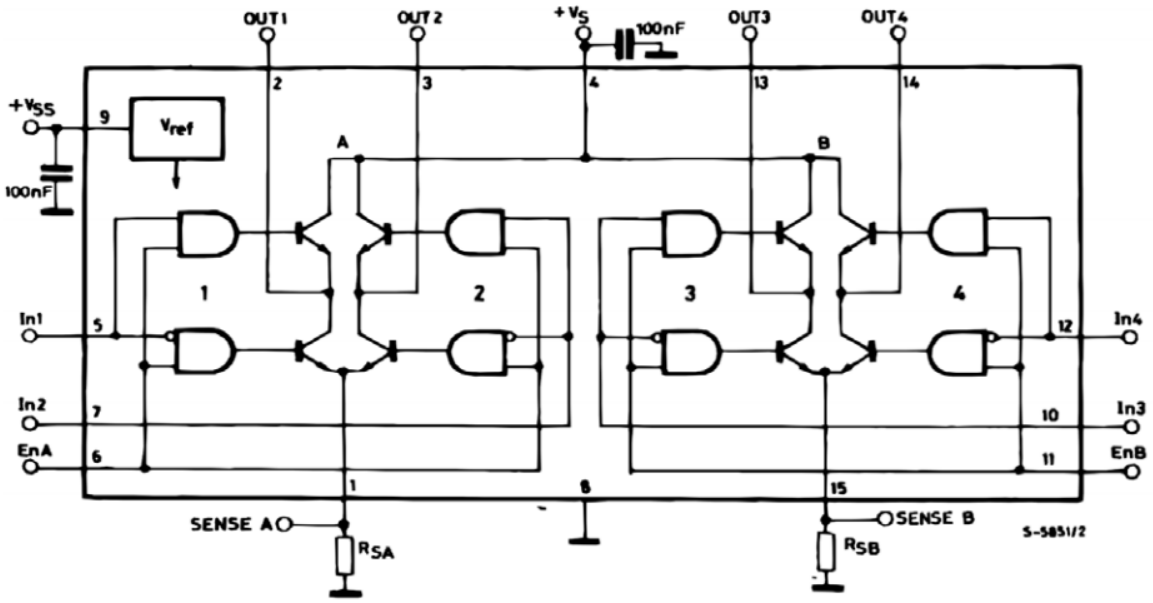


Figure 5: L298N Internal Block Diagram

High Level Description of Software

This program reads the states of two switches and determines the speed of rotation (0: 22.5 degrees per second, 1: 90 degrees per second.) and the direction of rotation (0: clockwise, 1: counter-clockwise). It then controls an L298N to drive a stepper motor.

Program Listing

```
#include <hidef.h>   /* common defines and macros */
#include "derivative.h"  /* derivative-specific definitions */

unsigned long i;
char rot_dir = 0; //PA0
char rot_spd = 0; //PA1
unsigned long speed_val = 30000;

void main(void) //OUT1=PB0 OUT2=PB1 OUT3=PB2 OUT4=PB3
{
    DDRB = 0xFF; //output for stepper
    DDRA = 0x00; //input for switch
    PORTB = 0x30;

    while (1)
    {
        //check switch
        rot_dir = PORTA & 0x01;
        rot_spd = PORTA & 0x02;

        //change speed
        if(rot_spd)
        {
            speed_val = 7500;//faster
        }
        else
        {
            speed_val = 30000;//slower
        }

        //change direction
        if(rot_dir)
        {
            //step CW
            PORTB = 0x36;
            for (i = 0; i < speed_val; i++);

            PORTB = 0x35;
            for (i = 0; i < speed_val; i++);

            PORTB = 0x39;
            for (i = 0; i < speed_val; i++);

            PORTB = 0x3A;
            for (i = 0; i < speed_val; i++);
        }
        else
        {
            //step CCW
```

```
50              PORTB = 0x3A;
51              for (i = 0; i < speed_val; i++);
52
53              PORTB = 0x39;
54              for (i = 0; i < speed_val; i++);
55
56              PORTB = 0x35;
57              for (i = 0; i < speed_val; i++);
58
59              PORTB = 0x36;
60              for (i = 0; i < speed_val; i++);
61          }
62      }
63  }
```

Technical Problems

Program wasn't working We had forgotten to change a parameter in the burner for the HS912.

Answers to Questions

Prelab Question

It can not change it instantly. This is because of the fact that each coil is an inductor and to conduct an instantaneous change in current would require an infinite voltage over the inductor coil. It can change direction only if the next coil has not been energized yet. Then changing direction is simply the result of activating another coil.

Lab Questions

We experienced the motor stalling only when we did not trigger the coils correctly. It could also happen if the program switched too quickly resulting in stalling because the coils would not be energized fast enough to move the motor.

Conclusion

In this lab we were able to successfully control the stepper motor. We did this lab with software loops, we could have also done it with the hardware timer.