Lab 12-13 | C Language ADC with Stepper Motor Integrations

ECET 20900

November 30th, 2023

Timothy McNamara, Kobe Liles

**Executive Summary**

This lab consists of two major parts and conjoins the two theories together. The first being the theory behind stepper motor operation; Creating an array in C that consists of specific hex values that correlate to the rotational position of the stepper motor. The second theory involves the analog to digital converter (ADC) and using a potentiometer to vary the incoming analog voltage signal that then becomes a digital output readable by our chip. The objective of this lab is to create a program that turns the stepper motor but uses the ADC converter to change the motor’s speed. The expected result of the lab is to achieve a working program. Once we download the program to the trainer board, by the turn of the potentiometer, we should see the speed of the stepper motor change.

The first step was to determine the range of the potentiometer. After that, we determined the maximum and minimum count we could have in our delay loop before the motor would stall. This delay loop determines how fast we step through the array and changing this value to a smaller one increases the motors speed. Subtracting both the maximum and minimum value gives us the motor speed range. Subsequently, we calculated speed step value by dividing the difference of the speed range by the maximum value the ADC can output. This was all the prework we had to do before we actually constructed the program.

We ended up also using another .c and .h file into the mix. But after we did our calculations, it was as simple as a copy and paste due to the fact that we had done both of these labs separately and were combining both of them but adding a formula and changing a function to have a value passed to it. The lab went as smoothly as you could ask for and we did indeed achieve a working program. I personally enjoy labs that are as creative as this. I have no recommendations. Overall, I found more reasons to enjoy the class and be happy with the experience I am gaining in this class.

**I started to do a formal lab report but realized the assignment did not dictate that a formal lab report was needed.**

**Design Documentation**

**Calculations:**

ATD Pot Range:

Highest possible range – Lowest possible range = ATD Pot Range

12910 – 010 = 12910

Calculating this value lets us know what the maximum value our potentiometer can achieve. When we use the potentiometer through the ATD Converter, we get a varying digital value that can be used in our formula at the end.

Motor Speed Range:

Highest delay number allowed – Lowest delay number allowed = Motor Speed Range

6553010 – 316910 = 6236110

Determining the possible range of our stepper motor is important to figuring out the limits of the stepper motor. Every single stepper motor is different and has different tolerances. They stall at different values. Figuring these out lets us get the step speed value to be as close as possible to our maximum and minimum delay allowed.

Step Speed Value:

Motor Speed Range / ATD Pot Range = Step Speed Value

6236110 / 12910 ≈ 48310

The step speed value gives us a number we use to increase the delay each time the digital value changes. 129 different times we add 483 together to get a delay of 62361. A different digital value determines how many times we add 483 together to get our delay value.

Scaled delay value formula:

Maximum motor speed range – (Converted digital value \* Step Speed Value) = Delay Value

(65530 – (Digital Value \* 483) = Delay Value

This is the actual formula we will be using in our main.c or in our function. I used it in the main.c file. This delay value and function is used to determine the speed of our stepper motor. The speed is also how fast we step through the hex array that corresponds to the rotation of the motor.

(65530 – (129 \* 483)) = 322310 | (65530 – ( 0\* 483)) = 6553010

Highest Speed Lowest Speed

A screenshot of a computer program

Description automatically generated



I have drawn circles to how I designed the formula to be used in the endless for loop. We also pass the calculated delay value to the delay function that changes the speed!

**Introduction**

This lab has an objective for us to combine two labs we have done and conglomerate both of them into one program. This is done by creating a formula that uses a returned digital conversion value to find out how big or small the delay in function will be. This delay controls the stepper motors turn rate. An added rule and new lesson in this lab were the creation and use of a “.c” and “.h” file. We created separate file for the function prototypes called “Protos.h” and we created a file filled with our functions called “MotorControl.c” This lab tests theory in both the analog to digital converter and the function of our stepper motor. The expected result is to use the potentiometer on our trainer board and change the speed of the stepper motor. Also, we should be able to change the rotation of our stepper motor with one of the eight dip switches. This lab will be important not only for our grade in the class but our ABET accreditation.

**Procedures**

**Materials/ Equipment:** Trainer Board, CodeWarrior Software, Laptop,