Robot Project Report

Brendan Rice, Caleb Hubbell

4/14/16

Robot Project Report

ENGR 152

Chad Stillinger

**Project Summary**

The basic requirement of this project is to build a robot that navigates a city-style maze, enters an inner chamber where it uses a light beacon to locate a lego guy, and travel back out of the maze with the lego guy in tow. More detailed specifications can be found in the Robot Project Guidelines document which is in the ENGR 152 Google Drive folder.

While brainstorming the best algorithms to use to traverse the maze, we realized that there were no rules keeping us from inputting a series of moves for the robot to follow in order to complete the maze, with algorithms used only as backup to complete the maze if there is an error. In order to accurately drive forwards a given distance and turn a given angle, we decided to add limit switches to the wheels which will click on and off a total of forty eight times per revolution. This should allow us to keep our robot driving straight and going controllable distances. Once we have entered the center building we will have a predefined direction that the robot will turn. It will then follow the wall until it finds the light signaling the lego guy. Once this has been accomplished it will follow the same instruction in reverse order to exit the maze.

As a backup, if the robot is even right in front of a wall and the current move or step is telling it to drive forwards, the robot will analyze its current location (left side or right side of the maze) and then follow the object or wall towards the burning building.

We believe that this method of traversing the course will produce significantly faster and more consistent run times than traditional maze-solving algorithms. Reducing the number of inputs that the other robot or course layout could affect (ie. sonar sensors, light sensors, etc.) is another way we enhanced efficiency.

In summary, the robot is designed to go through a maze and find a lego guy in the center. To do such, we chose a slightly abnormal method. Our robot will be programmable in the couple minutes before the competition begins. The very first input will tell the robot which wall to follow once it gets into the center of the building. The next series of inputs will tell the robot right, left, or straight. Once programmed, we will hit the fourth button which tells the robot to begin. It will begin iterating through the instruction memory, doing one move at a time. It will do this until it fails (hits a wall), or senses heat (goes into the house). If it fails, it will begin following whichever wall it is closest to, and just hug a wall until it finds the way out. Once it senses heat (which it will either way), it will go back to that first input and chose weather to follow the right or the left wall. It will do this until it senses the light. Once it senses the light it will run everything backwards to get back out. It then will spin in a circle to celebrate.

**States**

* INPUT\_STATE
  + Gets input from the programmer on where to go
  + Displays data on LCD, reads input from analog pin
    - Static
* MAZE\_STATE
  + Follows the pre-assigned instructions to go through the maze
  + Displays direction it is moving on the LCD, reads input from all sensors
    - Transitional
* ERROR\_STATE
  + Defaults to this state if the robot hits a wall while following the maze instructions
  + Figures out where the robot is, and then tells it to either follow left, or follow right
  + Displays direction it is moving on the LCD, reads input from all sensors
    - Transitional
* HOUSE\_STATE
  + Once the robot senses heat, it goes into this state
  + It looks back at the first instruction and listens to it, following left or right
  + Displays success on the LCD, reads input from sonar and light sensors
    - Transitional
* LIGHT\_STATE
  + It follows until it finds light, once it finds light it knows it is on top of the lego
  + It then goes back and forth a couple times before calling the next state
  + Displays success on the LCD
    - Transitional
* RHOUSE\_STATE
  + Looks at which wall it followed to get to the lego guy
  + Flips a 180, and then follows the same wall out
  + Displays success on the LCD, reads input from sonar and heat sensors
    - Transitional
* RMAZE\_STATE
  + Once the robot senses heat again, it knows it is back in the maze.
  + It then does the same thing it did before, look through the list and move through the maze, just backwards this time.
  + Displays direction it is moving on the LCD, reads input from all sensors
    - Transitional
* VICTORY\_STATE
  + Once it is done following the maze instructions it knows it is out
  + It will then spin in a circle and celebrate.
  + Displays celebration on the LCD
    - Static

**Robot I/O Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Digital Input | Digital Output | Analog Input | Device | Description |
|  |  | A0 | Push Buttons | Reads the value sent by the push buttons. Four are hooked up to separate resistors, so they will send different power values |
|  |  | A1 | Temperature Sensor | Reads air temperature, used to tell us when we are in the burning building. |
|  |  | A2 | Light Sensor | Reads light level, used to tell us when we are on top of the lego guy. |
|  |  | A4, A5 | I2C Pins | Used to communicate to the motor shield |
| 0 |  |  | Built-in USB Port | TX Pin, used to transmit data to the host computer |
|  | 1 |  | Built-in USB Port | TX Pin, used to transmit data to the host computer |
|  | 2, 3, 4, 5, 6, 7 |  | LCD Screen | Used to interface with the LCD screen. This screen tells us useful information as the robot progresses through the maze |
| 8 |  |  | Ultrasonic Sensor #1 | This is the ultrasonic sensor on the right side of the robot. Pin sends and receives the data from the sensor. |
| 9 |  |  | Ultrasonic Sensor #2 | This is the ultrasonic sensor on the front of the robot. Pin sends and receives the data from the sensor. |
| 10 |  |  | Ultrasonic Sensor #3 | This is the ultrasonic sensor on the left side of the robot. Pin sends and receives the data from the sensor. |
|  | 11 |  | Speaker | This pin connects directly to the speaker, it sends a simple on off signal. |
| 12 |  |  | Limit Switch 1 | This pin reads in data from the right limit switch. It reads a high/low signal. |
| 13 |  |  | Limit Switch 2 | This pin reads in data from the left limit switch. It reads a high/low signal. |