Scientific Journal Impact Factor (SJIF): 1.711

e-ISSN: 2349-9745
p-ISSN: 2393-8161

International Journal of Modern Trends in Engineering and Research
www.ijmter.com

Embedded Web Server based Home Automation using Raspberry PI

Girish Birajdar[1] ,Shrikant Mahindrakar[2]
[1]Electronics Department, MCE Nilanga,
[2]Computer Department, MCE Nilanga,

Abstract — In smart home's we have various high-tech appliances to get our jobs done and make the life easier. It is necessary to control these home appliances smartly from anywhere. In this paper we are implementing a system that will enable house owner to control their home through the internet with high mobility and security. We will use the Embedded Web Server (EWS) which enables controlling and monitoring the home appliances remotely with the help of any standard web browser.

Keywords – Embedded Web Server, Raspberry PI, Apache, WebIOPi, Smart home etc.

I. INTRODUCTION

Smart home is a home in which electronic devices such as light, fan, air conditioner, refrigerator, Television etc can be controlled smartly. Smart home will become more smarter if the home appliances can be controlled from any remote place. In this paper our aim is to find solution for this. The motivation behind the goal is very simple, always it is not possible to be near to the home physically but it's very important to control the appliances for many purposes. So the remote controlling takes the control of the home.

The numbers of media are available for remote communication but internet is good as it does not have any limit on geographical placement and provides high mobility to user. On the other hand better security can be achieved by authenticating the user. In this research we will embed web access functionality in a low cost, small sized device called as Embedded Web Server (EWS) which can be widely accessible from remote places. In turn Embedded Web Server will control the ON/OFF state of relay switch using relay driver.

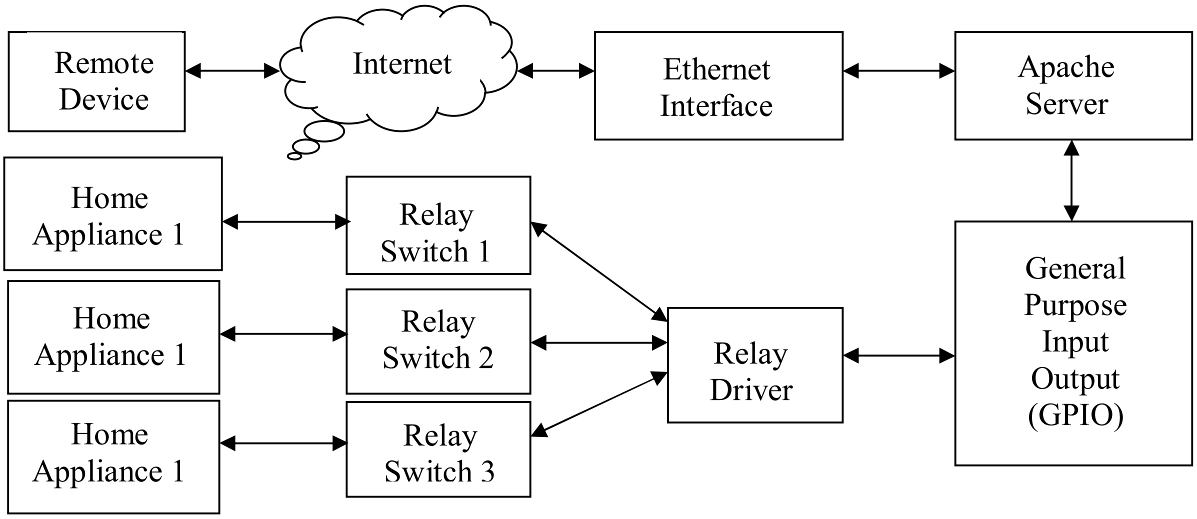


Fig. 1 The system block diagram

II. EMBEDDED WEB SERVER

The embedded system which can serve the web documents on request by a client from other system. Such type of a web server is called as Embedded Web Server (EWS). It mainly deals with management of dynamic contents and is fast, compact, simple to use.EWS design includes a complete web server with TCP/IP support, running different OS, memory, application wise. Several EWS based systems are designed for automation and monitoring purpose. Data transmission system is presented using with ARM processor that contains internet software which suite for monitoring, controlling and remotely access the system.

III. HARDWARE DESCRIPTION

We will use different hardware resources to implement this system, which are described in this section.

1. Raspberry Pi: The Raspberry Pi is credit card size computer with low cost. The Raspberry Pi has microprocessor ARM1176JZF-S which is a member of ARM11 family. It has ARMv6 architecture. ARM11 processor operates at 700 MHz frequency & it has 512 MB RAM. It is build around a BCM2835 Broadcom processor. It consumes 5V electricity at 1A current due to which power consumption of raspberry pi is also less. It comes with many peripherals such as USB port, 10/100 Ethernet, GPIO, HDMI & composite video outputs and SD card slot.SD card slot is used to connect the SD card, which consist of raspberry Linux operating system.

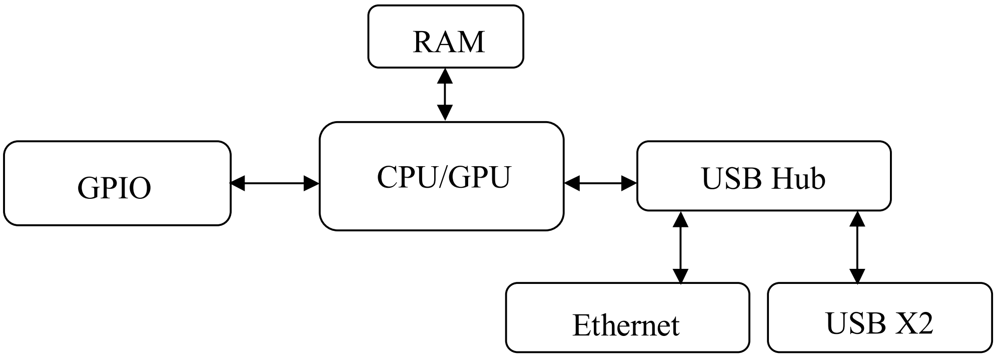


Fig.2 Block Diagram of Raspberry Pi

2. Ethernet: Ethernet is the computer networking technology which allows computers to communicate and share resources over the internet. Ethernet was standardized as IEEE 802.3. it is one of the most widely implemented LAN standard originally developed by Xerox. Different Ethernet networks also connect to a router that provides access to the internet.

3. Relay Driver: In this project we are using ULN2003 as a relay driver. It consists of seven NPN darlington transistor pairs array having high voltage and current capability. For each NPN darlington pair a cathode clamp diode is used, which makes this driver IC useful for switching inductive load's. It provides open collector output. We are using relay driver to drive the relay switch connected to it.

IV. SOFTWARE DESCRIPTION

Different software resources were used to implement the system, these are mentioned below:

1. Raspbian Wheezy: Raspbian wheezy is Linux kernel based operating system. It is debian optimized for the raspberry pi hardware. Raspbian wheezy uses the light weight ARMv6 instruction set in such a way that a typical Broadcom processor understand. It is open source operating system which can be downloaded and used free.

2. Apache2: Apache is most widely used web server which allows the online distribution of website service using Hyper Text Transfer Protocol (HTTP). It can be use on different operating systems such as Linux, Unix, Windows, Solaris, Mac OS X, Novell NetWare, OS/2 etc. Apache2 version was used in this project for creating web server. It is also known as "Apache HTTP Server".

3. My SQL: It is used to manage data related to web page and will runs on a web server hence it is also called as Database Management System. It support standard SQL and compiles on a number of platform such as Linux, Unix, Windows, Solaris, Mac OS X, Novell NetWare, OS/2 etc. My SQL is open source hence can be download and use free.

4. PHP5: The hypertext pre-processor (PHP) is a scripting language used on server side which is designed for web development. Web server with a PHP processor module integrates the PHP code which generates the resulting web page. It is basically used for developing web based software applications and also to manage database, dynamic content, session tracking, even build entire e-commerce sites. PHP5 version was used in this project.

5. WebIOPi: WebIOPi is browser based software which allows you to control, debug and use raspberry pi's GPIO locally or remotely, from a browser or any application. For the server it is written in python whereas JavaScript is used for client. It is open source to download. In this project latest WebIOPi version 0.0.6 is used.

V. IMPLEMENTATION

To implementation this project, firstly install all the software's on raspberry pi. Install the software's by following the below mentioned steps:

1. Install the operating system "Raspbian Wheezy". With the help of image write program, write image of operating system in SD card and then configure it while first boot by using command *sudo raspi-config*. Enable "ssh" during configuration, so that we can access raspberry pi remotely.

2. Before further installation update and upgrade the system by using commands *sudo apt-get update* and *sudo apt-get upgrade.* Now install apache2 by using command *sudo apt-get install apache2.* Apache server will run after the installation. We can start or stop the apache server by using the commands *service apache2 start* and *service apache2 stop.* After installation of apache server configure it correctly for raspberry pi by changing the host name to our chosen name for the server by using Linux text editor "nano".

3. Install Hypertext Pre-Processor PHP5 by executing the command *apt-get install php5 libapache2-mod-php5 php5-intl php5-mcrypt php5-curl php5-sqlite*

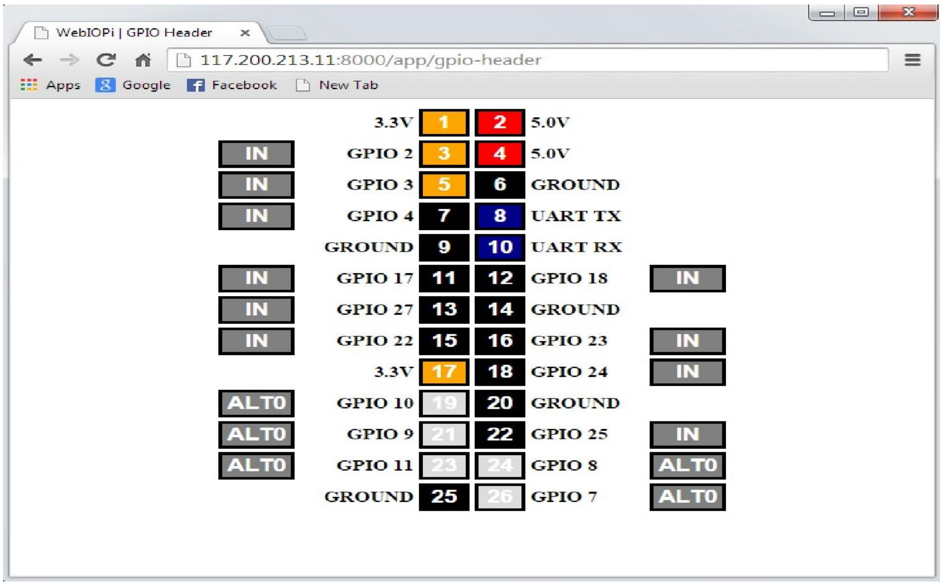
4. Install My SQL database management system by using command *apt-get install mysql-server mysql-client php5-mysql*

5. To install the WebIOPi firstly download it with 'wget' command or directly from browser. Extract tarball file using command *tar zxvf webiopi-0.6.0 tar.gz* now go to the extracted directory and install the WebIOPi using command *sudo./setup.sh* to start WebIOPi run the command *sudo /etc/init.d/webiopi start.*

After installing all powerful software's to raspberry pi, connect all the GPIO pin's to the inputs of a relay driver IC ULN2003 and relay switches at the outputs of the IC. Now you can able to control relay switches. By using relays we will be able to switch light, fan, Air Conditioner, refrigerator, Television etc ON and OFF.

VI. RESULT AND DISCUSSION

After the final setup of the project, it is time to test the system. User has to open browser window of remote device and access the WebIOPi by using IP address of Raspberry pi. Open the link *http://raspi-ip-address:8080/webiopi*. It will ask for a login and password, which you can change later. But the default login and password is *webiopi* and *raspberry,* respectively. User will see the WebIOPi menu. Now go to the GPIO header link, the GPIO header pins along with the pin function will appear, as shown in below figure:



	3.3V	1	2	5.0V	
IN	GPIO 2	3	4	5.0V	
IN	GPIO 3	5	6	GROUND	
IN	GPIO 4	7	8	UART TX	
	GROUND	9	10	UART RX	
IN	GPIO 17	11	12	GPIO 18	IN
IN	GPIO 27	13	14	GROUND	
IN	GPIO 22	15	16	GPIO 23	IN
	3.3V	17	18	GPIO 24	IN
ALT0	GPIO 10	19	20	GROUND	
ALT0	GPIO 9	21	22	GPIO 25	IN
ALT0	GPIO 11	23	24	GPIO 8	ALT0
	GROUND	25	26	GPIO 7	ALT0

Fig. 3 Accessing GPIO pins using WebIOPi in remote browser

The 'IN' means the corresponding GPIO pin can be used as input. Double click on 'IN' box; the caption will change to 'OUT'. Now, the pin can be used as output pin. Make all the eight GPIO pins as OUT. Click on each GPIO pin, such as pin 16 (GPIO 23) to begin with, on the browser screen; it will change from black to orange and the corresponding home appliance will get switched ON. To turn OFF, click on the same pin, where it will return to black from orange.

VII. CONCLUSION

This paper introduces an efficient, internet based system to control everyday home appliances remotely. It provides high mobility as well as security to user. The system implementation cost is very low, small in size; also power consumption is low which makes it different than existing systems.

REFERENCES

[1] Nakul Padhye and Preet Jain, Implementation of Arm Embedded Web Server for DAS using Raspberry pi, Vol. 3 No. 4, IJEECE, 4 April 2013.

[2] Abhishek Vichare and Shilpa Verma, Embedded Web Server for Home Appliances.IJERA, National Conference on Engineering Trends in Engineering & Technology, 30 March 2012.

[3] Karia D.C., Adajania V., Agrawal M. and Dandekar S., Embedded Web Server application based automation and monitoring system, International Conference on Signal processing, Communication, Computing and Networking Technologies, July 2011.

[4] Gowthami T. and Adiline Macriga G., Smart Home Monitoring and Controlling System uging Android Phone, IJETAE, Vol. 3 No. 11, Nov 2013.