PROJECT REQUIREMENTS

How to submit: Place a PDF document with the link/title "Project Requirements" in your group's resources folder and make a link to this on your group's Wiki page. Your page must be readable on a web browser. What to submit: Two-page paper per group, formatted as specified below

You are to specify what requirements that you plan for your project design to meet. Please keep in mind that this document does not describe how those requirements will be met or specify the design that you will use to meet the requirements. You may, in some cases, find diagrams to be useful in describing the requirements. This document will be used to judge both the difficulty of the project as well as your success in meeting the requirements. You should not devote significant portions of the document to repeating the specification that was given by the instructor for the project.

Created by:

Danny Duangphachanh
Leah Krynitsky
Brian Hilnbrand
Igor Janjic

September 14, 2014
[ECE 4534] Embedded Systems Design
*Virginia Polytechnic Institute and
State University*

Purpose

The purpose of the project is to design a rover that can traverse a room autonomously and be able to locate and drive over ramps.

Functional Requirements

The ARM board must perform all complex tasks. The rover will use an accelerometer to determine whether the rover is parsing flat ground, ascending a ramp, or descending a ramp. This accelerometer will also determine when the rover has reached the platform at the top of the ramp, and what the height of the platform is in comparrison to the ground. The rover will indicate whether it is driving forward unimpeded, has detected an obstacle, is maneuvering up or down a ramp, or has completed the course, and will report this status.

Technical Requirements

All software must be written in `C/C++` and must use the example code framework. Additionaly, the only hardware used must be an ARM board and PIC 18s. The ARM board may not power any other components and may only communicate to other components through I2C, USB, or Ethernet and to other processors through I2C or Ethernet. Additionaly, the rover must be remote from the ARM board.

Environmental Requirements

The rover should be able to be given a map of a large room and be able to use onboard sensors to find its location on the map and the location of ramps. These onboard sensors will continually send messages to the motor controller queue and the motor PIC will decide how the rover will operate. Depending on the surrounding area, the rover will turn left or right, move forward or backward at a normal speed, or climb an incline at an increased speed. The sensors will also be used to locate any obstacles throughout the course and to avoid collisions and navigate turns safely. Any obstacles detected in the path of the rover will be reported to the motor PIC and the rover will be re-routed accordingly.

When the rover detects a ramp or any incline via the accelerometer, the motor will output more power. The motor will lower the power that is being sent to the wheels when a marker is detected on the top of the ramp or the accelerometer signals that the rover is on level ground, whichever scenario occurs first. Once the rover finds this indicator, the motor will lower the power to normal operating speeds. If there is no indicator, whether a missing marker or the accelerometer is unleveled, the rover will continue at an increased speed.

Usability Requirements

The rover will be completely autonomous from the user in terms of sensing itâĂŹs surrounding environment, determining its destination, planning a path to the destination, and adjusting that path whenever an obstacle has been encountered. The rover will also be able to sense when a ramp is in its direct path and will change the speed the rover is moving in order to climb the ramp at a normal operating speed.By being able to sense its environment, the rover will navigate to its destination while also avoiding obstacles.

Evaluation Plan

During development, each function of the rover and the PIC controller will be thoroughly tested at benchmarks with the use of the simulation tools, running in Matlab. The simulation will provide a GUI interface to change simulated data to be sent to each component to ensure proper handling. A data analyzer will be connected to the I2C bus on the controllers to monitor the messages being sent, ensuring proper structure, order, and response. Once a function is working in simulation, it will then be tested in real scenarios to ensure true functionality. Success will be determined by it's autonomous performance in the physical track.