Multifunctional Remote Controlled Robot Using Android Application

T. A. Mithu[1], T. S. Reddy[2]

Jawaharlal Nehru Technological University Hyderabad

Abstract: *In this paper bidirectional remote controlled communication with robot is established using two android phones, one as transmitter and another as receiver. An embedded system is implemented with the Robot to execute the commands generated from Transmitter Android App and accordingly control the mechanical movements of Robot. Receiver Android Phone is mounted on Robot which establishes communication with Transmitter Phone and performs advanced tasks (i.e. sending messages, capturing photos, server uploads etc.) using Receiver Android App.*

Keywords: Transmitter, Receiver, Android, Robot, Microcontroller, Motor, Bluetooth, Camera, SMS, GPS, Server

1. Introduction

1.1 Research Objective

The main objective of this research is to develop effective android applications to implement an embedded system with Microcontroller by using which an efficient bidirectional communication with Robot can be established.

1.2 Proposed System

The proposed system uses the Bluetooth, GSM & GPRS/Wi-Fi Technology. Simpler but effective, cheaper and faster robotic operations are implemented with the system, consisting of:
- **Transmitting End:** An Android App is developed to search and connect to Bluetooth device, provide GUI to user and transmit commands to receiving end.
- **Receiving End:** There is a Bluetooth device to receive the serial data sent by transmitter. 8051 Series Micro Controller is implemented as control device. A Motor driver is used to execute movements of Robot.
- **Two-way Communication:** There is a smart phone mounted on robot. One android App is developed and installed in that phone. After getting command from Transmitter it can perform advanced tasks and communicate with Transmitter.

2. System Details

2.1 Modules Description

The system after careful analysis has been identified to be presented with the following four modules mainly:
- Transmitter Module
- Electro-Mechanical Module
- Receiver Module
- Server Module

2.1.1 Transmitter Module
1) Transmitter Android Phone: One smart phone is used having Android OS installed in it. Any version of OS ranging from Android Gingerbread (Version 2.3) to Android Lollypop (Version 5.0) will work for this study.

2) Transmitter App: An Android App is developed which has the features like searching for Bluetooth devices to connect with robot, establish connection with robot and camera device that mounted on robot.
3) Robot control unit of Transmitter App can send signals or command to control mechanical movement of Robot.
4) Camera control unit of Transmitter App can send commands to camera device on robot to perform different tasks like gathering location information, capturing photos, saving into SD card, uploading into server and sending informative SMS to the transmitting end.

2.1.2 Electro-Mechanical Module:
1) Bluetooth Receiver: It's a chip that is attached with Arduino board at receiving end. It receives the signal from transmitter. Before that Bluetooth receiver should be paired with Android device of transmitting end. There is an indicator LED with this device which helps to identify the status of the connection (fig-2).
2) Microcontroller: Bluetooth receiver is interfaced with 8051 series Microcontroller. Transmitted signal works as input to the Microcontroller, which is programmed using embedded C. Movement of robot is defined by Microcontroller for respective command from transmitter.
3) Motor Driver: Microcontroller's outputs are connected to a Motor driver IC. For the different combinations of output signal from Microcontroller, Motor driver controls the movement of Robot.
4) Robot unit: In this study main robot module consists of two wheels which can be controlled by Android App. Additionally there is another module consists of Android phone that is mounted on Robot.

2.1.3 Receiver Module
1) Bluetooth App Launcher (Auto Plug in): It helps to start the receiver application automatically upon receiving trigger file from Transmitter.
2) Photo Capture Unit: This unit is able to capture photo and save into the SD card programmatically.
3) Image Uploading Unit:It can upload photo from SD Card and save into the Server and send a response to the client Application.

4) **Location Management Functions:** This unit can enable GPS to capture longitude and latitude of current position of Robot.
5) **SMS Management Functions:** This portion is responsible to send SMS to Transmitter after successful operation including location information.

2.1.4 Server Module
1) **Server Scripting Unit:** This unit can handle image upload and manage server response after completion of task.
2) **Programmer Components:** These components are used to burn the Embedded-C program into the Microcontroller.

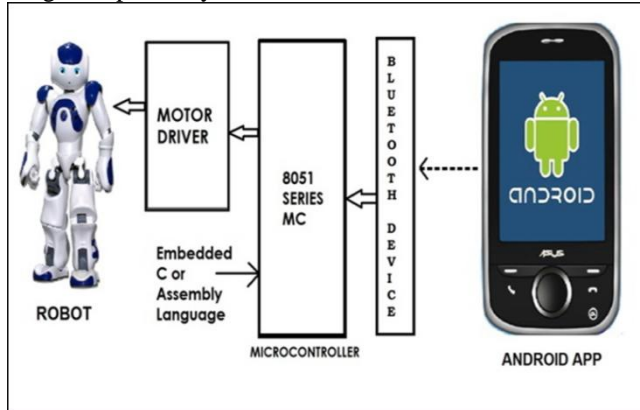System overview & practical setup have been shown in fig-1 & fig-3 respectively.
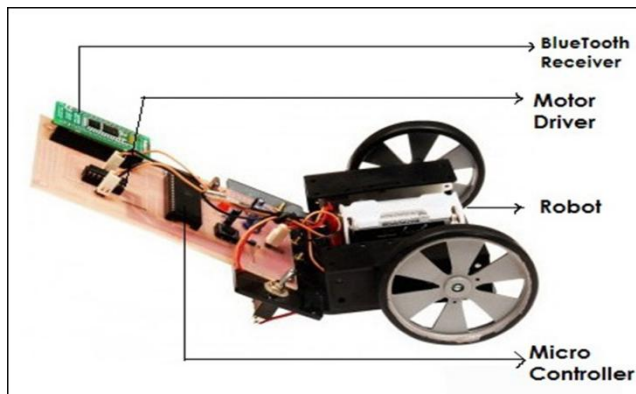

Figure 1: System Overview

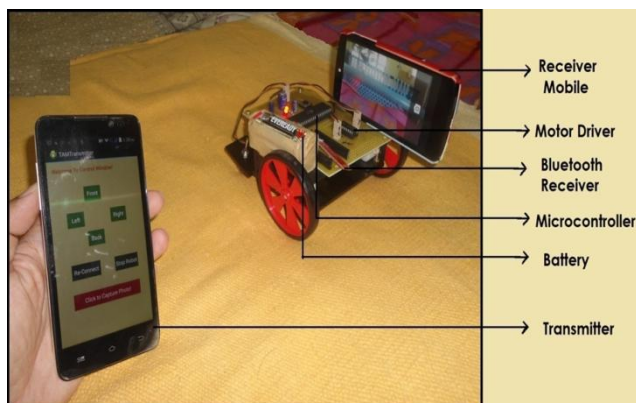**Figure 2:** Electromechanical Module

Figure 3: Practical Set Up

1.2 Software & Hardware Requirements

- Software Requirements: Eclipse IDE, Java & XML for Android Programming, PHP for server side programming, Embedded C for microcontroller programming using Keil Compiler, ProgISP software & driver as ISP programming tool, Bluetooth App Launcher plugin and ArgoUML as design tool.
- Hardware Requirements: Transmitter Android Device, Receiver Android Device, Microcontroller ATMEL 8051, Motor Driver L293D, Bluetooth device, Robotic & mechanical components, 8051 USB programmer, Power source, Circuit components (i.e. resistor, capacitor, battery, switches etc.).
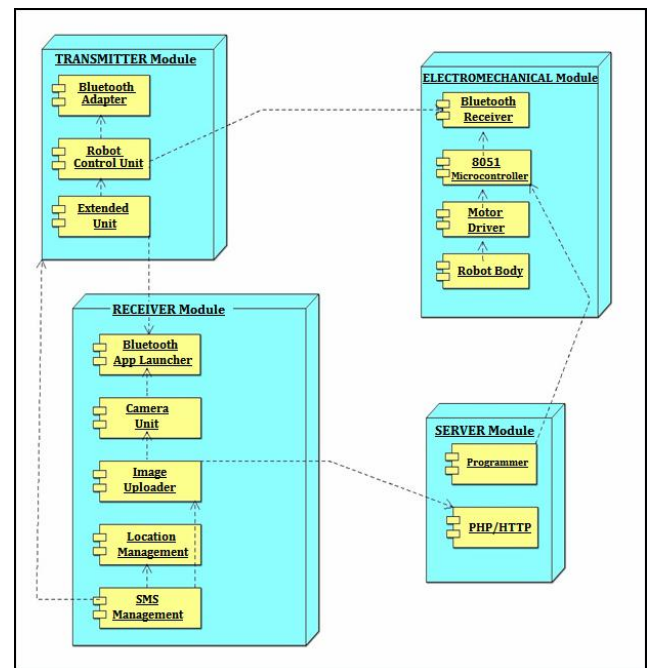
3. System Implementation

Figure 4: Deployment Diagram of All Modules

3.1 Hardware Implementation

Major components of electromechanical module such as Bluetooth Device, Motor Driver IC and 8051 Microcontroller are shownin the fig-5.Here serial data from transmitter enters into Microcontroller through pin no 1 of Port-1 via Bluetooth Receiver (fig-5). For a particular input different combination of outputs are generated at port-2, defined by program. These outputs are connected to input pins of motor driver IC. There are two motor drivers in motor driver IC, which can be enabled or disabled via two separate enable pins. With the different combination of enable pins and input pins, output pins will be activated or deactivated to run the motor of the robot. For example, if pin combination of motor driver IC becomes IN1 = 0, IN2 = 1, IN3 = 0, IN4 = 1, EN1 = 0, EN2 = 1, then first motor will be deactivated but second motor driver will be activated and its output will be in high state, which can move the robot in one of the directions. These configurations are determined by microcontroller programming.
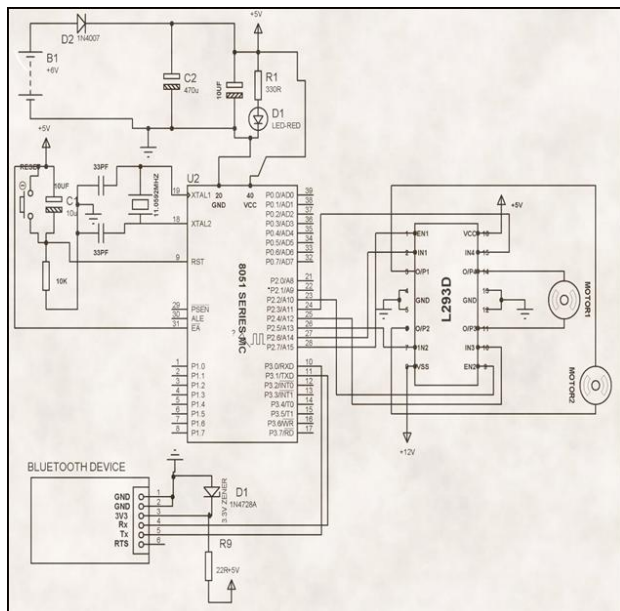
Figure 5: Hardware Implementation Diagram

3.2 Software Implementation

3.2.1 Microcontroller Programming:
Keil compiler has been used with Embedded C
programming.
- Commands sent from transmitter are passed through Switch-Case statement.
- For different commands different binary combinations of values are generated at specific output pins, which are connected to input pins of motor driver IC.
- Hex file is generated before burning the embedded C program.

3.2.2 Android App for Transmitter:
Four Activities are designed to handle different tasks of
Android Appat transmitting end (fig-6).
- 1st Activity will display the welcome screen(i.e. Splash Screen before opening control window).
- 2nd Activity will handle the Bluetooth device discovery functions(i.e. Bluetooth On/Off, List paired devices, Search new Devices & Close connection).
- 3rd Activity will send message/signal to the robot(i.e. buttons to move robot forward, backward, right & left).
- 4th Activity will handle the additional functionalities of robot (i.e. taking photo, get location information&server uploads).
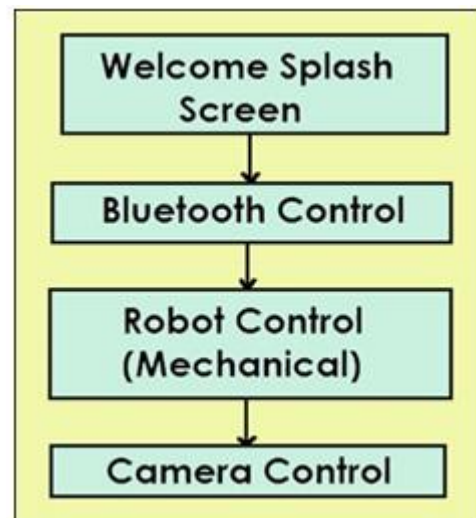

Figure 6: Transmitter Android App Activities

3.2.3 Android App for Receiving End/Robot:
App Features:
- Able to start automatically upon receiving transmitter's command.
- Able to take photo by using custom camera.
- Able to save photo into a folder in SD Card.
- Able to upload photo into Server from SD Card.
- Able to get location Information.
- Able to send SMS to Transmitter.

The process flows of location management, SMS management, camera management and server upload management have been shown in the figures 7, 8, 9 & 10 respectively.

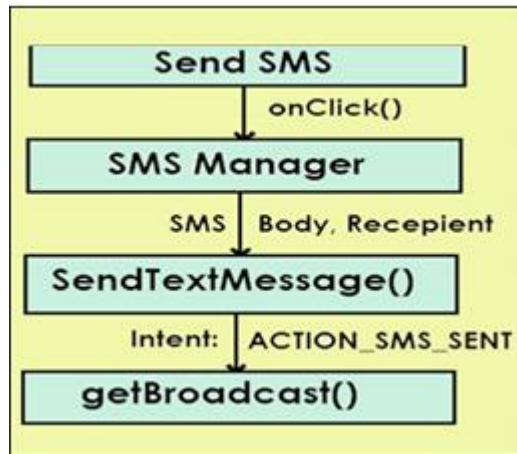**Figure 7:** Location Management with Receiver App

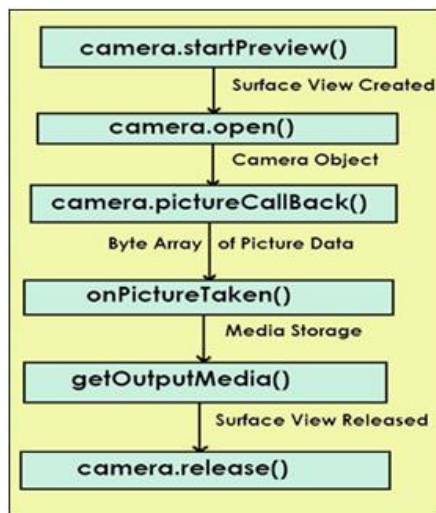**Figure 8:** SMS Management with Receiver App

Figure 9: Camera Management with Receiver App

3.2.4 Server side scripting for Server Upload:
PHP Scripting has been used as server side programming for photo upload. Steps included with PHP Scripting are mentioned as follows:

- Step 1: One PHP file is generated at the server location. The functionalities required to upload a photo such as identifying destination folder to save the photo, 'copy or move' mechanism of photo from client to server, handling server response upon task completion etc. are defined in the PHP script.
- Step2: Destination folder named as it is mentioned in PHP script is created at the server location to save the incoming photos.
- Step3: From Client Application (Android Receiver App) server URI is called using URL object to establish an http connection between Client & Server for photo transfer.
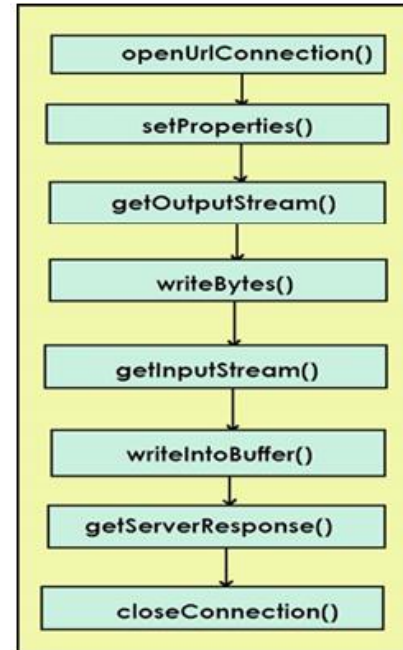

Figure 10: Server Upload Management with Receiver App

4. Future Scopes

- Due to the limitations with programmatic data sharing via Bluetooth or Direct Wi-Fi, in this study captured photos by Receiver Module have been uploaded into the server for monitoring rather than sending back to Transmitter mobile. It would be a significant advancement if captured photo can be sent back to transmitter Mobile from Receiver Mobile programmatically by utilizing cheap and effective mechanism.
- More robotic functions like capturing videos, recording sound, monitoring and gathering live information, playing alarm or alert, sensors activity etc. various types of smart and advanced features can be added and implemented via Receiver Android App in future.
- In this study, a simple two wheeler robot has been used. Instead multifunctional robot having several mechanical parts can be tested for advanced operations. It might include DC gear motors, Ultrasonic sensors, Light sensor, IR Sensors, IR remote control, Gripper with touch sensor etc. and Advanced Transmitter and Receiver App could be designed to handle additional functionalities.

5. Conclusion

- At transmitting end both the sub modules are functioning properly as android app at transmitting end can successfully send control signals to microcontroller and also communicate with Receiver Android Phone.
- At receiving end all the sub modules are functioning properly as Robot is responding to the commands of the transmitter, Receiver Android phone is responding as expected and all the required tasks are performed perfectly at receiving end.
- Much of the time spent in designing and understanding the entire system. Testing and Debugging were also significant sections of this study.

- The major challenges are faced while working with socket connections through Bluetooth. Devices of higher API level require additional methods to be implemented for socket connection.
- Overall system is functioning well and as expected.

6. Acknowledgement

Reference

Bibliography:
[1] Schildt H. 2006, 'Java: The Complete Reference (7th Edition)'; McGraw Hill Education Private Limited.
[2] Keogh J. 2003, 'J2ME: The Complete Reference'; McGraw Hill Education Private Limited.
[3] Mazidi M.A., Mazidi J.G.,McKinlay R. 2007 "The 8051 Microcontroller and Embedded systems (2nd Edition)"; Pearson Education.

Web References:
[4] Android Developer, http://*developer.android.com*
[5] Robotic Technologies, *http://www.intorobotics.com*
[6] Project Management, *http://www.startwright.com*
[7] Online Android Courses,*http://courses.oreillyschool.com/*
[8] Online Programming Tutorials, *http://www.pickatutorial.com*
[9] Wikipedia, *http://www.wikipedia.org*