CSCI 490: Autonomous Mobile Robotics

Assignment Name: Final Project (Maze Solver Robot)

Assignment Number: Final Project

Group Members: Ricky Hempel & Nisha Patel

Project Description:

The project that we are doing is a maze solving robot. The robot must know how to handle different situations like a dead end, when to turn, and when the labyrinth is solved.

What Worked (Source and/or Build):

The hardware that worked was the ultrasonic sensor and the touch sensors. The ultrasonic sensor was mounted on the side, so that it could use the right wall as a guide to find a path throughout the maze. The touch sensor was installed on the front so that the robot will know when it runs into a dead end. In front of the touch sensor is a bumper this bumper is loose enough so that it will be readily activated to detect the dead ends.

The software that worked was similar to the method we used for the line follower program, but instead of following a line we followed a wall. We used a distance of 5cm. When the robot got too far from the wall, it turns to get closer, and when it gets too close to the wall it turns to get away from it.

Next, the software involving the touch sensor did not include many steps. When the sensor is activated the robot will reverse, then the robot will make a turn to the right.

Lastly, how the software will handle when to make a turn and how to define the end of a maze. While the robot is going through the maze if it does not detect a wall it will move forward then do a turn to the right. Similarly, to find the end of a maze if the robot does not see any wall it will stop and turn off and therefore it will let the user know that the maze is over.

What Didn't Work (Source and/or Build):

The hardware that did not work was with the ultrasonic sensor. We had to find different distances from the wall, so that it could get an accurate measurement. Next, was the bumper for the touch sensor. We had to try about three different designs before we got the one we are using. The difficulty was getting the bumper to hook onto the robot and the sensor.

The software that did not work was with reading the distance from the wall. We tried many different distances from the wall before deciding on 5cm. We also tried different speeds the robot should go. We got it was too fast with 50 and too slow with ten, so we agreed on 25. Lastly, the ultrasonic sensor did not work as well on objects with rounded corners, so for the maze, we used boxes with sharper edges.

When the robot hit a dead end, at first we had it reverse too far back ,so that it would be too far and think that there is no wall, so it would make a right when it should have been going forward. Likewise, we had to set a distance to get it so that the robot would know that there is not a wall and make a right turn. Similarly, we had to adjust different distances, so that the robot would know that there is not a wall anywhere, and therefore that it made out of the maze.

What I feel that I learned from this assignment:

What we feel that we learn from this assignment. We learned how to achieve an independent task and research in the field of robotics. This is an important thing to learn because going into the real world we now know more about how robots work. We also learned more about how the ultrasonic sensors work. For example, it works best on solid objects and is does not work as well on pages on a book. Lastly, we learn how to make a test area for the robot meaning to use whatever materials that were found in the class to create a maze for the maze solver robot.

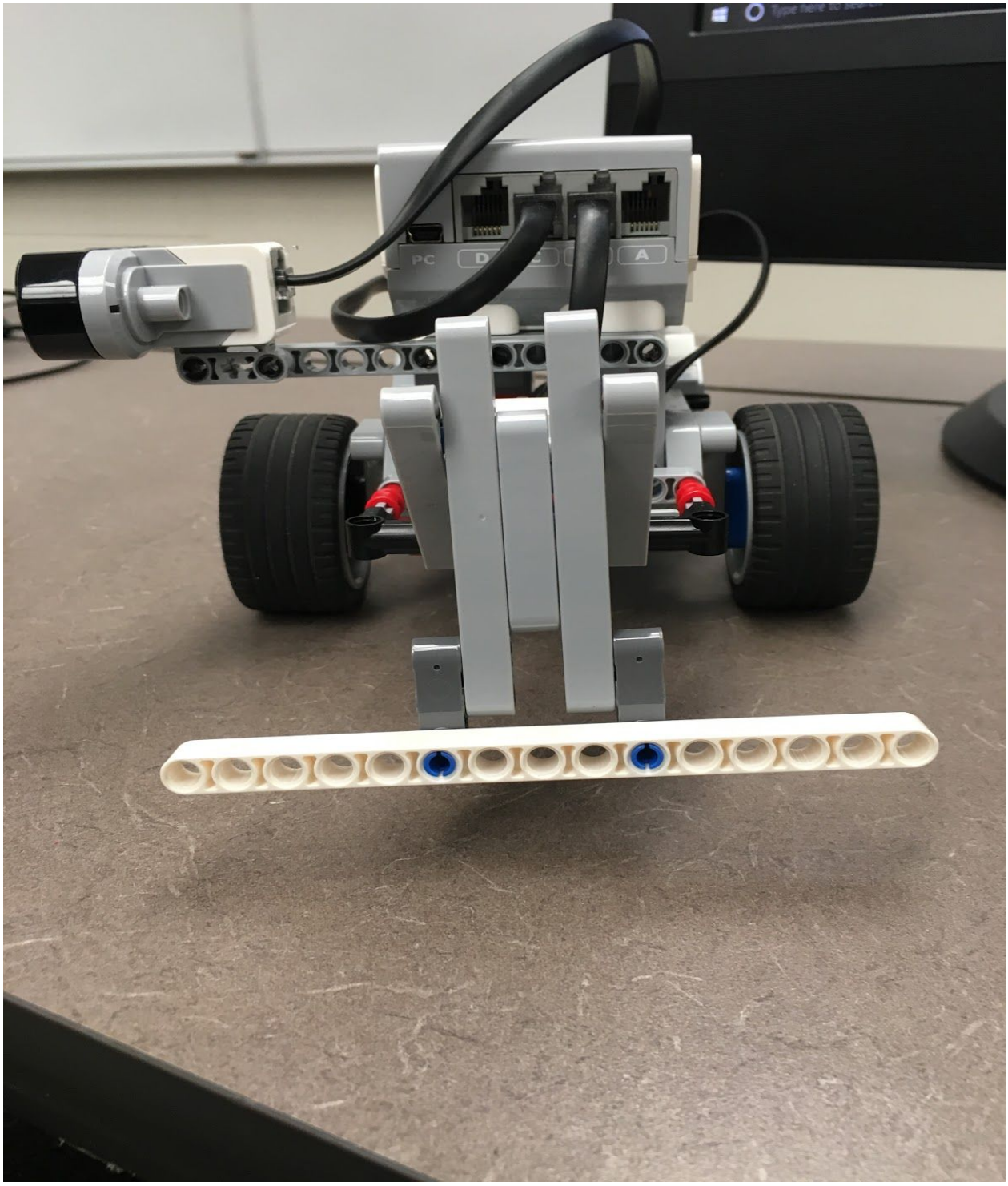

Figure 1: Build with touch sensor and ultrasonic sensor

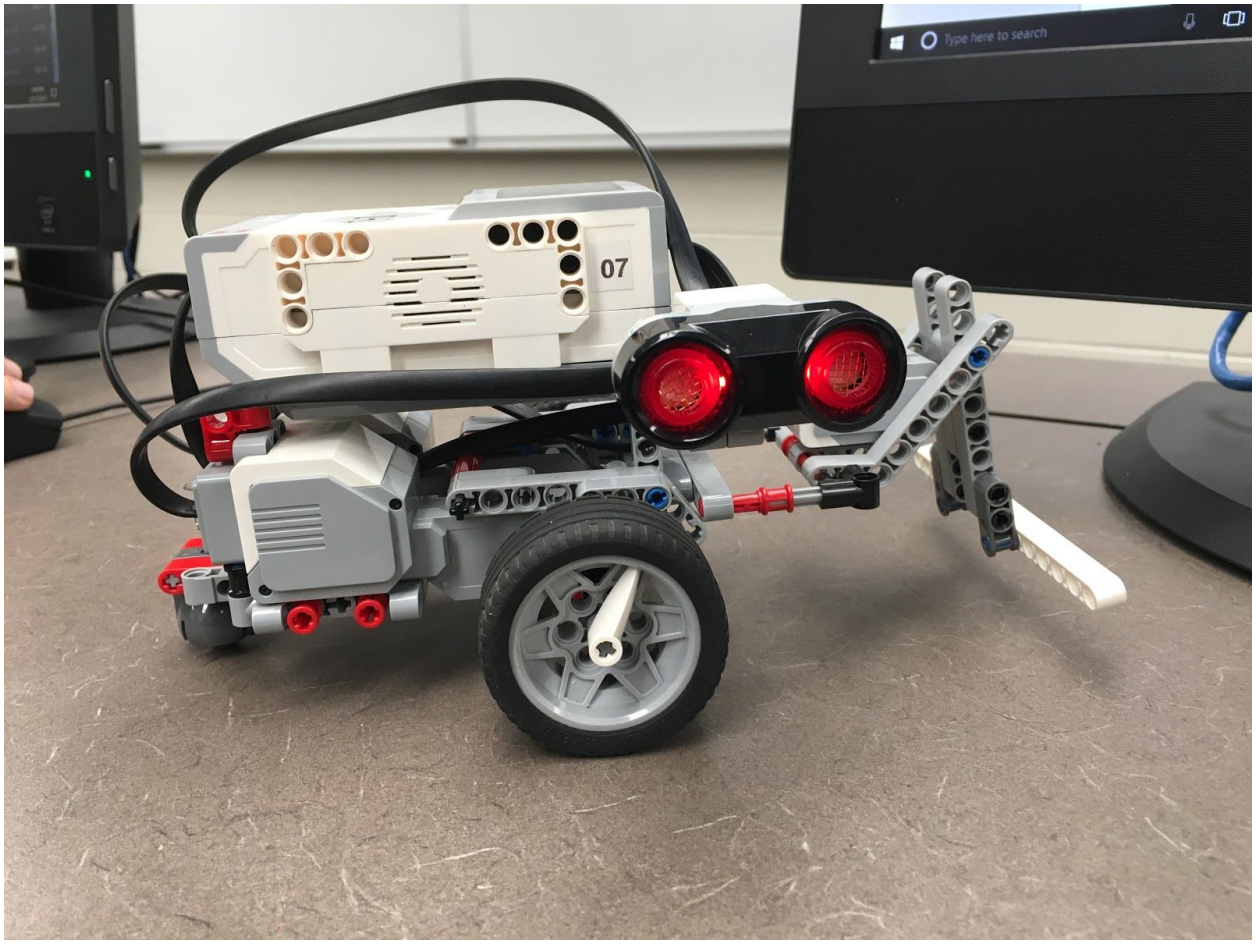

Figure 2: Build with ultrasonic sensor

```
/************************************
Name-RickyHempel & Nisha Patel
Date-12/12/2017
Purpose-Source for a maze solver robot
************************************/
//sensor info
#pragma config(StandardModel, "EV3_REMBOT")
#pragma config(Sensor, S1,    sonar1,         sensorEV3_Ultrasonic)
#pragma config(Sensor, S2,    sonar2,         sensorEV3_Touch)
/*
This program assumes you are using the standard LEGO Education robot
Sensor Ports:
S1 = Ultrasonic
S3 = Touch
*/
//motors
tMotor Left_Motor=motorB;
tMotor Right_Motor=motorC;
task main()
{
  //loop forever
  while (true){

        // if wall distance from wall is less then 5 turn left
         if (SensorValue(sonar1) < 5){
                setMotorSpeed(Left_Motor,0);
                setMotorSpeed(Right_Motor,25);
           }

          // else turn right
         else{
                setMotorSpeed(Left_Motor,25);
                 setMotorSpeed(Right_Motor,0);
           }
```

```
            //if sensor is touched back up and do a 90 degree turn
            if(SensorValue(sonar2)==1){
                setMotorSpeed(Left_Motor,-10);
                setMotorSpeed(Right_Motor,-10);
                sleep(550);
                setMotorSpeed(Left_Motor,-10);
                setMotorSpeed(Right_Motor,10);
                sleep(1750);
            }

              //if there is not a wall back and right turn
             if(SensorValue(sonar1) > 29){
                setMotorSpeed(Left_Motor,10);
                setMotorSpeed(Right_Motor,10);
                sleep(1000);
                setMotorSpeed(Left_Motor,10);
                setMotorSpeed(Right_Motor,-10);
                sleep(1000);
            }

            //if end of maze is found stop and stop all task
             if(SensorValue(sonar1)> 90) {
                setMotorSpeed(Left_Motor,0);
                setMotorSpeed(Right_Motor,0);
                sleep(1000);
                stopAllTasks();
            }
    }
}
```