International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-04, Issue-04, July 2018

An ARM Based Smart Home System Using ESP8266

Raihanath P.P, M-Tech ECE Dept., Mar Athanasius College of Engg., Kothamangalam, India, raihaluckon9@gmail.com

Haripriya P, Assistant Professor ECE Department, Mar Athanasius College of Engg., Kothamangalam, India, hpriya.manappattu@gmail.com

Abstract: The project Smart Home system, opening a smart controlling and monitoring way into a home/building's electrical system and android application based remote controlling of home appliances through IoT. The system including LPC1769 and ESP-01 using an android application for controlling it. The connectivity between the server and the mobile device is provided by standard Wireless Access Point. The main target of the project is to remote controlling of home appliances over internet under the basic principles of Internet of Things-IoT through android application. For this, we use an Android app developed in android studio with user configurable front end (GUI). Internet of things is a developing network for controlling the action and update the status of device for the busy life cycle of people. It is meant to save the electric power and human energy.

Keywords: ESP8266-01(Wi-Fi module),Android application, ARM LPC 1769, Printed circuit board(PCB).

I. INTRODUCTION

Today's life depends smart home system incorporating IoT more and more. Smart home systems provide increased comfort especially when settled in a private home. Automation systems installed in commercial buildings increase comfort, and also allow centralized control of air condition, lightning and ventilation. Hence, they facilitate to an overall cost reduction and also to energy saving make the life of people much better.

Smart home system using Wi-Fi module and android application reduce the installation cost and make life of people more comfortable[1]. The Wireless Fidelity (Wi-Fi) technology provides an excellent medium through which multiple devices can be connected to one network. Wi-Fi operates over an internationally approved frequency band of 2.4GHz. Home appliances are controlled from anywhere by using IoT technology[2]. A channel is created in Thing speak server for IoT purpose. Android application which includes the layout of each room is controlled devices which are connected to the relay module[3].

This paper mainly includes three steps which are AT command sending, Wi-Fi connection initialization and controlling by android application. Relay module connected With LPC1769 are worked based on the instruction sending from android application. Four devices are connected to the LPC1769 through a 4 channel relay module[4]. For the regulation of speed of fan and brightness of light dimmer circuits are used.

Here the light and fan are controlled through smart app developed in android studio. So based on the requirement of home android app can be changed by adding new buttons or slider. Dimmer circuits are used for speed controlling of fan and for this a slider is added in smart application.

LPC1769 PCB board is designed with 5V regulator for obtaining constant 5V supply to the relay. ESP01 is initialized by AT command sending from arm board.ESP01 making connection with mobile hotspot through Wi-Fi module initialization.ESP01 can set as a server or a station point. The data received at the Gateway is sent through the UART serial port to the Wi-Fi module. The Wi-Fi module sends data to the server where it is stored and can be retrieved for future use[5][6]. The user is provided with an android application based Graphic User Interface (GUI) to exercise the desired control over the lights, fan speed regulation, control of appliances[7].

II. METHODOLOGY

We have to design a smart home system for controlling home appliances by using android application through IoT. The status of the device is controlled by android application developed in android studio. AT commands are needed to communication of LPC1769 with Wi-Fi module. The input character is transferred to ThingSpeak database and when this input matches with character loaded to arm relay module works properly. Then it is feed to the ARM microcontroller.

 DOI : 10.18231/2454-9150.2018.0465

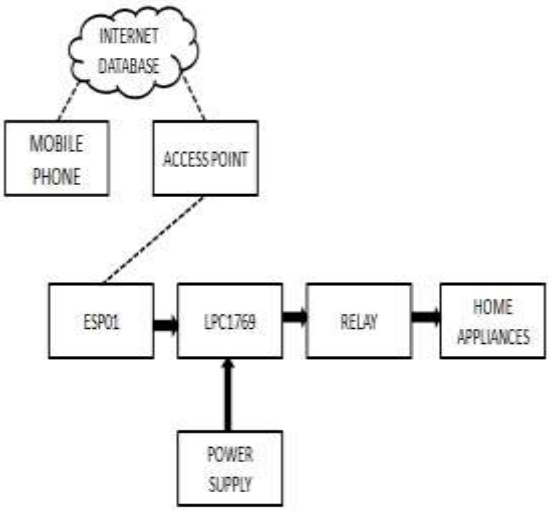

Fig. 1.Block diagram of proposed system

The concept of IoT is closely related with the popularization of smart home system. By employing the network structure of IoT and utilizing standard IoT protocols, the household appliances can be monitored and controlled remotely using android application through internet. A home that is equipped with such a wireless system can be called Smart Home in context of IoT. The Smart Home concept basically provide many new features to a regular home like security, remote access and flexibility of expansion.

User can use either a Smartphone Application or a Web Browser to access the UI of Home Automation. The user is presented with a graphical menu containing all the actuators
available in the system. For example, for a particular room, the menu shows the switches for Ceiling Light, Fan, Plugs. When user touches a switch the corresponding item will turn On/Off[8].

III. Software And Coding

Mainly three software is used to complete the entire project work.

3.1 Android Studio

Android Studio is used to develop an android application with user interface options. Application named smart which including layout of each room with corresponding buttons for each device. The visual part of android application is UI/UX which is written in the XML file. The application programming interface(API)of android application is written from java.

By using this Android studio software can change the status of the relay connected to the LPC1769.Java code is used to assign working of each button by programming in backend. The front end of the application is including the necessary buttons for controlling appliances.

Creating android application using android studio

In this project our android application is controlling home devices like light, fan from anywhere using IoT. For the easy understanding of which device of which room is controlled layout of each room with buttons for corresponding device is assigned.

In order to easily develop the android application first we design the user interface part and then programming the backend.

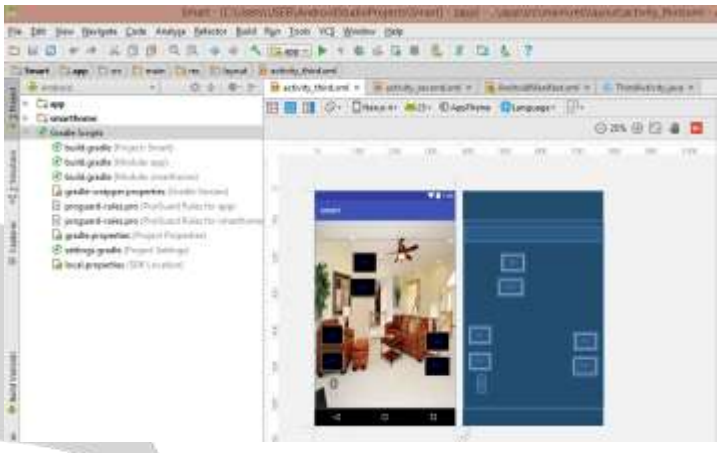

Fig.2.Making of android application in android studio

In Fig. 2the working area of android studio is shown with activities with drag and drop user interface. The button is selected using cursor and then change the buttonID and its properties. Fig 3 shows the simulation of android application on aandroid virtual device

Android Emulator

- An android virtual device (AVD) that represents a specific android device. The Android SDK includes a mobile device emulator. The emulator can develop and test Android applications without using a physical device.

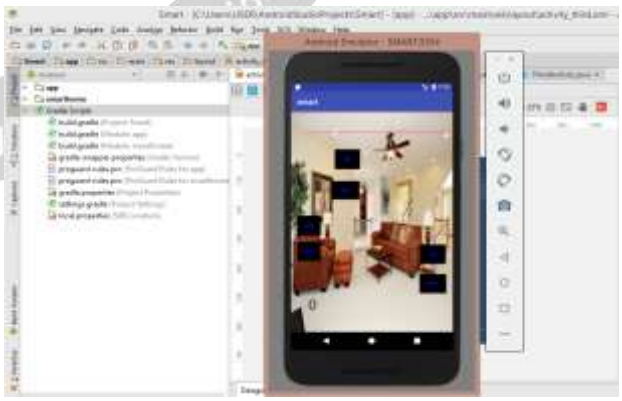

Fig.3.Simulating android application on android virtual device

. **AVD Manager**

- It provides a GUI in which we can create and manage Android Virtual Devices (AVDs), which are required by the Android Emulator.

3.2 ThingSpeak

 DOI : 10.18231/2454-9150.2018.0465

ThingSpeak is an open source Internet of Things (IoT) application and API to store and retrieve data from things using the HTTP protocol or via a Local Area Network. .To read and write to a ThingSpeak channel, the application sends requests to the ThingSpeak server, using TCP connection. Each ThingSpeak channel can have up to eight fields of data, in either numeric or alphanumeric format. Use the ThingSpeak API to process numeric data, which includes averaging, rounding, median and summing. By using HTTP POST You can create and update a ThingSpeak channel by posting a feed with your API key and data. For integration with applications ThingSpeak channel feeds supports JSON, XML, and CSV format.

The Fig. 4 shows the specific fields is creating on a channel in ThingSpeak.

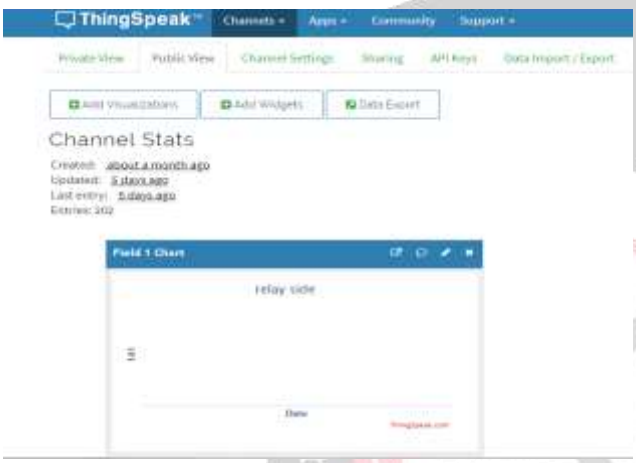

Fig.4.ThingSpeak channel

Mainly ThingSpeak channel is giving characters with asterisk to LPC1769 board.ARM board getting the access of instruction from ESP01 through hotspot connectivity.LPC1769 checks the character and asterisk , if they match relay can control from anywhere through IoT.

3.3 LPCXpresso 7.3.0

LPCXpresso is a low-cost microcontroller (MCU) development platform ecosystem from NXP, which provides an end-to-end solution to the engineers to develop embedded applications for final production.

The LPCXpresso IDE, a software development environment for creating applications for NXP's ARM-based "LPC" range of MCUs. In this project using LPC1769 platform.The LPCXpresso IDE is based on the Eclipse IDE and features many ease-of-use and MCU specific enhancements. The LPCXpresso IDE also includes the ARM GNU toolchain, providing professional quality development tools at low cost.

A Workspace is simply a directory that is used to store the currently working project. Each Workspace can contain multiple projects, and single computer can have multiple workspace. The LPCXpresso IDE can only access a single Workspace at a time, although it is possible to parallely run multiple instances and each instance can access different Workspace.

IV. EMBEDDED SOFTWARE

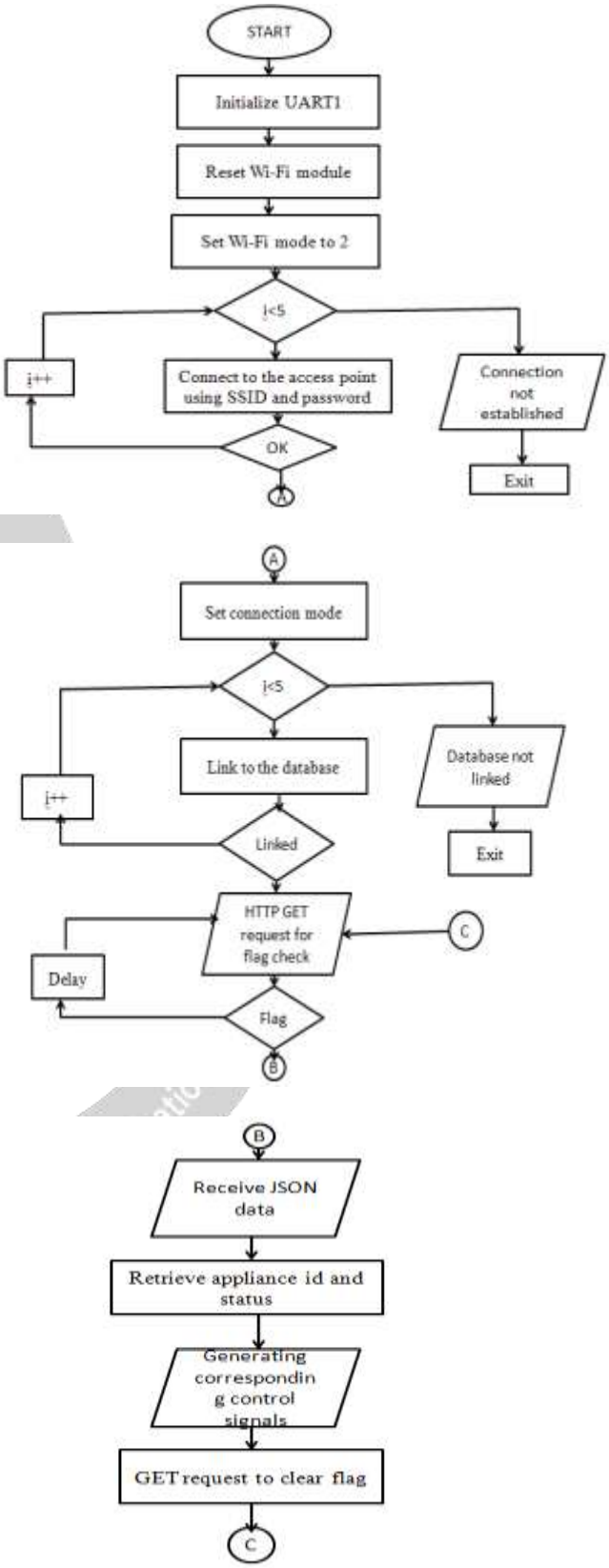

Fig.5. Flow chart of embedded web server software

Fig 5 shows the flowchart for the embedded web server software. The required UARTs are initialized. Then the baud rate is set. The Wi-Fi module is set up for its

 DOI : 10.18231/2454-9150.2018.0465

operation by sending AT commands. Once the Wi-Fi module establishes connection with the internet using HTTP request methods command is send to link with the URL of our web server. Once the server gets linked the code will jump into a continuous loop for checking the update flag. If the flag is clear it will wait for a delay and continue the process. If the flag is set, it will retrieve the appliance status from the server, make corresponding updates to the actuator ports. Then command is send to clear the flag and the flag checking loop is entered again. This is simply shown in the below figure 6.

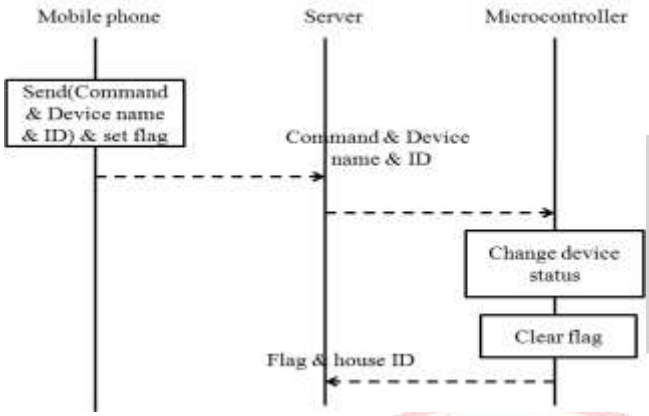

Fig.6.Process of family services controlling

V. HARDWARE IMPLEMENTATION

The UART, universal asynchronous receiver / transmitter, is mainly used for the communication between ESP01 and LPC1769. At the sender side, UART receive the data as bytes and then UART transmits them as sequential individual bits. At the receiver side, these sequential bits are assembled into bytes and stored into the receiver register until it is reading.

The basic communications that take place in the system is as shown in figure 7. The microcontroller will continuously interact with the web server over a fixed interval of time and waiting for any updates. If a user logs in to the server via android application he can access the any appliances which he require to control and change its status. When a user update the power state of an appliance, a flag will be set for the corresponding house in the database. The flag will be continuously checking by the microcontroller. When this flag status is set, the microcontroller will receive the power status of appliances. This power status is received in JSON format. This JSON encoded data is decoded and tabulate the power status for each appliance. Then changes to the actuators can be done by sending control signals to the output port.

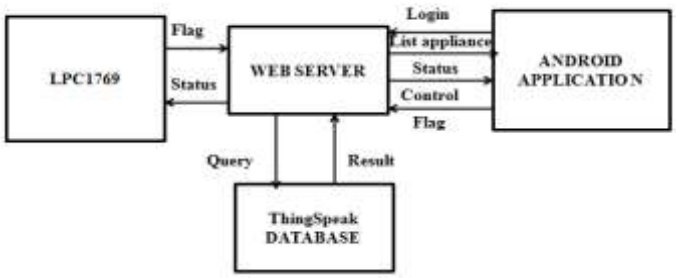

Fig .7.Basic communications between the different modules

Fig 8 shows the hardware implementation of the system. The LPC1769 will interact with the ThingSpeak server continuously over a duration of time and check for any updates. If a user logs in to the server via android application he can access the appliances under his ownership and change its status. Once the power state of an appliance is updated by the user, a flag will be set for the corresponding house in the database.

Fig.8. Controlling bulb and phone using Smart app

The LPC1769 will be repeatedly checking the status of flag. When it reads the flag status as set, the microcontroller will retrieve the power status of appliances. This will be received in JSON format. The embedded software will decode the JSON encoded data and tabulate the power status for each appliance. Then it will send control signal as characters to the output ports to make the required changes to the relay module.

VI. EXPERIMENT RESULTS

Wi-Fi based smart home system is implemented using LPC1769 with ESP-01 and relay controlling through android application. The set-up consist of an arm board with ESP-01 and 3 channel relay module. The status of the device connected to the relay is monitored and controlled by android application developed in android studio.

Figure 9 shows the experimental arrangement used for controlling home appliances through IoT using android application. It consists of FT232 module for serial communication with PC. GUI is developed using android studio.

 DOI : 10.18231/2454-9150.2018.0465

Fig. 9. Smart home system

5.1 Relay controlling through android app

An android application is needed to control the home appliances which are connected to LPC1769 through relay module. Developed smart app can control the on and off of devices through ThingSpeak server.

(a)

(b)

Fig. 10. Android application UI view(a)Login screen(b)Room layout

Fig 10 shows the login screen and room layout of android application.The Android application opens to a login screen consisting of fields for username and password and a login button. On pressing the login button the data entered in fields will be send to the URL for web server. If the username and password matches with the data in the database, access will be provided and the user will be logged in. Once the user logged in, it will retrieve the rooms of house owned by the user from the database and display it. The user can select the room by clicking on it.Power on and off buttons are can be used to switch on and off the appliances and also for changing the device states.

VII. Conclusion

While the initial objective of the home intelligence was to automatically control devices and systems according to specific rules previously implemented, here, an algorithm is used to enable an interactive home environment capable of adapting its operations to accommodate the users. This paper proposes a secure, low cost and internet of things controlled solution. The method discussed in the paper is achieved the target to control home appliances over the internet through the android to satisfying smart home requirements. Android application controlled Wi-Fi capable solution has proved to be cost-effective and controlled remotely through IoT compared to the previous systems.

REFERENCES

[1] Silviu Folea ,Daniela Bordencea , Casiana Hotea , Honoriu Valean, "Smart Home Automation System using Wi-Fi low-power devices", *IEEE International Conference on Automation Quality and TestingRobotics(AQTR), pp. 569-574, 24-27 May 2012*

[2] S. Hong, D. Kim, M. Ha, S. Bae, S. Park, W. Jung and J. E. Kim,"SNAIL: An IP-based wireless sensor network approach to the internet of things", *IEEE Wireless Commun., vol. 17, no. 6, pp. 34-42, 2010.*

[3] R.A.Ramlee ,M. A. Othman ; M. H. Leong ; M. M. Ismail ; S. S. S. Ranjit,"Smart Home using Android Application",*ICoICTIEEEConferenence,Bandung,Malaysia, pp. 277-280, 20-22 Mar. 2013.*

[4] Bader M. O. Al-thobaiti, Iman I. M. Abosolaiman, Mahdi H. M.Alzaharani, Sami H. . Almalki, Mohamed S. Soliman, "Design and Implementation of a Reliable Wireless Real-Time Home AutomationSystem Based on Single-Board Microcontroller",International Journal of Control, Automation and Systems, Vol. 3(3), pp.11 - 15,2014.

[5] Ahmed ElShafee, Karim Alaa Hamed (2012) " *Design and Implementation of a WiFi Based HomeAutomation System*", World Academy of Science, Engineering and Technology

[6] Wireless Networking Handbook : Jim Geier

[7] Android Development with Android Studio or Eclipse ADT - Tutorial: www.vogella.com

[8] Rajeev Piyare1 and Seong Ro Lee1 (2013)," *Smart Home-Control and Monitoring System Using Smart Phone*", Mokpo National University, ICCA 2013, ASTL Vol. 24, pp. 83 – 86

 DOI : 10.18231/2454-9150.2018.0465