

CECS 347 Spring 2018 Project #2

by

Doan Peter, Takatsu Naoaki

12, April 2018

A DC motor car controlled by PWM. 2 IR distance sensors connected for distance detection. 1 potentiometer used to control speed of motor.

**Introduction**

In this project, we created a 12-volt battery pack powered DC motor car with two IR distance sensors for detecting distance and one potentiometer for controlling motor speed. The IR distance sensor will read in ADC values, which will then be converted to centimeter units by passing it through a linear regression function. The potentiometer’s ADC output values will become the PWM duty cycle.

**Operation**

By twisting the potentiometer counter-clockwise, we can speed up the DC motors approximately from range 0% to 99%. The speed percentage is displayed on the LCD.

The two IR distance sensors will automatically measure distances with a 100ms frequency. Since the slowest ADC0\_PC\_R sampling rate is 125,000 samples per second, which is too fast for us to read, we included a systick delay of 100ms inside the IR sensor function. The distance between the sensors and obstacles are displayed on the LCD with units of cm. When the distance exceeds 35cm, the LCD will display “OOR” (Out Of Range).

<https://www.youtube.com/watch?v=HwsUBwcTuUk>

**Theory**

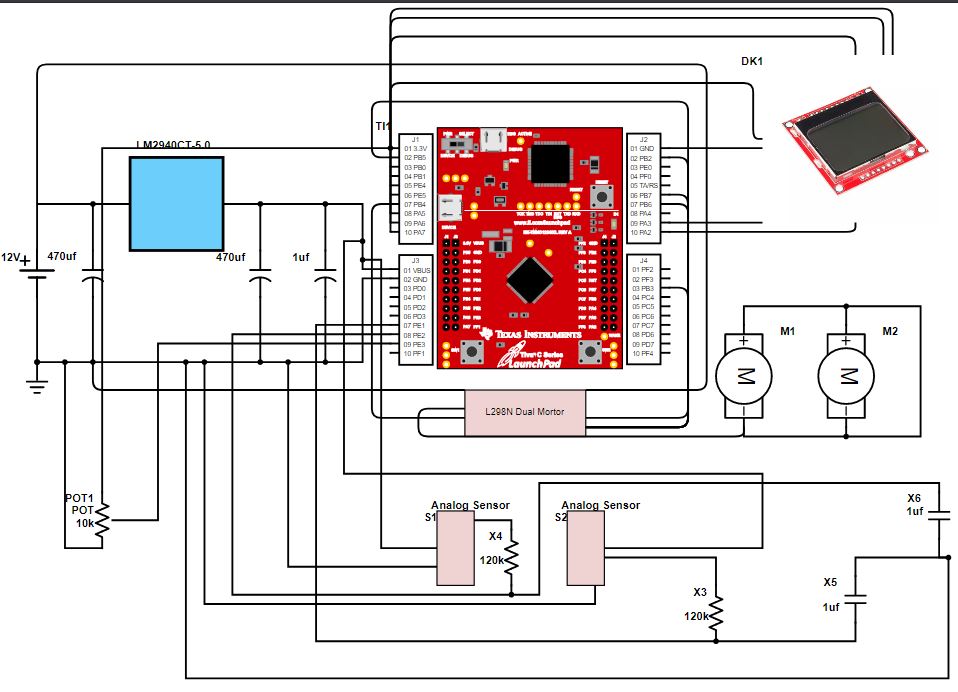
Motor: The motor’s rotation speed is controlled by PWM (Pulse Width Modulator) duty cycles. To obtain certain amount of speed, we set the duty cycle equal to the percentage of the speed we want multiply by the whole period. Since the duration of the duty cycle directly connects to the duration of how long the motor runs each period, we can set the % to be higher for faster speed and lower for slow speed.

Nokia 5110 LCD: The Tiva TM4C123 Launchpad acts as the master and the Nokia LCD serves as a slave. Data/information obtained from the launchpad is passed to the LCD to display its contents.

Potentiometer: A potentiometer can change its resistance by twisting its handle. Since V=IR, and current stays constant, changing the resistance will change the voltage reading. The varying voltages are converted to ADC values; then we can obtain the duty cycle of the PWM by dividing the measured ADC value by the max ADC value. Finally, the duty cycle will control the speed of the PWM motors.

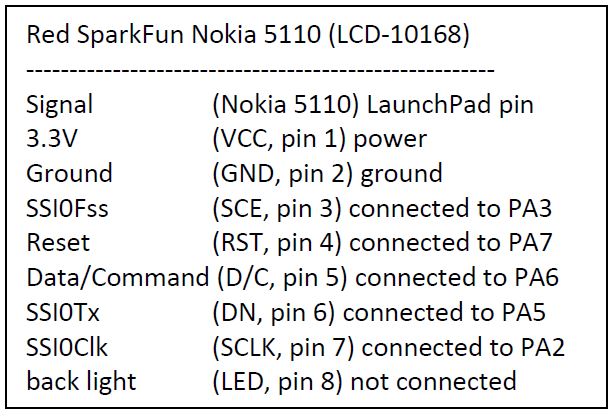
IR distance sensor: The IR LED shoots out infrared waves towards an object and the IR sensor receives the brightness bounced back from the reflection. When the sensor receives information, it will convert the brightness (analog information) into ADC values (digital information). Since ADC values are not suitable as distance measuring units, we must convert the ADC values into centimeter values. To do this, we must first measure and record both ADC value and distance (in cm) from 10cm to 80cm in 1cm intervals. Since IR sensor’s voltage output jumps up to a peak voltage and then drops gradually before the 10cm point, we will use 10cm as our starting point. As the distance between the sensor and the object become farther apart, the ADC value will drop. The ADC drop rate follows a linear regression path in respect to distance, so we input both information into a linear regression equation generator to obtain a fitting formula. By passing real time measured ADC values into this formula, we can obtain real time distance values. Since the distance readings become unstable after 30cm, we have included an out of range display when the readings exceed 35cm.

**Hardware Design**



**Software Design**

The software of this design uses Port A, B, and E. Port A is used to write to the LCD screen that will display the distance detected from the two sharp IR sensors and the PWM percentage based on how much the 10k ohms potentiometer is turned. Port B is used to give four signals and two PWM values to the motor driver. Port E receives the inputs from the two sensors and the single potentiometer.

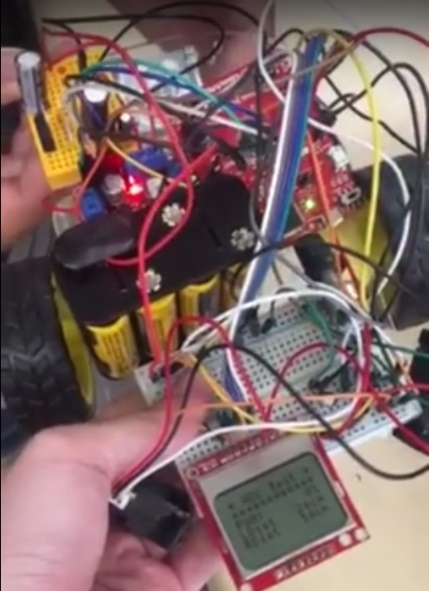


The figure above shows the pins for Port A and where it is connected to in the hardware.

The pins used for Port E is 1, 2 and 3. The three pins uses the alternate analog function that converts the analog signal to a digital signal (ADC). Port 1 and 2 are the sensors and the output are then put into an equation left\_dist = -3.3654785 + (40628.2059 / left\_ADC) and the left\_dist is the conversion to centimeters. Likewise for the right sensor, but replace all “left” with “right”. The potentiometer reads in a certain ADC value based on the resistance of the pot that ranges from 0 to 10k ohms. It then goes into an equation that divides the ADC value obtained by the maximum ADC value it can be to get a decimal that is the percent in which the speed should be. To achieve that speed, the period is multiplied by the decimal and the result will be passed and changed to become the new duty cycle. Since we need to measure 3 ADC values, we used SS2, which can handle up to 4 ADCs, as our sample sequencer: potentiometer as PE3 (Ain 0), left IR as PE1 (Ain 2), and right IR as PE2 (Ain 1).

The pins used for Port B is 2, 3, 4, 5, 6, and 7. Pins 2, 3, 4, 5 are signals that inform the motor driver which direction to drive. In this project, there is no need to go reverse, so IN1 and IN3 on the motor driver will always receive a signal of 1 while IN2 and IN4 will receive a signal of 0. Pins 6 and 7 are the PWM that outputs based on the duty cycle. It is connected to ENA and ENB to manipulate the speed the motor will go.

In the main super loop, there is only equations and terms to update values that goes into the equations meant for the sensors. The results are then constantly outputted into the LCD screen that shows the left distance, right distance and PWM percentage.



**Conclusion**

In this project, we had encountered few troubles. First, when we connected our potentiometer to the board, it did not read values even though we twisted the handle. By reconnecting the circuit and reprogramming the code, we somehow managed to fix this problem. Next, our Nokia LCD faded frequently so we assumed that it was either loose connection or dead batteries. Since replacing them to new batteries didn’t solve the problem, we found out that it was loose connections on the circuit after all. Lastly, since the motor, sensor, h-bridge, etc. draws power from the same battery source, the IR distance measurement could not exceed around 35cm. Luckily, the professor allowed us to set that as our maximum, we included the “out of range” display on the LCD in regard to that issue.