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

**9/3/2013**

|  |
| --- |
|  |

Table of Contents

[Change Log 3](#__RefHeading__784_885007471)

[Requirements 4](#__RefHeading__786_885007471)

[Hello World Robot 4](#__RefHeading__788_885007471)

[Hello World Android 4](#__RefHeading__790_885007471)

[Robot and Android Not Damaged 4](#__RefHeading__792_885007471)

[Robot able to go Straight Forward, Straight Backward, Turn Right and Left 4](#__RefHeading__794_885007471)

[Obstacle Avoidance 4](#__RefHeading__796_885007471)

[Communication between Android and Android 4](#__RefHeading__798_885007471)

[Communication between Android and Vex 4](#__RefHeading__800_885007471)

[Robot Handles 10% Grade 4](#__RefHeading__802_885007471)

[Remote Control using Vex Radio 4](#__RefHeading__804_885007471)

[Risk Analysis 6](#__RefHeading__806_885007471)

[Hello World Robot 6](#__RefHeading__808_885007471)

[Hello World Android 6](#__RefHeading__810_885007471)

[Robot and Android Not Damaged 6](#__RefHeading__812_885007471)

[Robot able to go Straight Forward and Backward and turn Left and Right 7](#__RefHeading__814_885007471)

[Obstacle Avoidance 7](#__RefHeading__816_885007471)

[Communication between Android and Android 7](#__RefHeading__818_885007471)

[Communication between Android and Vex 7](#__RefHeading__820_885007471)

[Robot Handles 10% Grade 7](#__RefHeading__822_885007471)

[Remote Control using Vex Radio 7](#__RefHeading__824_885007471)

[Plan 7](#__RefHeading__826_885007471)

# Change Log

|  |  |  |
| --- | --- | --- |
| Date | Person | Reason |
| 8/29/2013 | All | Initial draft |
| 9/2/2013 | Andrew | Added more description to requirements |
| 9/2/2013 | Jenny | Edited/proof read document |

# Requirements

### Hello World Robot

We need to establish a connection with the Vex to which we can obtain debugging information and test any data flowing through the connection. This will be done through the serial port and a simple “Hello World” program will be written to verify the working operation of this communication.

### Hello World Android

We will be using the Android devices to communicate both with each other and with the Vex. We need to create a simple “Hello World” program on the Android to make sure we are able to communicate with these devices. This requirement will start out with the Android simply printing “Hello World” on a local trigger and then be expanded to print on an external trigger – such as receiving a command from another Android device.

### Robot and Android Not Damaged

We have only been given one of each item to build the robot. We should take care not to damage the materials to ensure correct working operation throughout the build. We also took advantage of the lockers in Engineering Hall.

### Robot able to go Straight Forward, Straight Backward, Turn Right and Left

The robot needs to be able to move in every direction in order to complete the final challenge. This requirement also details the actual construction of the robot. The robot should be able to move forward and backward until being told to stop, as well as being able to turn left and right until being told to stop.

### Obstacle Avoidance

The robot should eventually be able to avoid obstacles on its own. To start, we need to be able to avoid obstacles through manual control. This will ensure that the latency between applications does not impact the performance of the robot.

### Communication between Android and Android

We need to communicate with the robot through the Android devices. In order to do this, we need to be able to communicate from one Android device to the other. One Android device needs to host a Wi-Fi network that will allow the other Android to connect to it. We will then create an application to listen for and send commands. We will expand upon the “Hello World” application to display messages sent from one device to the other.

### Communication between Android and Vex

We will need to control the robot with an Android device. In order to do this, we need to establish communication between an Android device and the Vex. One Android device needs to host a Wi-Fi network that will allow the Vex to join it. We will use the same application to send commands from Android to Android to send commands to the Vex. We will need to figure out a way for the Vex to listen for these commands over the Wi-Fi network.

### Robot Handles 10% Grade

The robot must be able to travel up a 10% grade incline. This equates to a 5.71o slope in both the up and down direction. The robot should have sufficient torque to drive up the slope. It should also have its weight evenly distributed so it does not topple over on the slope.

### Remote Control using Vex Radio

The Vex Radio controller we were given should be able to control the robot. This will allow us to test the working construction of the Vex without needing the additional applications. The Vex Radio should be a simple plug-and-play operation.

Remote Emergency Stop Mechanism  
 In the event of a failure, we need to be able to stop the robot. This should be a button that will immediately cease power to the wheels. Some thought should also be given to automating this stop mechanism if the robot reaches a certain distance away from the home controller.

# Risk Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement** | **Estimated Time To**  **Completion (hours)** | **Points** | **Risk** | **Final Value (points/hour)** |
| Hello World Robot | 2 | 5 | .5 | 1.25 |
| Hello World Android | 1 | 5 | 1 | 5 |
| Robot/Android not damaged | N/A | 10 | 1 | N/A |
| Robot goes forward/backwards  Turns left/right | 5 | 10 | .5 | 1 |
| Obstacle Avoidance | 1 | 10 | .8 | 8 |
| Communication Android to Android | 10 | 10 | .1 | 0.1 |
| Communication Android to Vex | 8 | 10 | .2 | 0.25 |
| 10% Grade | 1 | 10 | .9 | 9 |
| Remote Control Using Vex Radio | 2 | 10 | .8 | 4 |
| Remote emergency stop | 3 | 15 | .8 | 4 |
| Totals: | 33 | 95 | N/A | N/A |

### Hello World Robot

The objectives in this task are unknown. We are inexperienced in operating the Vex. While it sounds simple enough, we are unsure how the internal connections will operate. Our team has sufficient experience with Linux and C code, leading to a greater confidence in the short time.

### Hello World Android

The basic requirement to communicate with an Android device is something that we feel is an easy task. Our team is experienced in both Android devices and with the Android SDK leading to a high confidence that this task will be short to accomplish. The communication with other Android devices is broken down in a later requirement.

### Robot and Android Not Damaged

A project locker will be used to store all the devices and equipment. The members of our group will be careful when handling devices. The materials are also fairly sturdy and we do not think that there will be a problem in taking care of the devices.

### Robot able to go Straight Forward and Backward and turn Left and Right

The task sounds very straight forward, and the API for the Vex makes it sound like this requirement shouldn’t be too difficult. However, this is new technology that none of us have dealt with leading to a higher risk factor and lower level of confidence. The majority of the time comes from the overall construction of the robot required to perform this task.

### Obstacle Avoidance

We interpreted this requirement to be a manual avoidance control rather than an autonomous one as Iteration Two introduces the sensors. Therefore, we perceived this requirement to be fairly easy with the time devoted to learning the controls and the operation of the Vex Radio and our own Android Application.

### Communication between Android and Android

The protocols that are in place to do this are unknown at the current time, and applications must still be written that can send and receive messages between the multiple devices. Our team does not have a lot of experience working with Internet protocols on the Android. We do have experience with Ad-Hoc networks, however. We anticipate this taking a long time to create a single application that will work on both Android applications. With the introduction of the new 4.3 API, we are unsure of the ease of accomplishing these tasks.

### Communication between Android and Vex

The protocols that are in place to do this are unknown at the current time, and applications must be written that can send and receive messages. The Vex requires a separate application for listening as well. We are unsure of the easiest method of creating an application to run on the Vex, as well as how the communication itself is going to work.

### Robot Handles 10% Grade

We don’t anticipate there being any problems in the construction of the robot. However, if the initial construction is unable to handle a 10% grade, it will most likely require a reconstruction of the robot.

### Remote Control using Vex Radio

The Vex Radio and the Vex should be plug-and-play which means it should work out of the box. We will need to make sure the robot will be able to work at a decent range away from the radio. We will also need to make sure the robot is properly constructed in order for it to be able to move correctly.

Remote Emergency Stop Mechanism  
 A button will need to be implemented to cut power to the device immediately. We may want a local button on the Vex in addition to one on the Android in the event of a Wi-Fi failure. We do not think this will be too difficult to implement given the Vex API, but getting the Android button to work properly may be a problem.

# Plan

|  |  |  |
| --- | --- | --- |
| What | Who | When |
| Hello World Robot | Jenny | 5-Sep |
| Hello World Android | Andrew | 5-Sep |
| Build drivetrain | Jason | 5-Sep |
| Attach motors | Jason | 5-Sep |
| Mount Vex to structure | Jason | 5-Sep |
| Plug in motors | Jason | 5-Sep |
| Remote Control Using Vex Radio | Andrew | 12-Sep |
| Robot goes forward/backwards/left/right | Jenny | 12-Sep |
| Obstacle Avoidance | Andrew | 12-Sep |
| 10% Grade | Jason | 12-Sep |
| Robot/Android not damaged | All | 19-Sep |
| Communication Android to Android | All | 19-Sep |
| Communication Android to Vex | All | 19-Sep |
| Vex Remote emergency stop | Jenny | 19-Sep |
| Setting up Ad Hoc Network | Jason | 19-Sep |
| Android Remote emergency stop | Andrew | 19-Sep |
| Send a command | Jenny | 19-Sep |
| Receive a Command on android | Andrew | 19-Sep |
| Receive a Command on vex | Jason | 19-Sep |
| Connect Android to Android | Andrew | 19-Sep |
| Connect Android to vex | Jenny | 19-Sep |