

SEMI-AUTONOMOUS 2-D MAPPING ROVER

1. Introduction:

This is a project on Arduino rover. The rover is controlled using an application created on android mobile – NEXUS 4, connected via Bluetooth. Rover has an ultra-sonic sensor connected to detect any obstacle. The rover's path is drawn on Android screen.

2. Project Build Specs:

Building Rover: Rover is constructed using the DFRobot manual and YouTube video (www.youtube.com/watch?v=MWWoEuI9Qsk).

Arduino Programming: Arduino software is used to program the rover.

Android Application: Eclipse (JUNO) is used to create an android application to interact with the rover. The application is programmed in eclipse and then installed directly to phone using USB connection.

A simple android interface is made using drag and drop option with 3 buttons. A button to connect to the paired rover via Bluetooth, button to send start command to rover, and a button to send stop command to rover.

Canvas is used in the same screen to draw the rover's traversal 2D-map.

3. The Project/Program:

Android Program:

On click of connect button, the phone and rover are connected via Bluetooth. Note: the rover and phone are already paired using the phone's generic Bluetooth connection. If the devices are not paired, then "Not Connected" notification is displayed on the phone.

If the rover is connected to the phone, then, on click of start button, rover receives a command which is validated on Arduino program. And, then, sets the rover in motion if the command received is true. If the command received is not from the start button, then, rover does not move forward.

On click of stop button, stop command is sent to the rover and slows down the rover to stop.

The application's got the control to start and stop the rover. However, avoiding the obstacle is programmed on Arduino.

Arduino Program:

Firstly, the ultra-sonic sensor was set with a program on EPROM to set the sensor on autonomous trigger mode. The method to do this is available on DFRobot.com.

Secondly, once the EPROM of ultra-sonic sensor is set with the auto-trigger mode, the main Arduino program makes use of ultra-sonic EPROM trigger mode to check for any obstacle within the threshold (defined as 16 cms in auto trigger function on EPROM). If any obstacle within the range, the rover moves to right, else, the rover sets in forward motion unless it receives a command from the phone to stop.

Forward motion is set only when the command from the phone is of char 's' and stop for char 't'.

4. Safety Critical Issues:

The rover and phone are paired outside the application. Rover's Bluetooth connects only with the one device at a time. So, the rover is secured before the application is started. However, the rover's Bluetooth uses a default pin. So, this could be hacked or controlled by any other if accessed before the actual user. This issue can be addressed with adding a login page to the application so that specific users can only access the application also, pairing the device only after logging in to the page and giving a specific authentication key to rover's Bluetooth.

5. Why this way?

Firstly, ultra-sonic sensor, if the EPROM is set with an auto trigger function, every time the sensor need not have to waste time on processing the sensor for threshold and autonomous mode.

Secondly, using the code set on EPROM to validate and function is easier and makes the Arduino code simple.

Following, through this approach, most of the critical part could be learnt and addressed first before moving on to develop the complex application. If the connection between rover and phone are established and found working, then it is easier to identify any further communication between the devices to be made with ease.

6. Concluding:

This project has helped to develop an understanding of how the communication between multiple devices/environment happens. Also, helps

to learn to tackle issues with communication, authentication and security between assorted platforms.

This project could be developed further by making rover move in all the directions. Also, the application could be developed with a switch between autonomous mode and manual control mode. Adding, Wi-Fi and Bluetooth control mode can also be implemented. Adding login page with users' authentication, which makes the application secure. Also, other sensors could be included to do multiple activities.

7. References:

<http://arduino.cc/en/Tutorial/HomePage>

http://www.dfrobot.com/index.php?route=product/product&product_id=53

<http://developer.android.com/training/basics/firstapp/index.html>