

CSCE 236

Embedded Systems

Robot Design Project 1 Report

“Theseus”



Michael Martin

Department of Computer Science and Engineering  
University of Nebraska – Lincoln

## Contents

Objectives or Purpose .....	3
Preliminary design: .....	3
Software flow chart or algorithms .....	5
Hardware schematic .....	6
Debugging .....	7
Testing Methodology or Results .....	7
Answers to Lab Questions.....	8
Observations and Conclusions .....	8
Documentation .....	8

## Objectives or Purpose

This is a project for embedded systems using an Arduino board and an Elegoo Basic Robot kit. The objective of this project is to familiarize myself with the physical assembly and software that is required to work with an embedded system. The ultimate purpose of the robot assembly and code is to get the robot to be able to follow a maze containing walls and objects that it needs to move around.

## Preliminary design:

I first built the robot following the instruction video that was included with the equipment. The idea on how to tackle the project was to break things down to each component starting with the ultrasonic sensor and servo.

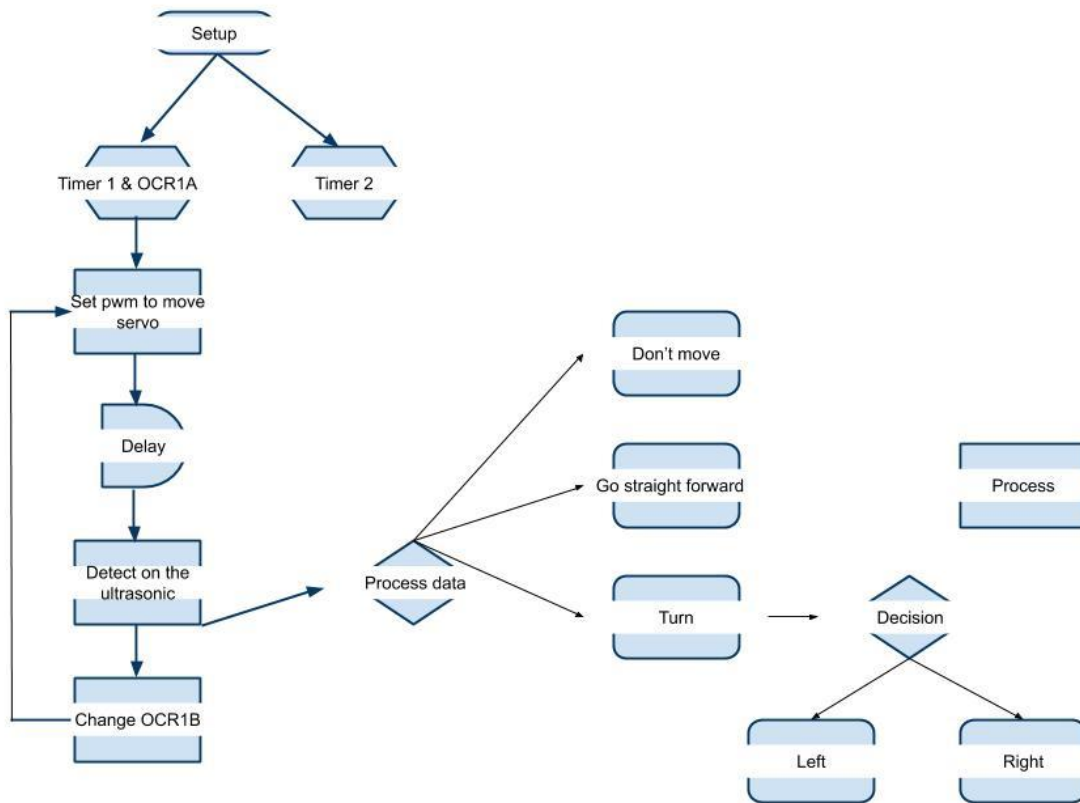
I chose to use timer 1 for both functionalities of the servo and ultrasonic. This frees up timer 2 to be used for the motors. The idea that I had was to use the input capture pin to detect when the signal goes from high to low. Resetting the timer when the signal goes high and then using the ICRA register that captures the time when the signal goes low to count the time the ultrasonic sound took to go out and come back. Using calculations on the speed of sound, we can figure out how far an object is from the sensor.

The sensor is controlled via a pwm signal that is sent using timer 1 features to output a waveform and a specific voltage. The variation of the voltage will determine where the servo points. After a figured time, the servo can stop receiving a pwm signal since it will be in position.

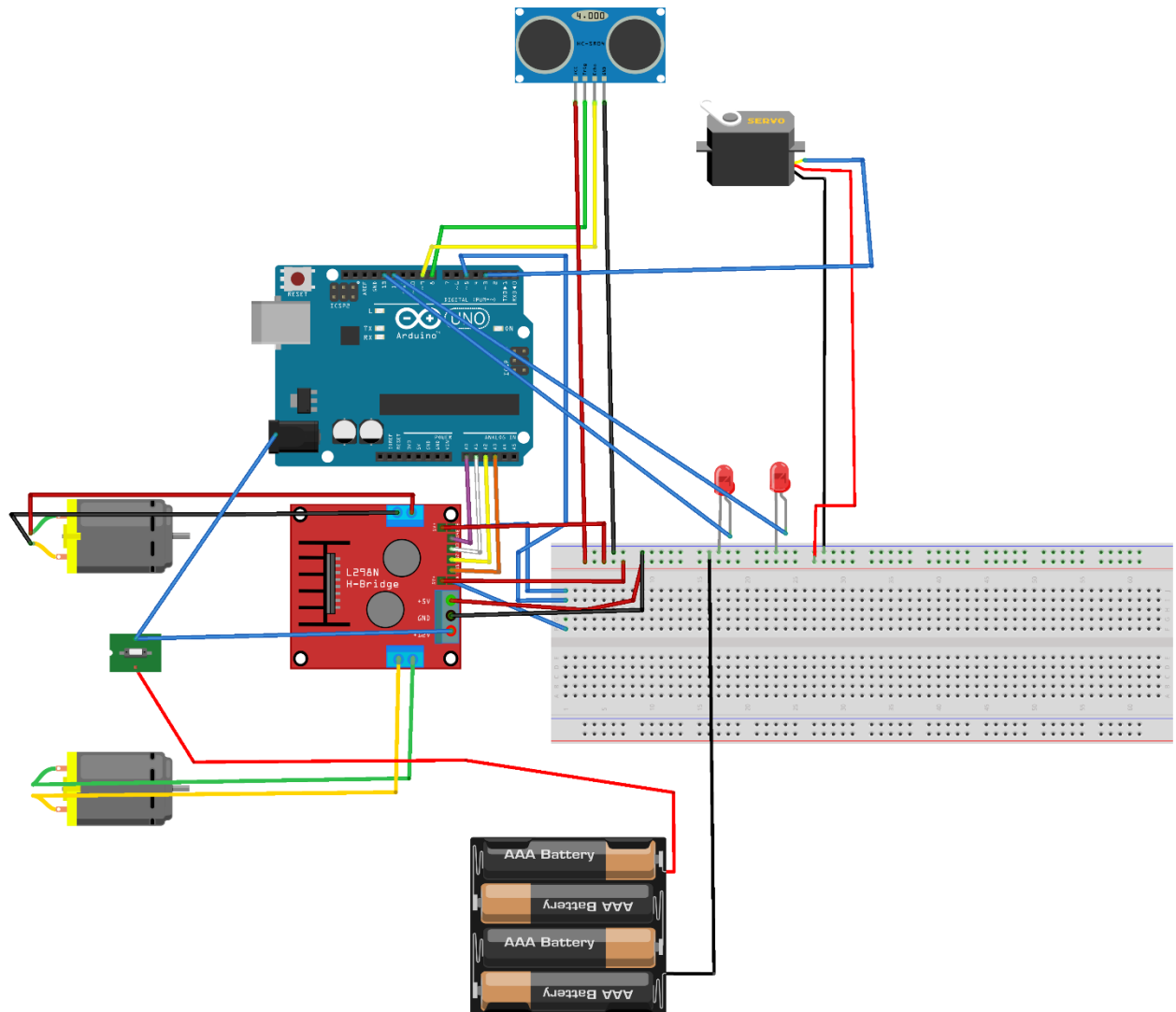
Also, once that position is reached a signal to output can be sent to the ultrasonic.

For the motors, I used timer 2 as timer 0 also messes with the delay function from the Arduino library. I plan on using the pwm signals from timer 2 for the enable pins on the H-bridge and just using high and low signals on the positive and negative terminals to choose direction of the motors. This allows me to set a direction easily and then set the speed on a separate signal.

## Software flow chart or algorithms



## Hardware schematic



fritzing

## Debugging

I had issues with each phase of the project, that I ultimately solved. First was figuring out how to send and receive valid data from the ultrasonic sensor. I kept getting complete garbage values back, so after some diving into the data sheets I found out that I needed to reset the timer when the signal coming from the echo pin went high and use the ICR1 to measure the time it took, since that is on the capture pin. After figuring that out I was able to adjust the prescaler to get an accurate number and then did a few measured tests to figure out how to scale that to inches.

When I got to trying to get my motors to run, I knew that everything was wired correctly as I had put the sample code that came with the robot kit onto it and both motors were able to run. But getting my code to have the same effect took some time. I found out that I was sending full signal to the enable pins because I was incorrectly assigning the OCCR2 registers. Once I figured that out I was able to adjust the speed that I ran the motors to something that was desirable for operation.

## Testing Methodology or Results

Testing the ultrasonic once it was working required a measuring tape and using the Serial port to print the returned time from the ICR1 that I was using. After 5 set test measurements I used excel to plot the graph, noting that the pattern was linear I figured a scale to change the time on the timer to inches measurement.

The next thing that required the most amount of testing was the wall following operation. I tested over 40 different variations of code before I was able to find a speed to run the motors at that kept the robot heading in a straight forward direction. Then using the ultrasonic to gauge how close I was to a wall and speed up either motor to adjust the heading to stay within a foot of the wall. Since this operation relied

upon the specific motors that I was using it was a lot of trial and error to find a set of values that would work.

### Answers to Lab Questions

Answers to lab questions have been submitted to canvas, will update with future labs.

### Observations and Conclusions

Observations that I have made thus far are that careful consideration must be made for each timer that is used and what pins those timers operate on. Messing with timer 0 can cause delay function to not work as desired. Taking the time to go through and also search the data sheet for the Atmega 328p is well spent. I have found many useful gems of information in there that has diagnosed issues that I was having.

### Documentation

Fritzing Software used for the schematic

Google Draw used for the algorithm flowchart