**CSCE 236**

**Embedded Systems**

**Robot Design Project 1 Report**

## Project KYLE

**Brandon Ramos**

**Department of Computer Science and Engineering**

**University of Nebraska – Lincoln**

**Due Date: 3/31/20**

**Objectives or Purpose:**

Project KYLE will be a couple weeks long project insisting on the learnings of the course Embedded Systems at the University of Nebraska. The goal for the project is to utilize the Arduino and its peripherals. Starting the project, KYLE has already had attached an Ultrasonic Sensor, Servo, and Directional LED’s to show the user what the robot is looking at if a wall is within a foot from the sensor. First, Kyle will have attached two motors and the H-Bridge with L298N.

**Preliminary design:**

For the past labs and homework, knowledge of the Arduino’s 8bit and 16bit timer/counters has helped with talking to the Ultrasonic Sensor and more importantly, the Servo. In detail explanation on functions will be listed.

**Algorithms:**

Listed below is the algorithm used for the motors. We set all 4 values for IN1to 4 when the drive functions are chosen.

void driveForward(int numOfQuarterseconds, int compareRegA, int compareRegB){

  //OCR0A-OCR0B=19 to go straight

  OCR0A = compareRegA;

  OCR0B = compareRegB;

  PORTD\_Reg |= (1<<In1);

  PORTB\_Reg |= (1<<In4);

  PORTD\_Reg &= ~(1<<In3);

  PORTD\_Reg &= ~(1<<In2);

  myDelay(numOfQuarterseconds);

  //turn off all motors and stop giving power

  OCR0A = 0;

  OCR0B = 0;

  PORTD\_Reg &= ~(1<<In1);

  PORTB\_Reg &= ~(1<<In4);

  PORTD\_Reg &= ~(1<<In3);

  PORTD\_Reg &= ~(1<<In2);

  myDelay(numOfQuarterseconds);

}

Image 1: Basis of driver functions

**Hardware schematic:**

Figure 1 shows what the project schematic looks like wired up. This can be used to tell what is where for debugging.

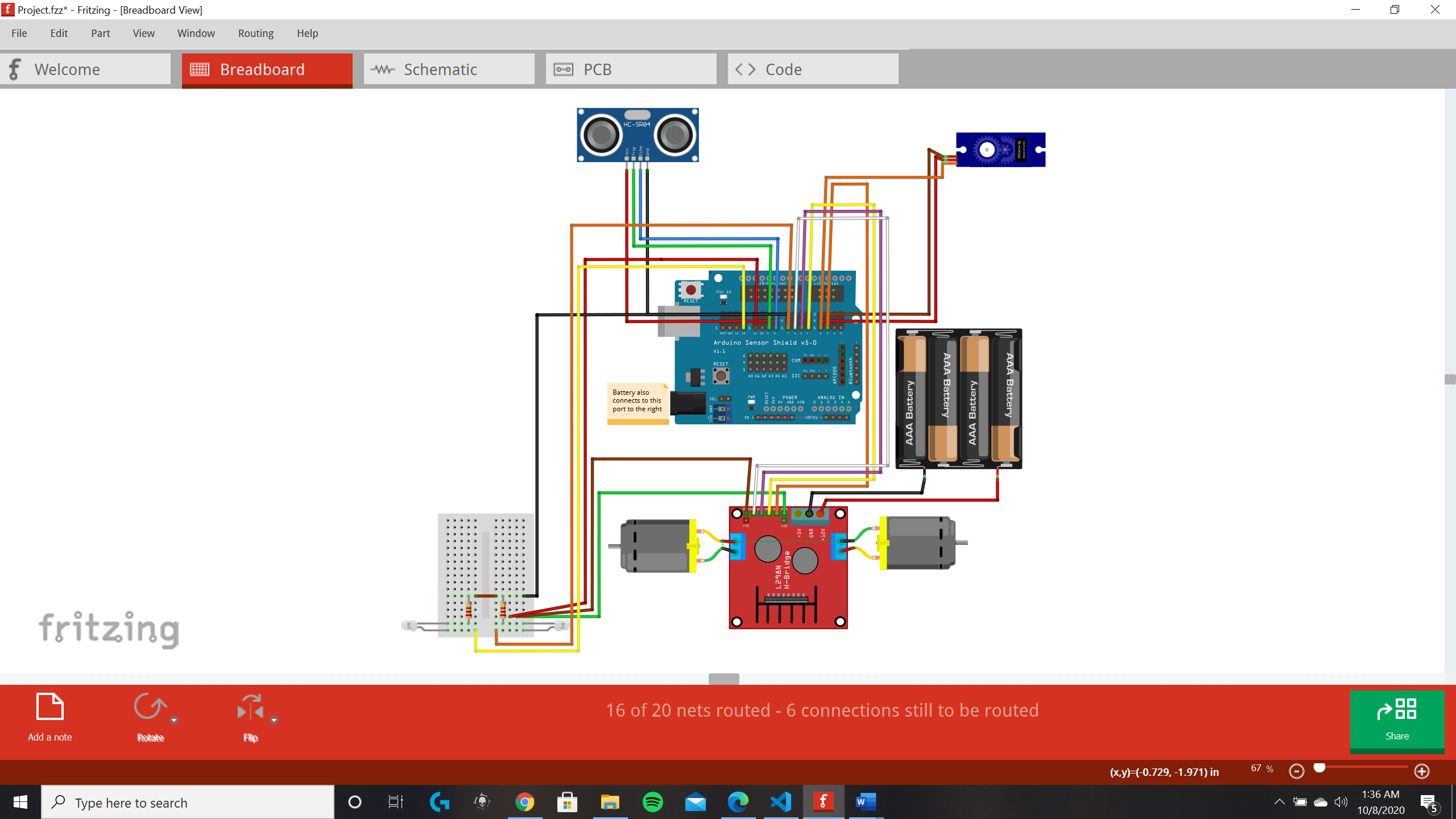


Figure 1: Schematic for KYLE

**Debugging:**

What went wrong were the motors in this project. To debug this and to get them spinning.

TCCR0A = (1<<COM0A1) | (1<<COM0B1) | (1<<WGM00) | (1<<WGM01);//on until OCR0B

  TCCR0B = (1<<CS00);

Shown above is the settings for Timer 0 which will modify the delay functions. But instead we create our own delay functions. Otherwise this is for Fast PWM with no prescaler.

What we did to debut this was to get TA help and change the setting from phase correct to fast PWM.

**Answers to Lab Questions:**

Checkpoint 1:

Embedded pictures of Figure 1 and Figure 2 of KYLE

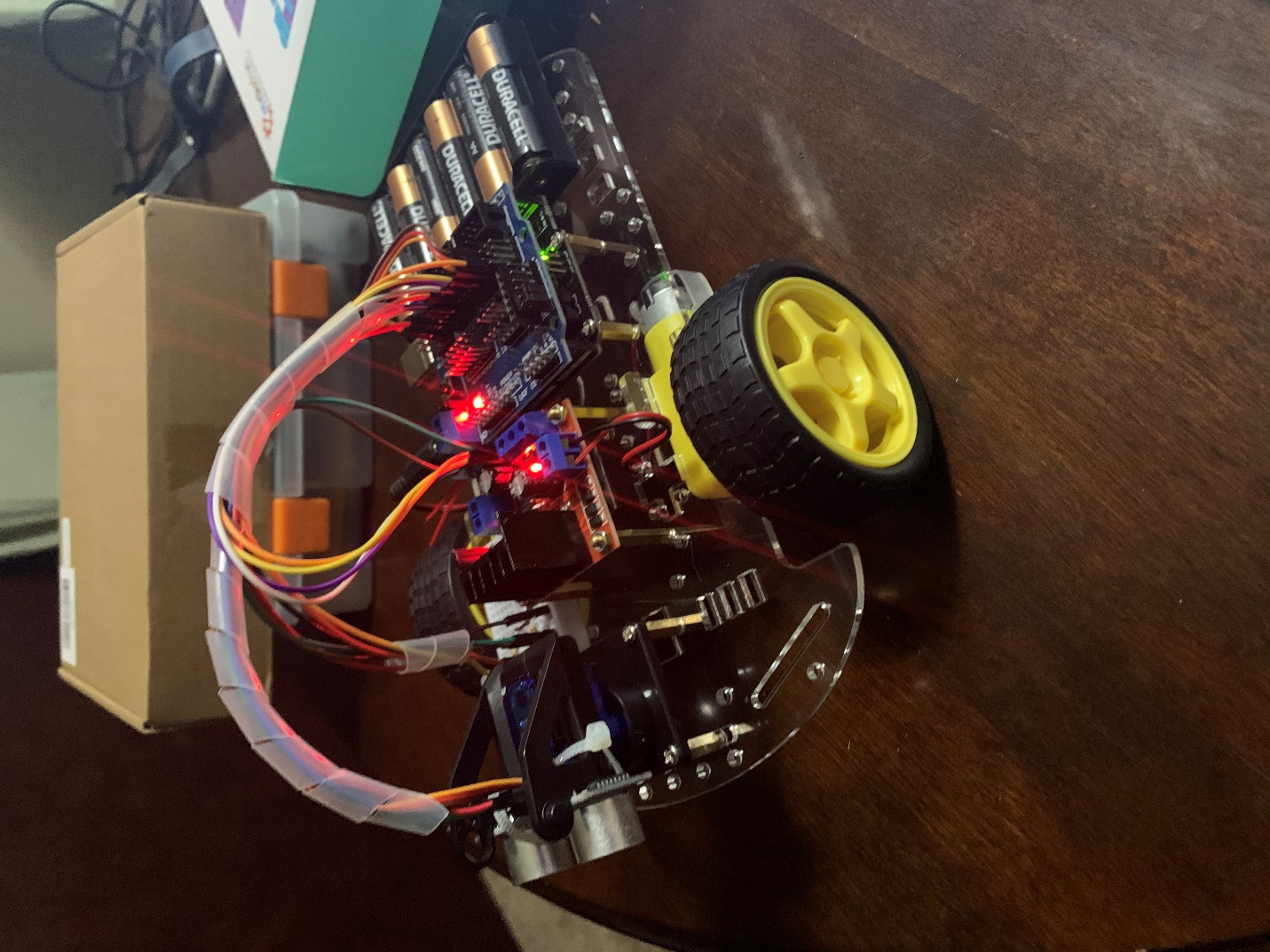


Figure 2: Checkpoint 1 picture of KYLE

Question: Discuss how will you be wiring up your motors? Do you plan on using a two or three wire interface to your motors?

Answer: I will be wiring my motors so that it takes a total of 5 wires including PWM from timer1.

Question: What will you be doing with the motor enable pins?

Answer: I will be disabling the motor disable pins to 5 volts and instead I will be using timer counter 1 to control the speed of the motor.

Question: What pins/timer will you be using to create your PWM signals and how will you change directions on the motors?

Answer: I will use timer 1 and pin 11 for PWM, for input 1,2,3 and 4 for the motor, I will use digital pins 2,4,5 and 6.

2 will connect so that both motors go forward, 4 will be backwards, turning left and right will be 5 and 6 respectively. these pins will be high and the others will be low.

Ex.) 2 will connect to the motors in1 and in4.

**Observations and Conclusions:**

Some of the observations that can be seen while working on this project are as follows. The motors running together to go straight are almost never accurate, so it is hard to get a good test when doing the wall following test on the robot. Most of the time this test was being done on carpet which could be the reason this was affected. Otherwise if tile was used, then it could have changed the accuracy of the motors. So instead, passing in parameters for the power to each motor was a best fit for the motors function.

**Documentation:**

Help received for the Project was mainly since the motors were hard to get turning. Lot of the code written was done and correct for the motor functions to work. However, setting up timer 0 for the motors had its difficulties. At first, we tried to use phase correct instead of fast PWM and the settings for that in the registers were needed to be changed.

Another