Team Five: Erin Keith & Qandeel Sajid
Lab 3
Date: 10/1/12

Introduction

This lab is based on a tunnel escape program. The robot will rely on programmed obstacle avoidance capabilities to go through a tunnel, turn completely around, and exit the tunnel the way it came, without getting stuck. Each robot is timed, and each team's robot is expected to navigate the tunnel in a reasonable amount of time.

Description

A new robot kit was introduced mid-project; the Handy Bugs we had built in the previous labs were replaced by Lego NXT robots. The Handy Bug's design was modified using the NXT parts, as seen below.

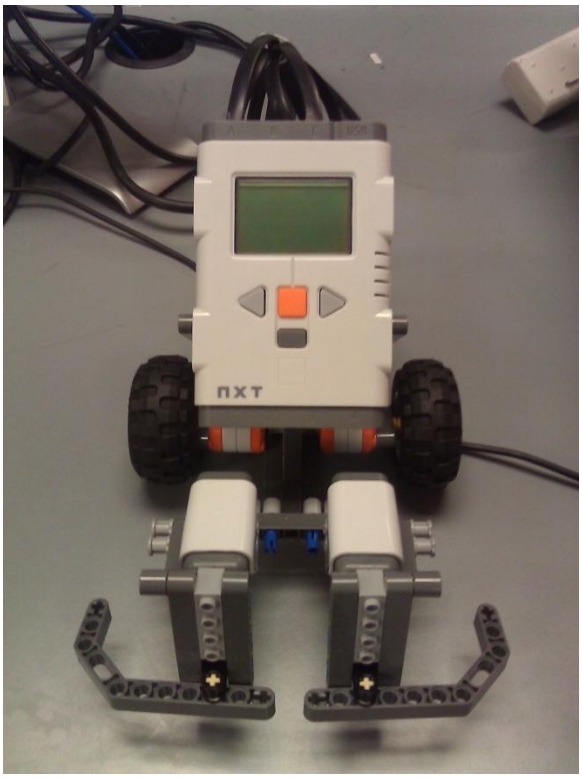

Figure 1: NXT Robot

There are still two touch sensor cables attached to the NXT robot, but they are now triggered by independent bumpers. Depending on which bumper is triggered, the robot will move away from the obstacle. For example, if the right bumper is depressed, the robot will back up a little and turn to the left. The turn angle will be less than 45 degrees, unless the robot has encountered 4 or more obstacles in 4 seconds or less, at which point it will turn a random angle greater than 45 degrees, and sound a beep.

Difficulties Encountered

This problem was pretty straight forward. Redesigning the robot was challenging, as was quickly learning the new language (Not eXactly C). The biggest challenge was keeping track of 4 bumps within 4 seconds. We figured out how to keep track of the time ticks, and used a round robin

system to test if the fourth bump occurred within 4 seconds. Finally, we figured out that each time tick was just a fraction of a second and our “stuck” routine worked.

Discussion of Unsolved Problems
There were no unsolved problems in this lab.

Results
The results have not been posted, but our robot did not perform as well as we expected. It did a fantastic job running through the tunnel towards the end, and turned around perfectly. Unfortunately when it ran into a little ledge, the turning angle was so small, that it could not turn enough to avoid it. Our robot continued to get stuck on that ledge and turn completely around, which I’m sure lost us points. It did eventually return to its starting place, and so did successfully navigate the tunnel.

Conclusion
Our robot solved the problem, but not as well as expected. The turning angle should have been slightly larger, or the timing for the bumps to determine whether the robot was “stuck” should have been reduced from 4 to 2 seconds.

Appendix
See website for source code.