

Controlling Home Appliances by Using Universal Remote Control System (IoT and Bluetooth)

K. NarendraReddy (M.Tech), P.SUKUMAR

K. NarendraReddy (M.Tech) , Embedded Systems, AITS, Rajampet, India

P.SUKUMAR, professor, ECE, AITS, Rajampet, India

Abstract - *The smart phone is one of the representative fields of Internet of Things (IoT). The intelligent home system creates the more comfortable, safer and intelligent living environment. It can be resolve the problems facing by the people who have spend less amount of time at home .For the solution of this problem peoples can depends on the automated machines and smart phones. In existing systems users may be confused with the controller to simple operations. These problems are overcome by communication between remote controller and target devices. To design a circuit for Internet of Things (IoT), Wi-Fi and Bluetooth technologies are used to controlling home appliances on ARM board by using universal remote controller. The circuit is designed with less complexity, security will be provided in the controlling and with low power by taking advantage of the concept of multiple data fusion technique.*

Key Words: Remote control, IoT, smart home, user-friendly UI, ARM Board, Android studio.

1. INTRODUCTION

In recent years, the rapid development of Internet of Things (IOT) makes the intelligent home as people expect. The intelligent home system creates the more comfortable, safer, humane and intelligent living environment. It can resolve the problems facing by the people who have busy schedules and get a very less amount of time to spend at home. which is increasing rapidly around the world. For the solution of this problem, user can depend on the automated machines and gadgets like smart phones. The need for comfort and a convenient life are especially important in smart homes.

The goal of this paper is to design a circuit for Internet of Things (IoT), Wi-Fi and Bluetooth technologies are controlling home appliances by using universal remote controller using ARM board. The circuit is designed with less complexity, security will be provided in the controlling and with low power by taking advantage of the concept of multiple data fusion technique. The data can be seen on mobile phone through IoT, Wi-Fi and Bluetooth technologies and develop an intelligent universal remote control system for home appliances. A user interface for controlling this device is immediately displayed on the screen of the controller. UI's, which enables users to simply enable and

control the target device among the complex functionality of home devices in a shared space for IoT based smart homes.

Now a days, the majority of devices and appliances in smart homes are equipped with a remote controller, which includes a number of buttons and wireless transceivers. This setup provides higher operational complexity around the space with numerous devices or appliances. Thus, the idea of the universal remote controller is introduced to integrate multiple functions of home devices or applications into one single remote controller. Nevertheless, various functions and buttons of a URC results in more complicated operations, the problems of intuition and user-friendliness remain. Numerous solutions are proposed to develop URCs with a Liquid Crystal Display (LCD) screen, networking capability, and several techniques. Built in with these techniques, the device or appliance can be automatically detected via a network and a UI is dynamically generated from descriptions and properties of the device or appliance.

2. EXISTED SYSTEMS

Prototyping design of Electronic End Devices for Smart home appliances is one of the existed system. In this system end-devices prototyping design for supporting several main electronic based applications which are commonly found in a home. Motivation of this work is to propose end -devices prototyping design which are (1) to represent several main electronic functions in a common Home, (2) Able to support our existing and efficient smart home platform and (3) Able to provide smart functions on existing appliances without any major changes and modifications. Home appliances controlled by Bluetooth and Wi-Fi. Smart home switches to connect and disconnect by using IOT are some other relative existed systems.

3. PROPOSED SYSTEMS

In this paper discussed different Home Automation System with their technology with features, benefit and limitations. In this my project, the Home automation system that uses IoT, Wi-Fi, Bluetooth technology. System consists of three main components; web server, which presents system core that controls, and monitors users' home and hardware interface module, which provides appropriate interface to sensors and actuator of home automation system. The

System is better from the scalability and flexibility point of view than the commercially available home automation systems. The User may use the same technology to login to the server web based application.

Smart homes that have appliances lighting or any electronic devices that can be controlled remotely by the owner often via a mobile application. An android phone is a smart phone running on Google’s open source android operating systems.

A smart phone equipped with infrared (IR) capabilities is easier to realize functions of PPRC. Otherwise some IR USB dongles that can provide support for android operation systems. And several open source universal plug and play (UPnP) libraries are useful to help us to reduce the development time. Finally, for implementation of the PPCB, the design of the small and narrow hole is trick for the IR mechanism. The comprehensive system architecture and its primary components as shown below.

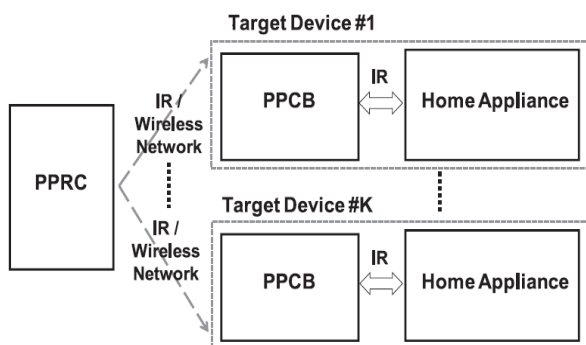


Fig 1: System architecture of the proposed system

3.1 System Architecture

The system architecture consists of two parts: 1) the Point-n-Press remote controller (PPRC) and 2) a number of target devices, which embed in the Point-n-Press Control Box (PPCB) for interfacing with the PPRC, as shown in Fig 1.

The functions of the components in the PPRC are described as follows.

1. IG (Interface Generator)
2. DPR (Device Profile Registry)
3. URC controller
4. Android Platform
5. Communication Interfaces

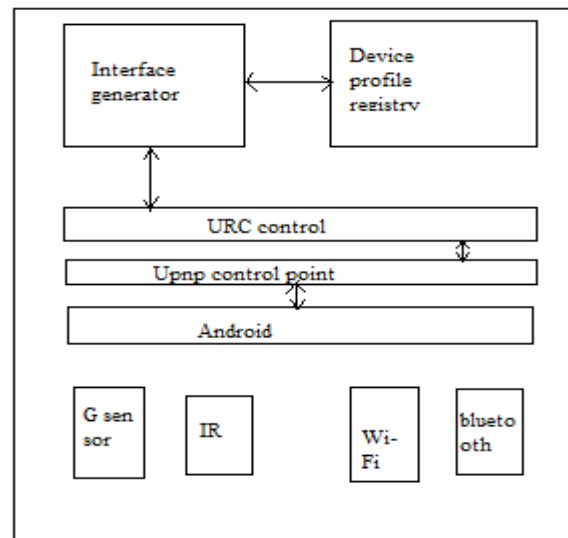


Fig 2: Software stacks for the PPRC

IG stands for Interface Generator. The Interface Generator creates a UI according to the properties and descriptions of the target devices and its current state. Device Profile Registry store the information of the current target device, such as its current state and dependency between each state. URC Control Responsible for receiving DCPs from target devices and analyses the DCP and the current state of the target device to perform additional control operations.



Fig 3: Universal Remote Controller (Android Mobile phone)

Communication interfaces are used between the Universal Remote controller (Android Mobile Phone) and the target devices, including Bluetooth, Wi-Fi, ZigBee and IR.

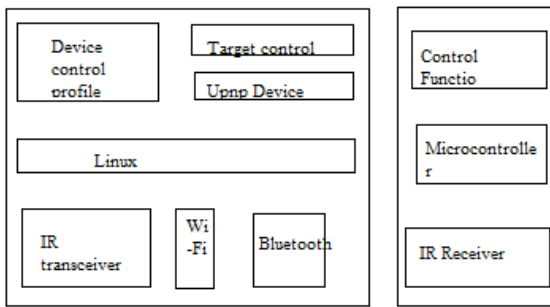


Fig 4: Software Stacks for the PPCB



Fig 5: LPC2129 ARM-7 Board

Device Control Profile is a file that specifies the properties and descriptions of a target device and remote controller controls appliances by the transmission of messages. Underlying information communication interfaces use an IR transceiver to receive detection from the Remote Controller when the Remote controller is pointing to the IR transceiver of the control Box. Target Control generates DCPs and corresponding DCPs to the Universal remote Control of the remote controller. Linux platform is applied as the operating platform of the control Box. UPnP Device executes the received UPnP Control commands.

3.2 Device Control Profile

A remote controller controls the appliances by the transmission of messages. The XML messages increases the convenience of discovering and controlling appliances, the complicated control formats and sizes consume significant bandwidth, produce heavy loads and create bottlenecks in the network transmission.

Thus design of the DCP includes:1) A finite state machine that represents the dependencies of every operational state of appliance and 2)multiple bit-string formed.

4. SYSTEM IMPLIMENTATION

The feasibility of the proposed control system, two real prototypes is implemented, including a fan and a light control system.

A. LPC2129 ARM-7 Board

The control Box is embedded with an ARM7-based development board, which is equipped with 200 MHz, 1M bytes Flash, 64 M bytes NAND Flash, 64 M bytes SDRAM and 22 general purpose input/output (GPIO) interfaces. The internal software components are developed based on the Linux 2.4.18 version.

The communication between the PPRC and the PPCB are IR, Bluetooth, and Wi-Fi. The control gateway is equipped with 2.3 GHz, 2G bytes SDRAM, two USB connectors, and one USB IR receiver. The internal data transmission between the control gateway and the power line controller uses an USB cable. The plugs from 1 to 8 are connected to a power line via a power line network.

Two control prototypes have been launched in IoT-based smart homes. The Point and Press Remote Controller is implemented on the platform of a commercial smart phone, which is equipped with 1.2 GHz, 1G bytes Random Access Memory (RAM), 16G bytes Read Only Memory (ROM), a Bluetooth Communication, a Wi-Fi Communication, a G sensor, and a micro-universal serial bus interface. In additional, all software components are developed based on the Android 4.1 version.

5. APPLICATION DEVELOPMENT

In this project we develop the android application for controlling the home appliances. Now a day so many android application developed tools are available like as Android Studio, Eclipse, Appcelerator, Xamarin.. etc.

Eclipse is a java based open-source platform that allows a software developer to a create a IDE from plug-in components built by Eclipses members . It is start in 2001 when IBM donated 3 million lines code from its java tools. Appceletor titanium is an open source frame work that allows the creation of mobile applications on platforms IOS, Android , windows phone from a single java script code base developed by Appcelerator. Appcelerator studio IDE at \$36 per month per seat. Xamarin is Microsoft-owned scan-Francisco, California-based s/w company founded in my mono for android, mono for touch.

Android Studio (2013) is the official IDE for android platform. It was announced on May at the Google-I/O conference.

In this paper we are develop the android application by using Android Studio Tool. It supports gradle (this allows you to really have control over the build , create different application flavors, different signing configuration and so on). In the Android studio layout view you have the option to view both the actual layout and the xml at the same time, while in Eclipse you must choose between the two tabs. It feels better. From doing a refactoring to viewing the Logcat The refactoring seems more solid and the Logcat does not “disappear” as it happens in Eclipse. Android Monitor that displays the debug messages. It display messages in real time and also keeps a history. So we can view older messages. Logcat is a command line tool that dumps a log of s/m messages including stack traces when the devices thous an error and message that you have written from your application with log clear. Its the future. Sooner or later this is going to be the standard, as Google is actively developing it.

5.1 Application Development Flowchart

Android Studio is the official integrated development environment (IDE) for the Android platform. Android Studio was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. It is available for download on Windows and Linux and replace Eclipse Android Development Tools (ADT) as Google’s primary IDE for native Android application development.

In this paper the application developed flow chart of develop the Android Application as shown below. Smart homes that have appliances lighting or any electronic devices that can be controlled remotely by the owner often via mobile Application. An android phone is a smart phone running on Google’s open source android operating systems.

Flowchart management:

Most management of the life cycle is done automatically by the system via the activity stack.

The activity class has the following method call backs to helps you manage the application steps:

- onCreate()
- onStart()
- onResume()
- onPause()
- onStop()
- onRestart()
- onDestroy()

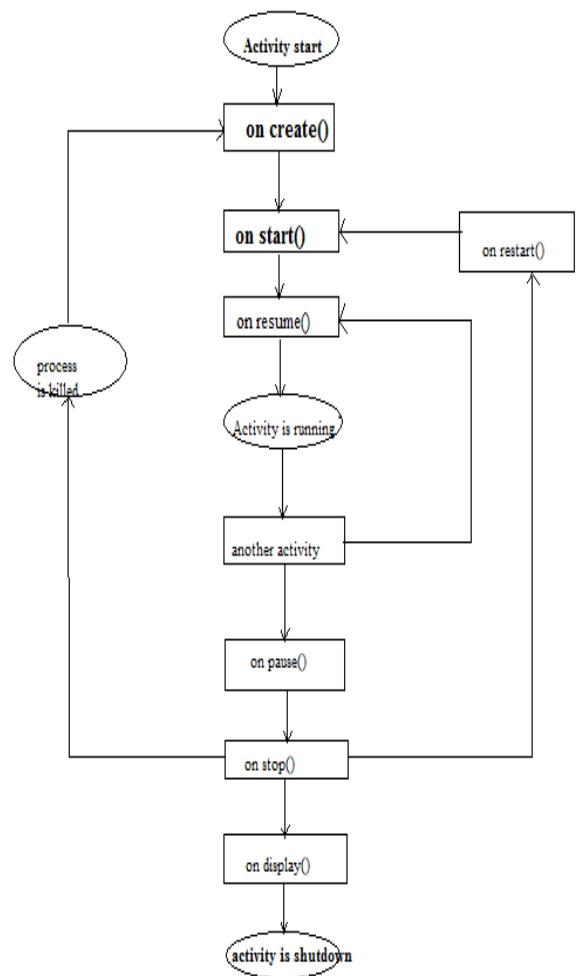


Fig 6: Basic Application Development flow chart

5.2 Sampling Coding Of The Android Application

The sampling code of the android application as given below. The Android main activity control code is @SuppressWarnings("WifiManagerLeak")

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    fan = (ImageView) findViewById(R.id.im_fan);
    fan.setOnClickListener(this);
    motor = (ImageView) findViewById(R.id.im_motor);
    motor.setOnClickListener(this);
    light = (ImageView) findViewById(R.id.im_light);
    light.setOnClickListener(this);
    changeStatus = (AppCompatActivity)
    findViewById(R.id.changeStatus);
    changeStatus.setOnClickListener(this);

    //reference to the text view
    status = (AppCompatActivity)
  
```

```
findViewById(R.id.status);
```

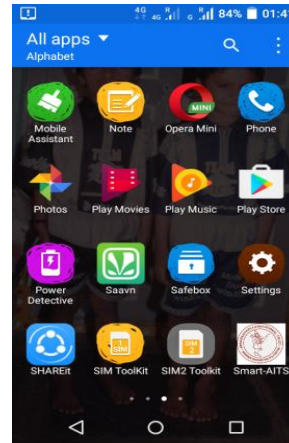
```

//reference to the bluetooth adapter
mBluetoothAdapter =
BluetoothAdapter.getDefaultAdapter();
//check if adapter is available, please note if you running
//this application in emulator currently there is no
support for bluetooth
if(mBluetoothAdapter == null){
    status.setText("BlueTooth adapter not found");
    changeStatus.setText("BlueTooth Disabled");
    changeStatus.setEnabled(false);
}
//check the status and set the button text accordingly
else {
    if (mBluetoothAdapter.isEnabled()) {
ON");
        changeStatus.setText("Switch OFF Bluetooth");
    }else{
        status.setText("BlueTooth is currently switched
OFF");
        changeStatus.setText("Switch ON Bluetooth");
    }
}
toggleButton = (ToggleButton)
findViewById(R.id.toggleButton);

textView = (AppCompatActivity)
findViewById(R.id.text_view);

toggleButton.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener() {
@Override
    public void onCheckedChanged(CompoundButton
compoundButton, boolean checked) {
        if (checked) {
            textView.setText("WiFi is ON");
            WifiManager wifi = (WifiManager)
getSystemService(Context.WIFI_SERVICE);
            wifi.setWifiEnabled(true);
        } else {
            textView.setText("WiFi is OFF");
            WifiManager wifi = (WifiManager)
getSystemService(Context.WIFI_SERVICE);
            wifi.setWifiEnabled(false);
        }
    }
});
}
}

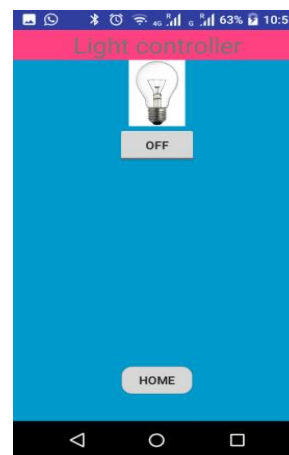
```



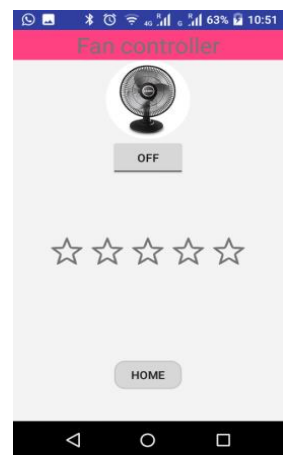
(a)



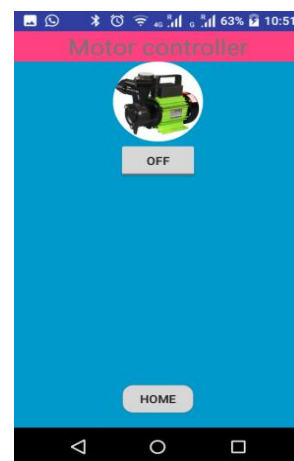
(b)



(c)



(d)



(e)

Fig 7: Android Application Implementation for Smart home Appliance (a) Android Application icon (b) Main Activity for Android Application (c) Light Control Activity (d) Fan Control Activity (e) Motor Control Activity.

6. RESULTS

We have evaluated the proposed design by performing demo test. Results are proven to work properly. Fig 8 (a) & (b) shows that Light, Fan controlled , power switch applications are proven to work properly. The power delivered to the Light can be switched “ON” and “OFF” as we need it.

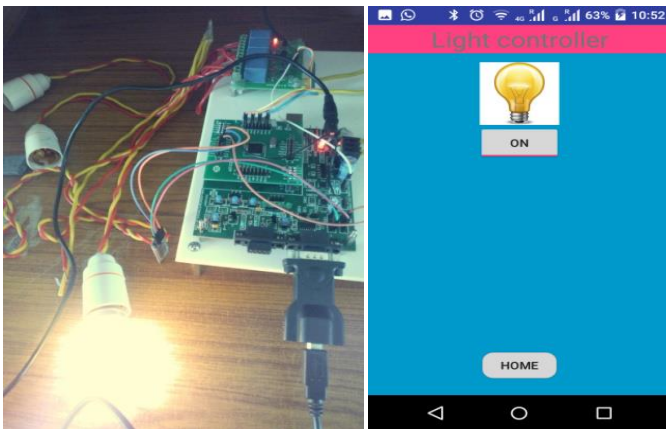


Fig 8(a): Lamp(light) controlled (ON State)

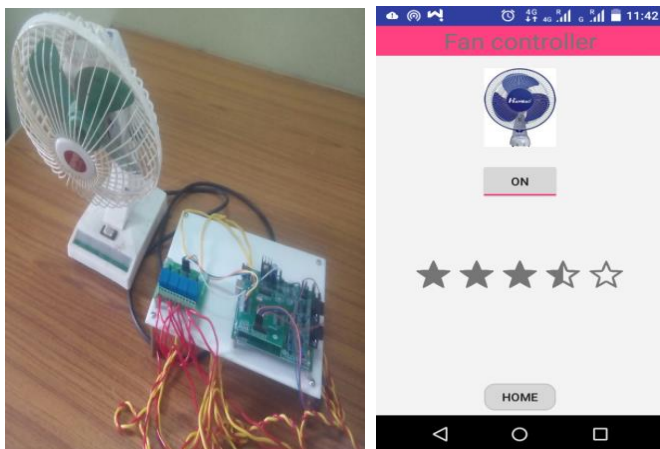


Fig 8(b): Fan controlled (ON State)

7. CONCLUSION AND FUTURESCOPE

Finally we concluded that our method works effectively on smart home appliances by using android application controlled through Bluetooth and Wi-Fi technologies.

The implementation of the proposed control system is currently limited to IR sensors. More state dependent devices must be identified. Therefore, to control devices with a more precise pointing mechanism, and support an auto discovery mechanism of state dependencies are two possible directions for future research.

ACKNOWLEDGEMENT

I thanks to my project guide, Mr. P. Sukumar for providing necessary facilities towards carrying out this work. I am also thankful to Mr. S. NarayanaRaju, Embedded systems design Specialist, for all the support successfully completion of this project. I am also very much thankful to my project co-ordinator and entire ECE department faculty for giving the freedom to choose this project, for continuous support and encouragement. The authors can acknowledge any person/authorities in this section. This is not mandatory.

REFERENCES

- [1] M.javier et al., “From the internet of things to the internet of people”, IEEE Int.Comput., vol.19,no.2,pp. 40-47, Mar.2015.
- [2] Wang, L.Hu, J.Zhao, “A survey from the perspective of evolutionary process in the Internet of Things,” Int. J.Distrib. Sens. Netw., vol.2015, no.9, pp.1-9, jan.2015
- [3] <https://developer.android.com/studio/index.html> and <https://developer.android.com/training/basics/firststep/index.html>
- [4] <https://developer.android.com/training/index.html> and https://www.tutorialspoint.com/android/android_studio.htm
- [5] https://www.youtube.com/watch?v=ZddHwDYqyIw&list=PL_PaSTBkUwk4PXlhZle4COpwnWeCRnw2F

BIOGRAPHIES

Mr.K.NARENDRA REDDY, studying (M.Tech) in Embedded Systems, at AITS, Rajampet. INDIA. Currently he is undergoing internship training at EmWare Technologies (INDIA) pvt Ltd. At Bangalore center. He is expected in embedded systems peripheral drivers and 8/16/32 architectures etc.

Sri. Mr. P.Sukumar, professor – He is having vast Experience of industry as well as Academics. He presented several papers in International and National Conferences and journals.