|  |
| --- |
| HAROLD's Mechanics |
| Project Plan |
| Embedded Systems  Iteration Two |
|  |
| **Jenny Reiman** |
| **Andrew Kirkham**  **Jason Laqua** |

**10/1/2013**

|  |
| --- |
|  |

Table of Contents

Table of Contents 2

Requirements 3

Android UI 3

Communication between Vex and Android 3

Android can get GPS coordinates 3

Robot able to move to GPS coordinates 3

Robot move in straight line 3

Robot uses sensors 3

Cliff test 3

Risk Analysis 4

Android UI 4

Communication between Vex and Android 4

Android can get GPS coordinates 4

Robot able to move to GPS coordinates 4

Robot move in straight line 4

Robot uses sensors 4

Cliff test 4

Plan 5

# Requirements

### Android UI

The Android UI must incorporate buttons for remote control, returning the robot home, and entering waypoints.

### Communication between Vex and Android

An Android device can receive a command from a different Android device and send it to the VEXpro.

### Android can get GPS coordinates

The Android smartphone will need to receive a GPS coordinate from the Android tablet and send a confirmation back to the tablet when the robot has reached the given waypoint.

### Robot able to move to GPS coordinates

When a coordinate is received by the on-board Android smartphone, the VEXpro will need to be able to drive the robot to that location.

### Robot move in straight line

The robot will need to be able to drive forward in a straight line for 20 meters. This will be done with the use of the optical encoder sensor.

### Robot uses sensors

The robot will need to use an optical encoder sensor, an ultrasonic senor, and another senor of choice to aid the drivability of the robot.

### Cliff test

Using a sensor, the robot will need to be able to detect a sudden drop in elevation of the ground that it is moving on and stop driving forward.

# Risk Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement** | **Estimated Time To**  **Completion (hours)** | **Points** | **Risk** | **Final Value (points/hour)\*Risk** |
| Android UI | 1 | 20 | 1.0 | 20 |
| Communication Vex to Android | 4 | 20 | 0.7 | 3.5 |
| Android can get GPS coordinates | 10 | 15 | 0.5 | 0.75 |
| Robot able to move to GPS coordinate | 10 | 10 | 0.2 | 0.2 |
| Robot moves in straight line | 5 | 10 | 0.4 | 0.8 |
| Robot uses sensors | 10 | 10 | 0.6 | 0.6 |
| Cliff test | 2 | 15 | 0.5 | 3.75 |
| Totals: | 42 | 100 | N/A | N/A |

### Android UI

There are virtually no risks associated with creating the UI since most of it has been completed. This task will only involve adding a few new buttons and input methods.

### Communication between Vex and Android

The risk here is latency and performing consistent communication.

### Android can get GPS coordinates

Learning the Google Map API and retrieving coordinates from a waypoint are currently tasks that will need to be researched.

### Robot able to move to GPS coordinates

This poses the biggest risk since the GPS coordinate on the phone will need to be translated into a moveable operation by the robot.

### Robot move in straight line

This is dependent on being able to use the optical encoders to self-correct the path of the robot.

### Robot uses sensors

The main risk here will be learning how the sensors work and efficiently incorporating them into the robot mobility. The possibility of needing to use interrupts could pose additional problems.

### Cliff test

If the sensors fail to work for this test the robot risks damage from falling off an elevated surface.

# Plan

|  |  |  |
| --- | --- | --- |
| What | Who | When |
| Android UI | Andrew | Oct - 3 |
| Communication Vex to Android | All | Oct-10 |
| Android can send GPS coordinates | Andrew | Oct-10 |
| Robot able to move to GPS coordinate | All | Oct-31 |
| Robot Moves in straight line | Jason | Oct-17 |
| Robot Uses Sensors | Jenny | Oct-30 |
| Cliff Test | Jenny | Oct-24 |