Tanmay Sane
tsane@uncc.edu
Erick Chacon
echacon@uncc.edu

ECE 4161/5196 Lab 6 Report

Demonstration points:

Requirements:

Req. 1 The robot must traverse the obstacle course from a designated beginning point to the designated end point. ______
Req. 2 A LabVIEW program will be written that will help the robot to traverse the obstacle course. ______
Req. 3 The robot will operate autonomously. ______
Req. 4 The robot will retain 4-wheeled locomotion. ______
Req. 5 Students must follow the floor plan of the obstacle course given to them by the instructor. ______
Req. 6 There will be no adjustments made to the obstacle course for any reason. ______
Req. 7 Use one of the methods previously learned to traverse the obstacle course ______

Tanmay Sane
tsane@uncc.edu
Erick Chacon
echacon@uncc.edu

General Learning Objectives:

The general learning objectives of this lab was to implement the knowledge obtained throughout the semester and create a robot which used two sensors to navigate through a course.

General Steps Needed to Complete the Lab:

The general steps needed to complete this lab were to create a plan initially on how the program would be setup in order to complete the course. The two options to choose from were to use the A* program and lay out the course within the map, or to use sensors to detect the course and move the robot accordingly. We achieved the given set of requirements by using an ultrasonic and a pair of infrared sensors. The sensors were initially wired up and tested in order to secure the proper function of them. The program was then created which used the two sensors to detect the path of the course. Various test trials were run in order to accurately make the robot move as straight as possible. Once this was completed, the finished course program was uploaded onto the robot and was tested.

Procedure / Detailed Steps to Complete the Lab:

1. The first step that was performed was to analyze the course in which the robot had to maneuver through. An algorithm in which the Ultrasonic and Infrared sensors were to be used was chosen over the A* course planning.
2. In order to be certain that the sensors were functioning on the robots that were used for testing, an infrared sensor detection program was created. This program also allowed the users to determine what voltage was needed in order for the sensor to detect the ramp when it approached it. An image of this program is shown if Figure 1.

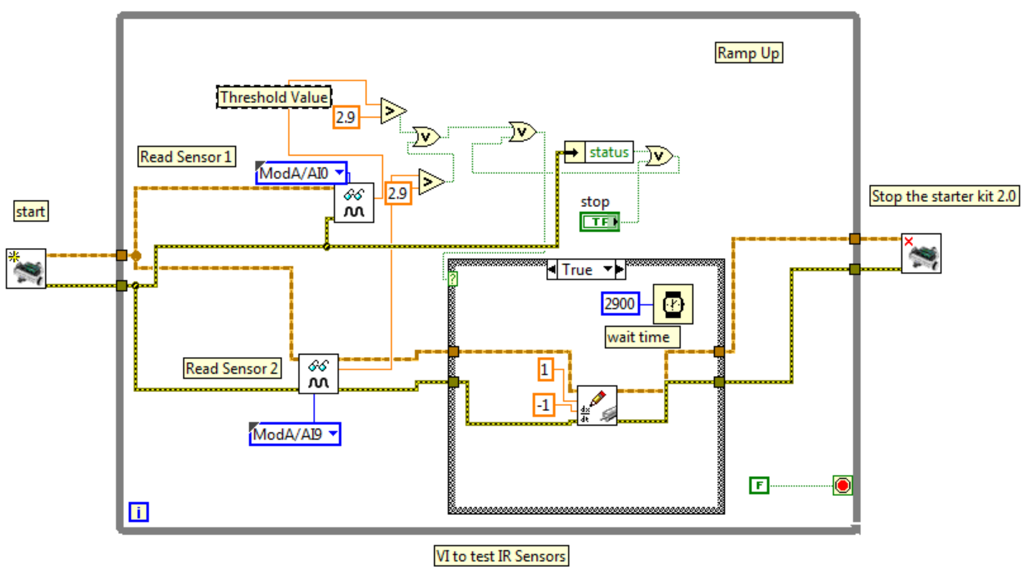

Figure 1: Infrared Sensor Detection Program

The method used to detect how much voltage was used was to put the robot close to the track. This is shown in Figure 2.

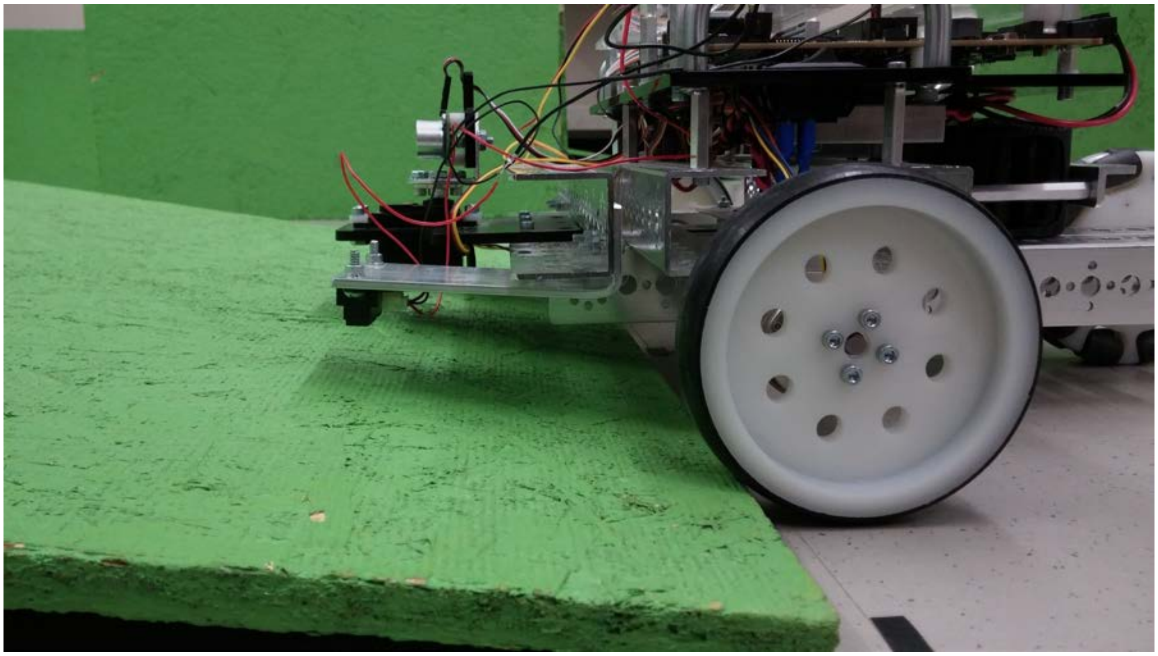

Figure 2: Robot Infrared Detection Test

3. After the sensors were tested and a voltage value was obtained, the motors were then calibrated so that the robot would follow a straight line. This was done to ensure the accuracy of the robot motion when traversing the course. A simple forward motion program was created in order to test the motors at different speeds. The values were changed slightly each run until a desired straight line was achieved.
4. Once both the sensors and motors were tested and adjusted, the program was created. This program uses flat sequence inside a while loop. Each frame in the flat sequence is a different operation for every movement of the robot. There are a total of 23 frames in the program. The first frame includes a time delay before the robot’s motors start. This delay was set to 20 seconds in order to have enough time to disconnect the robot and set it and the starting position. The second frame activates the motors and moves the robot forward for 2 seconds. The third frame lets the motors continue its forward motion until it detects a wall within 0.3 meters of distance. When the wall is detected the motors turn in order to make the robot turn 90 degrees right. The fifth frame shows the robot moving forward until a wall is detected within 0.3 meters. Figure 3 shows this set of frames.

tsane@uncc.edu
Erick Chacon
echacon@uncc.edu

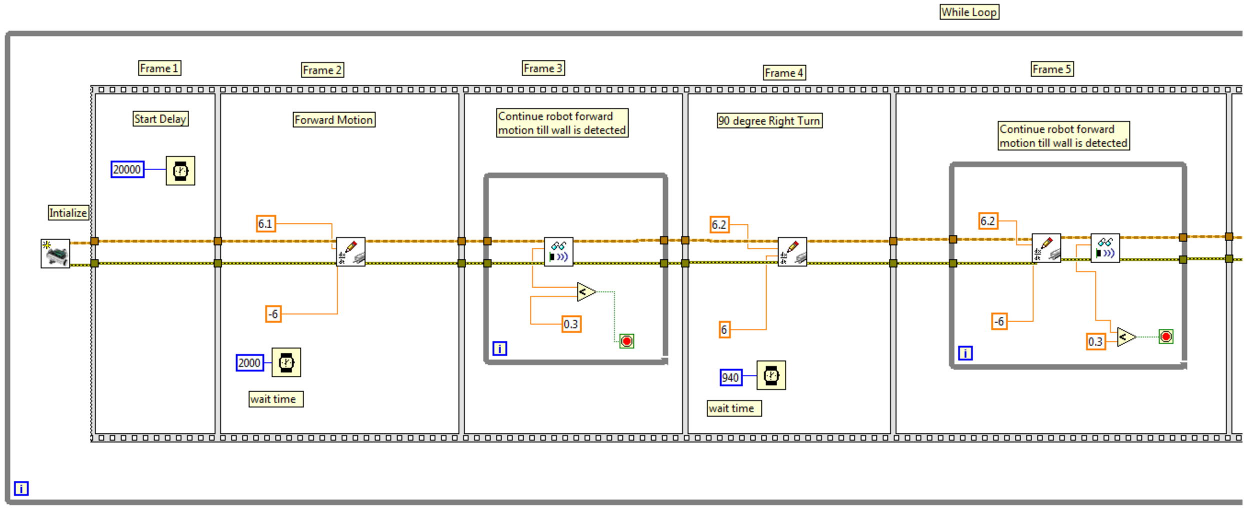

Figure 3: Complete Program for Course Completion Frames 1-5

The sixth frame turns the motors in a way that the robot turns 90 degrees left after the wall has been detected. This action is followed by the next frame which moves the robot forward for 3.3 seconds. This moves the robot just below the ramp as shown in Figure 2 previously. The eighth frame shows the IR sensor detection program implemented into the code. This detects the ramp and speeds up the robot in order to allow it to climb up the ramp. This part of the code is shown in Figure 4 below.

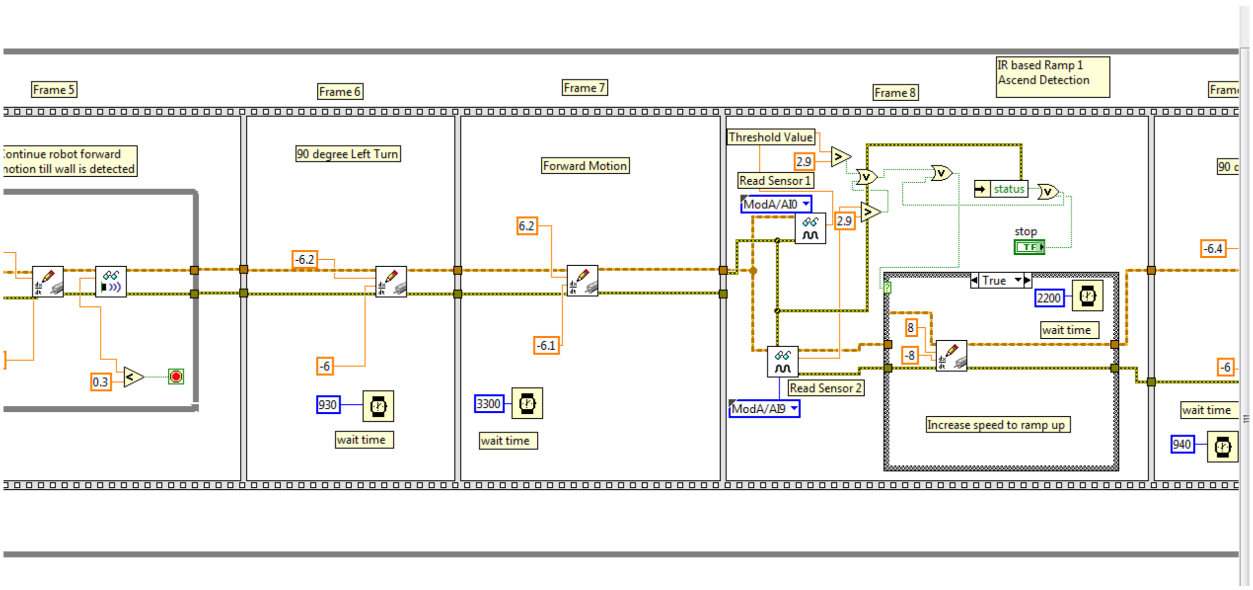

Figure 4: Complete Program for Course Completion Frames 6-8

After the robot has finished climbing the ramp after the elapsed time, frame 9 turns the robot 90 degrees left. Frame 10 shows the motors moving the robot forward until the next wall is detected. Once the wall is detected, the next frame activates and turns the robot 90 degrees right. The robot then moves forward until it detects another wall. These frames are shown in Figure 5 below.

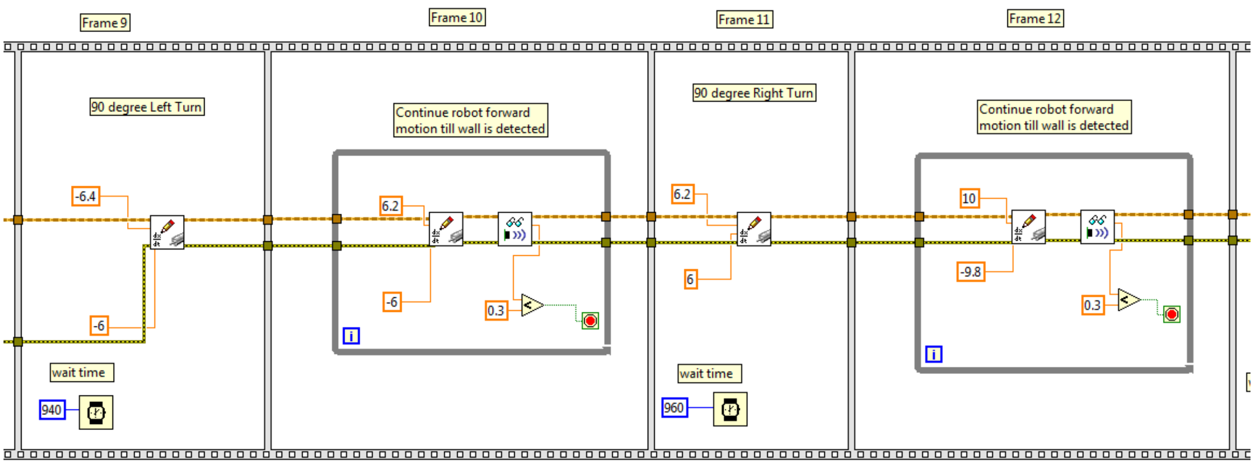

Figure 5: Complete Program for Course Completion Frames 9-12

Once the wall has been detected frame 13 shows the robot motors turning the entire robot 90 degrees left. After this action, the robot moves forward for a set period of time. In the case for this course, the time was set to 1.2 seconds. This allowed enough time for the robot to approach the next ramp and perform the same operation again with the infrared sensor detection program. This sped up the robot in order for it to be able to climb the ramp. Frame 16 shows decrease in speed of the robot due to a hinge placement on the ramp. This reduction in speed was implemented in order to prevent the robot from shifting directions once the wheels bumped into the hinge. These frames are shown in Figure 6.

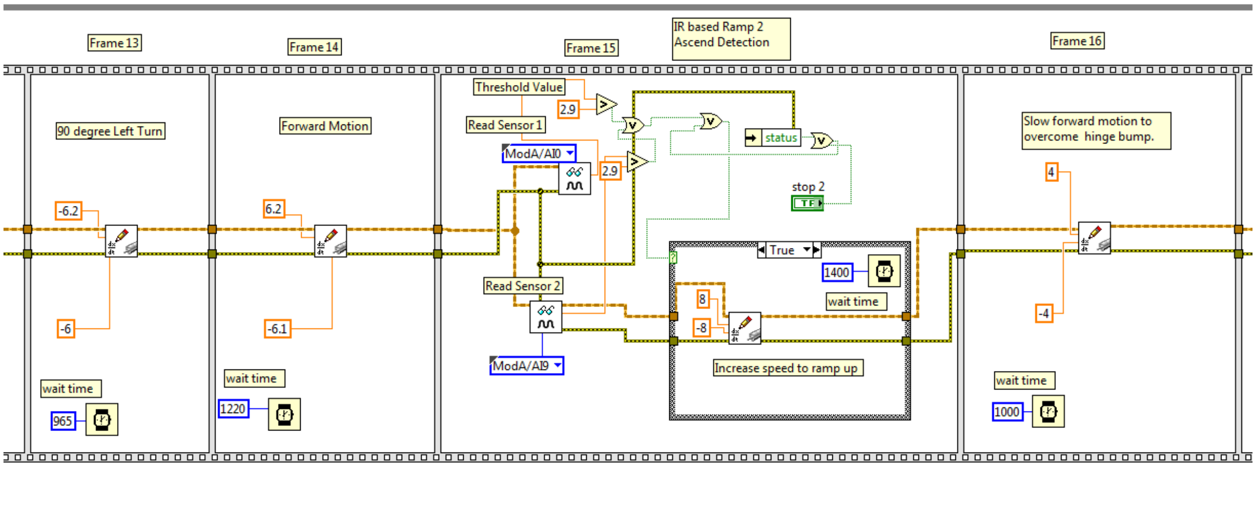

Figure 6: Complete Program for Course Completion Frames 13-16

After the robot has overcome the ramp and is at the top, the motors then turn the robot 90 degrees left. From then the robot continues a forward motion until a wall is detected. Once the wall is detected, the robot turns 90 degrees left once more and continues a forward motion until another wall is detected. These frames of the code are shown in Figure 7.

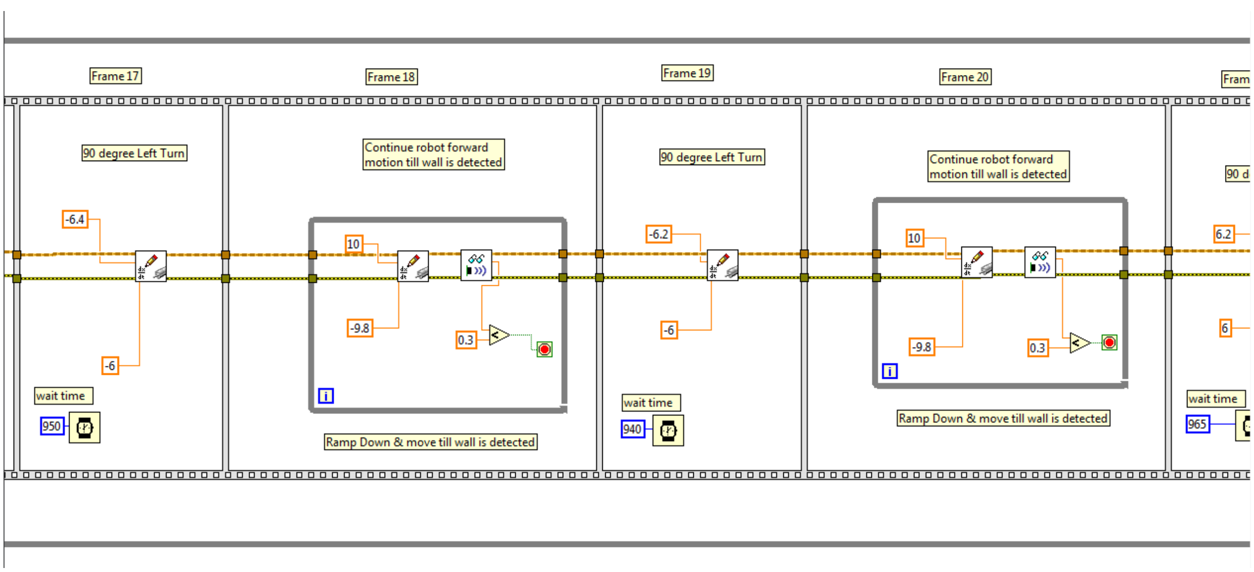

Figure 7: Complete Program for Course Completion Frames 17-20

The last set of frames continue from the forward motion of the robot in frame 20. Once the wall is detected, the robot then turns 90 degrees to the right. Afterwards the robot speeds up for 2.6 seconds and completes the obstacle by stopping the motors afterwards. This last set of frames are shown in Figure 8.

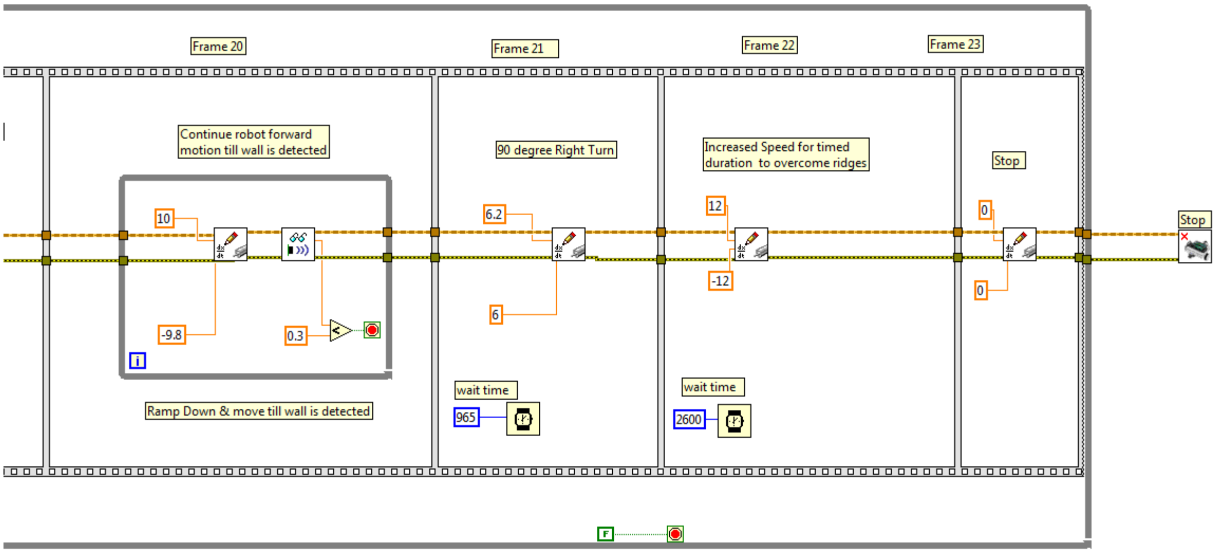

Figure 8: Complete Program for Course Completion Frames 21-23

5. After the program was completed, it was then compiled into the robot's controller. The program was then executed and the robot was put on the course. There were many instances during the runs in which motor speeds and timers had to be adjusted in order to traverse the course correctly. After these modifications were made, a successful run was made.

Observations while completing/testing the Lab:

It was observed during this lab that planning an algorithm for an autonomous robot can prove to be challenging. There are many factors that must be taken into account when creating the algorithm. Some of these challenges include how the robot's components will function during testing and whether the sensors are as reliable as predicted. Changing between different robots will alter these values, as well as the amount of battery life the robot that was being tested has. Another observation that was made was that the obstacle layout caused some problems to the robot. One of these problems was the hinges on the second ramp up. This hinge created many problems because when the robot passed over them, its angle changed randomly and the robot was not programed to determine this change in position and align itself in order to move down the ramp. This was discussed with the TA and the hinges were removed from this ramp. The final observation that was made was that the ridges could not be detected by the infrared sensors. The change in height between the floor and the first stair in the ridge was too little for the sensor to detect. In order to bypass this problem, the robot was made to simply move forward until the ultrasonic sensor detected a wall.

Lessons Learned:

In this lab we learned the following concepts & implementations.

a. In comparison to the A* program, the method used needed a more detailed path planning. Each frame in the program determined every move of the robot throughout the course. The A* program planned the path itself and made the robot move according to that path.
b. When using sensors or more than one motor, it is needed to calibrate each of the components to a desired value in order to achieve a desired performance from the robot.
c. For a successful completion of the track, a well-planned algorithm must be done. This ensures that appropriate measures are taken into account when the robot is ready to be tested.

References: Lab 3-5