IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 8, August 2015.
www.ijiset.com
ISSN 2348 – 7968

Android controlled Integrated Semi - Autonomous Fire Fighting Mobile Robot

Saravanan P[1], Soni Ishwarya V[2] and Padmini K M[3]

[1]Assistant Professor, Department of Electrical and Electronics Engineering, Ganadipathy Tulsi's Jain Engineering College, Anna University
Vellore, Tamilnadu, India.

[2]Assistant Professor, Department of Electronics and Communication Engineering, Ganadipathy Tulsi's Jain Engineering College, Anna University
Vellore, Tamilnadu, India

[3] Anna University
Chennai, Tamilnadu, India

Abstract
The Era of human machine interface is far more advanced and effective now. Its goal is interaction between the user and the machine in an effective manner where user can easily control various parts of the machine in-order to meet the desired results. These may be started through pen and membrane switches but it's deviated and gestures became pervasive and compatible in this modern world. Here in this paper we analyzed a method to control all the kinetics of fire fighting mobile robot using a blue-tooth module installed in the unit through a smart-phone which operates on android operating system.

Keywords: **Atmega2560, Blue-tooth module, HC-05, SA-BOT, Firefighting Robot, Android studio.**

1. Introduction

Now a days machines have become essential parts of human life and robots are designed in order to minimize the discomforts thereby making life easy. Robots are used in variety of fields in order to minimize the difficulties like medical, space exploration, underwater exploration, defense and humanoid robots. The Fire extinguishing robots are one of them which were designed to replace humans from any kind of hazards. Smart-phones are mobile phone built on a mobile computing platform, with advanced computing and connectivity features. These are much more affordable and effective to use in these days as a result there are more advancement in mobile phone technology. Day by day humans are exploring new ways to communicate and control machine since their majority of daily life deals with machines. Gesture is a form of non verbal communication method where bodily actions are used as communication parameters. These Gesture recognition technologies pave a way to a huge development on human machine interface. Modern smart-phones are embedded with various types of sensor like accelerometer, gyroscope, proximity sensor, light sensor etc.. and powered by various operating system like apple IOS, Blackberry OS, Windows OS, BADA, Symbian, Android, Web-OS etc. Among them Android usage is very common these days in mobile world. For communication smart phones have Wi-Fi module, Blue-tooth modules. We are concentrating on this communication module in-order to communicate and control our mobile robot most particularly blue-tooth [2][3][4]. Here data are transferred serially to our semi autonomous fire fighting mobile robot (SABOT) via blue-tooth communication module which is already installed on smart phones using an android application. Based on the received data the controller performs a particular action appropriate to the data received.

2. System Architecture

Fig.1 Full System Architecture

Figure 1 shows the block diagram of whole system's architecture
The system architecture of SA-BOT is shown in figure below. SA-BOT contain three different kind of system unit 1. Locomotion system 2. Fire detection system 3. Extinguishing system 4. Communication system.

The four integrated ultrasonic sensor and infra red sensor forms the location system, LDR and thermistor forms the detection system, water container and sprinkler forms the extinguishing system and the communication system is by the blue-tooth module through which the locomotion of robot are also controlled [1].

2.1 Locomotion Unit

SA-BOT uses four IR sensor to detect the obstacle and four ultrasonic range finder to find the distance between

ISSN 2348 – 7968

the BOT and obstacle. We have integrated the IR and ultrasonic range finder to get the layout of the whole environment.

2.2 Fire detection unit

SA-BOT's fire detection unit comprises of LDR and Temperature sensor. When there is a fire there will be a potential difference in LDR which is proportional to intensity of light and it is fed to the micro-controller as input and in-order to prevent false triggering from ambient we are integrating LDR with temperature sensor.

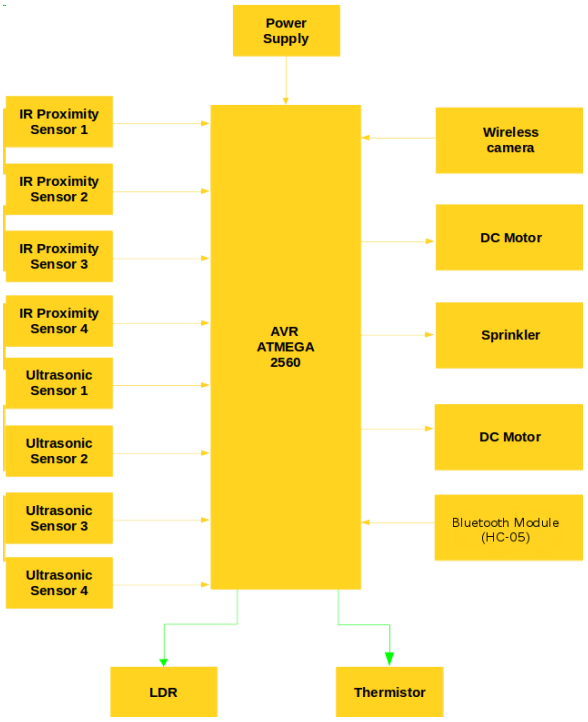


Fig.2 SABOT's Architecture

2.3 Extinguishing Unit

Here the water container and BLDC motor forms an extinguishing unit. Inputs from the fire detection unit are fed to the micro-controller, and the locomotion unit helps the BOT to reach the correct location ie, where the fire is, if the BOT reaches the correct position and the detection unit is activated then the controller activates the extinguishing unit (i.e) water is sprayed by the sprinkler and the fire will be extinguished. When the data is received from the smart-phones, the locomotion and extinguishing unit stops its autonomous operation and becomes semi- autonomous. Then the total navigation of SABOT is controlled from android app which sends the control commands for robot's navigation.

3. Hardware Module

Micro-controllers play a vital role here in controlling the different units of SA-BOT. We have chosen At-mega2560 as a controller which is low powered 8 bit AVR RISC based controller combines 8KB of SRAM, 4KB of EEPROM, 86 GPIO's , Six flexible timers /counters with compare mode, 10 bit A/D converters for signal conversion and 4 USART's which supports serial Communication between controller and peripheral. Throughput of controller is defined by the instructions executed per clock, powerful single cycle execution provides a high code density and device achieve a throughput of 16 MIPS at 16 MHz operates between 4.5-5.5 volts.

We are using two +12v DC motors for BOT locomotion purpose these two motors are connected with micro-controller through LD293D which operates the motors in bidirectional conditions. Blue-tooth module HC-05 is based on serial port protocol which was designed for transparent wireless serial communication. BluetoothV2.0 and enhanced data rate, 3Mbps modulation with complete 2.4GHz radio transceiver and baseband. When compared with HC-06, HC-03 and HC-04 this module HC -05 can be switched as master or slave thereby making it perfect for serial wireless communication. We are using Moto-G (generation1) as a smart phone for control application which has blue-tooth technology version 4.0. Where the app was installed.

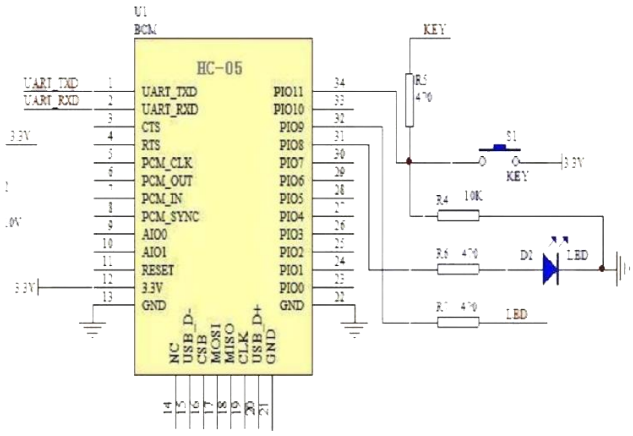


Fig.3 HC-05 Blue-tooth module

4. Software Module

Android Applications are generally developed using JAVA language there are various IDE's which have been used to build the applications. Here we are using Android Studio for creating android based application. It's an official ID for Android Development based on IntelliJ IDEA. It offers grade based build system design,

ISSN 2348 – 7968

multiple .apk file generation, inbuilt code template to help us build common feature of an applications.

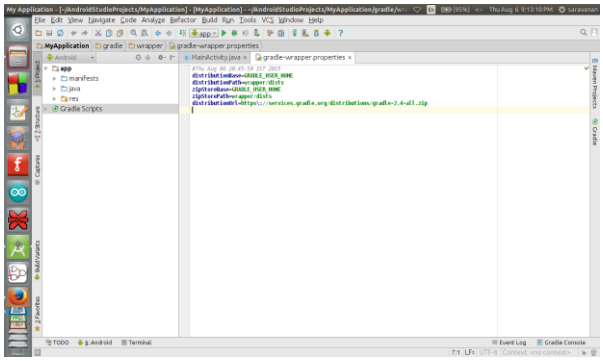


Fig.4 Android Studio IDE Layout

For debugging android studio have improved assistance in debugging which includes virtual device management, various performance analysis tools and in-line debugging which help us to perform rubber duck debugging methodology. Android AVD manager gives us the various mobile platforms as an emulator in order to debug the code on various device configuration, screen size and resolution for your app previews. These AVD managers have Nexus 6 and Nexus 9 Emulators which has different skins based on various hardware available in the market so it gives us a wide range of debugging facilities.

Android studio supports for the blue-tooth network stack, since this paper mainly concentrates on the communication between the robot and smart-phones through blue-tooth these network stack enables serial communication between the devices [5]. The functionality is made by Blue-tooth APIs which allows point to point communication between the devices. For initializing blue tooth functions permissions has to be enabled. The permissions are BLUETOOTH_ADMIN and BLUETOOTH these must be enabled on application manifest of project file. *BluetoothAdapter* is required for all blue tooth related activity. It calls a static function *getDefaultAdapter()* and *BluetoothAdapter* objects and its returns null value if there is no device support. After invoking the BluetoothAdapter we have to ensure that the Bluetooth on the device is turned ON.

For the ensuring purpose the fuction isEnabled() is called if this method returns false condition, the Bluetooth is in OFF Condition. In order to Switch ON blue-tooth *startActivityForResult()* and ACTION_REQUEST_ENABLE (action Intent) is called for and as a result dialog box in figure.5 will appear. If the user selected yes then it will be turned ON.

Once the device is turned ON it has to be discovered in order to initiate communication start activity for Result(INTENT,int) with the ACTION_REQUEST_ DISCOVERABLE action intent this will set the device in discoverable mode. Once its initiated device will be discoverable for the next 120 seconds and it can be extended by EXTRA_DISCOVERABLE_DURATION Intent.

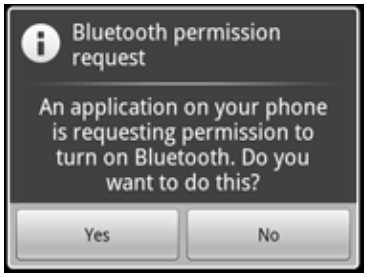


Fig.5 Permission Dialog box

Now the blue tooth on the smart-phone acts as a master and the blue-tooth device HC-05 operates in slave mode which is one of its three different operating modes. The device paring is initiated by calling listenUsingRfcommWith SerivceRecord() then have to wait and listen for the connection requests by evoking the accept() function once the slave device is found the pair is exchanged between the devices.

Once the communication is initiated the control commands starts to send from the mobile app appropriate to the user key through write (byte []). Here we have designed six key for forward, reverse, left and right for Robot navigation and two separate keys to control blue-tooth and BLDC motor which make the water spraying unit.

4. Results

Our codes are tested with an emulator and the results are shown in figure 6. Once the coding is done an apk file is created using Generate signed apk option from Build menu in Android IDE. After creating the apk file is installed in moto-g smartphone, it is tested as shown in figure 7.

Fig.6 Testing out app with Nexus 5 Emulator

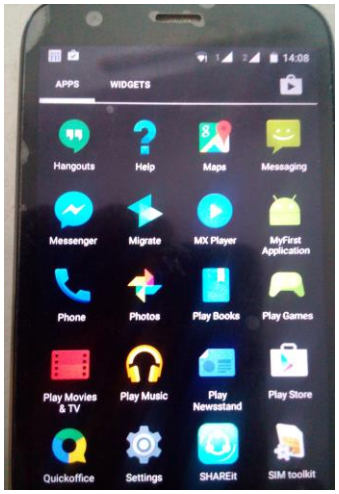
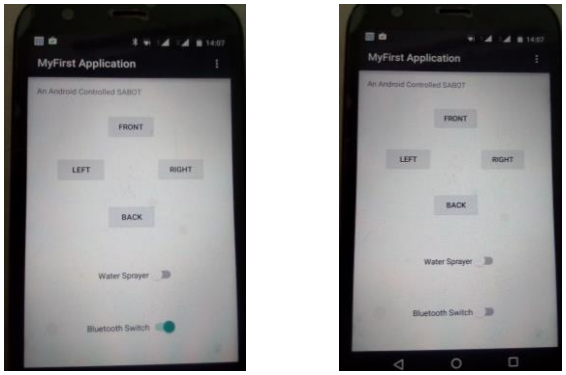


Fig 7 Installed Android Application



(a) (b)

Fig 8 (a) Bluetooth ON (b) Bluetooth OFF

Fig 9 SABOT

5. Conclusions

We have developed an app using android studio and it has been tested with moto-g smartphone and the results are compromising.

References

[1] Saravanan P "Design and Development of Integrated Semi -Autonomous Fire Fighting Mobile Robot "International Journal of Engineering Science and Innovative Technology Volume 4 Issue 2 March 2015

[2] Loïc Frund "Interfacing an Android-based phone robot with webots" Master thesis project Ecole polytechnique fédérale de Lausanne (EPFL) August 2012.

[3] Angel Gonzalez Villan, and Joseph Jorba "Remote Control of Mobile Devices in Android Platform" Universitat Oberta de Catalunya, Barcelona, Spain 22 Oct 2013.

[4] Jong Hoon Ahnn "The Robot Control using wireless communication and Serial Communication", Design Project Cornell University May 2007.

[5] http://developer.android.com/tools/studio/index.html

First Author Saravanan P, received his B.E Eng Degree in Electronics and Communication Engineering from Anna University in 2009 and Master of Technology in Embedded systems from Hindustan University, Chennai in 2012. His research areas include Distributed Algorithms, embedded OS, Automotive Embedded Systems and Embedded system design. Currently he is working as an assistant professor at the department of Electrical and Electronics Engineering Ganadipathy Tulsi's jain Engineering College Vellore. He published his papers in various journals and conferences.

Second Author Soni Ishwarya.V, received her B.E Eng Degree in Electronics and Communication Engineering from Karunya University in 2009 and Master Degree in Embedded system and technologies from Anna University, Chennai in 2011. Her research areas include embedded system design, Signal Processing. Currently she is working as an assistant professor at the department of Electrical and Communication Engineering Ganadipathy Tulsi's jain Engineering College Vellore.

Third Author Padmini K M, received her B.E Eng degree in Electronics and Communication Engineering in 2012 from Priyadarshini Engineering College and Master Degree in Embedded System Technologies from Anna University Chennai in 2014.