

# Embedded Based Web Server for CMS and Automation System

WebLOPI | Light Control x Blog Pacific - Sharing All x MJPG-streamer x  
← → ↻ 169.254.132.67:8000

## Embedded Based Web Server for CMS and Automation System

Light 1  
Light 2  
Light 3

Guided by-  
Prof. Prabhat Panday

Submitted by-  
Prashant SEDIWAJ

Shut Down

11:40 PM  
19/07/2014

## **ABSTRACT**

This research deals with designing a “Embedded Based Web Server for CMS and Automation System” in which, we will host any website (wordpress, joomla, drupal) from our own server. So because of this web server we have not to pay a hosting charge to any server provider and serve your website to internet without paying any rent charge for hosting. And also to explain more utility of this project we will control the appliances through the web browser running on the ARM11 Controller by this same hardware. We can also remotely control the appliances by opening in the I.P address from any system that’s connected on the same network and that can support Browser natively. For showing this utility i design three load which operate on 220 volt. I control the bulb switching through my website.

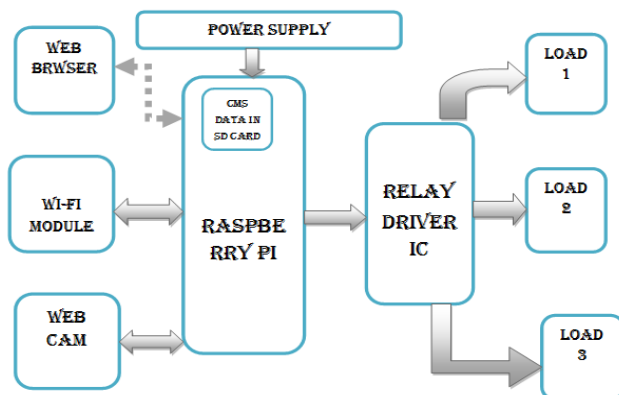
In this project not only to control the home or industries appliances, we can also connect temperature sensor to the raspberry pi board to check temperature of home. As raspberry pi board doesn’t have in-built ADC we require to connect additional ADC Ic.

## 1. Introduction

The project aims in designing a system, which host our website and also support CMS (content management system like wordpress, joomla, drupal) from our web server that provides faster execution and better security, and also we use this for controls the Electrical appliances in Home or industries through a browser remotely from laptop or Android phone using Wi-Fi module. We will control the appliances through the web browser running on the ARM11 Controller. We can also remotely control the appliances by opening in the I.P address from any system that's connected on the same network and that can support Browser natively.

The main blocks of the whole system is **raspberry pi** development board. Through its GPIO pins we are connecting the appliances. We are connecting a relay board to act as a switch between the pi board and power supply. To control the devices we require of web browser and Wi-Fi module, through Wi-Fi module we are connecting pi board to the web browser. So from web browser we are controlling the home appliances. To cross verify the system we are using a webcam to watch the live streaming of home in web browser.

## 2. Block Diagram



**Figure 1 block diagram for embedded based web server and automation system**

## 3. Installation of wordpressCMS on our sever

### A Quick Overview

Here's a basic summary of what we're going to do. There's a lot here, but it should all be reasonably quick and painless if you follow it in sequence:

### Preparation

- Burning **Raspbianos** to an SD card
- SSH configuration & some security recommendations

### Apache and PHP

Install & configure Apache, the web server  
Install PHP, the server side scripting language

### Database Installation & Configuration

Install some Database software & create a root username and password Install phpMyAdmin for GUI database management

### WordPress

- Download the latest version of WordPress as a .tar.gz file from wordpress.org and decompress it.
- Create a username and database for WordPress
- Create the WordPress configuration file (wp-config.php) and fill it in with details of the WordPress database.
- Move the WordPress files to the correct location in Apache's data directory
- Run the WordPress installation script.

### WAN access

- Set up DNS records to point your domain name at your WAN IP address
- Set up port forwarding on your router to allow users to access the site from outside your LAN

### 3.1 Installation Of Apache 2

In our RaspbianOs terminal by giving some command we can install Apache2 (the web server). Internet connection is required.

```
$sudo apt-get update
```

```
$sudo apt-get install apache2
```

This will also install a few dependencies.

### 3.2 Installation Of PHP

PHP is a server side scripting language that is used by many applications, including WordPress. It does all of the interactive parts of your site: comment boxes, logins, etc.

Here's how to install it:

```
$sudo apt-get install libapache2-mod-php5 php5
```

Now you can install a package that will allow PHP scripts to connect to databases:

```
$sudo apt-get install php5-mysql
```

PHP Cache used for faster page loading times. When you request a PHP page on the server, the server has to compile the page from the PHP source before it sends the data you requested.

Because your Pi is such a low powered machine, this can be really slow.

Luckily, there's a PHP cache you can use that will store a pre-compiled copy of pages in your RAM after they have been visited already. This makes a significant improvement to page loading times.

Install it by using this command:

```
$sudo apt-get install php-apc
```

### 3.3 Installation of Database Software:- MySQL

WordPress uses a database to store your site's content, as well as other information like usernames and passwords. We need to install some database software to allow this to work.

Install mysql server and client:

```
$sudo apt-get install mysql-server mysql-client
```

The package installation will bring up a configuration wizard, which will ask you to set a root username and password. Enter the username you'd like to use, and a password. Write these down!

GUI database management with phpMyAdmin

Now that we have some database software installed, let's install a program called phpMyAdmin, which allows you to manage databases from within a web browser instead of the commandline.

```
$sudo apt-get install phpmyadmin
```

You'll be asked to confirm which web server you have installed... choose Apache2.

### 3.4 Installation of WordPress

SSH into your Pi, and then make sure you're in your home folder:

```
$cd ~
```

Download & Extract

Now download the latest version of WordPress from [wordpress.org](http://wordpress.org):

```
$wget http://wordpress.org/latest.tar.gz
```

The file you just downloaded is a compressed archive. Unpack it with this command:

```
$tar -xzf latest.tar.gz
```

This should result in the creation of a folder in your home directory called **wordpress**. Change directory into it and list the files to get a feel for what's inside:

```
$cd wordpress
```

```
$ls -l
```

Now from within the wordpress folder use this command to copy the entire contents to Apache's data directory (/var/www/):

```
$sudo cp -r * /var/www/
```

#### 4. Step to develop Automation system on same hardware (raspberrypi)

For demonstration of an example of automation system we take 3 bulb which are on and off through our web page.

For this we require a WEBIOPI frame work, HTML page, GPIO pin configuration, python code for operate on, off function through webpage

##### 4.1 Installiom of WEBIOPI frame work

*WebIOPI is developed and tested on Raspbian.* You only need Python, either 2.7 or 3.2. Download, then extract and install WebIOPI. The setup script will automatically download and install required dependencies using apt-get. You may have to manually install GCC and Python development headers if you are *not* using Raspbian.

```
$ tar xvzfWebIOPI-x.y.z.tar.gz
```

```
$ cdWebIOPI-x.y.z
```

```
$ $sudo./setup.sh
```

The above commands should be entered into the SSH shell to unzip the webiopi file and setup the file using above command.

##### Running WebIOPI

Finally, use webiopi command:

```
$$sudowebiopi[-h][--config][--log][--script][--d][port]
```

##### Options:

-h,--help	Displaythis help
-c,--config file	Loadconfigfrom file
-l,--log file	Log to file
-s,--script file	Load script from file
-d,--debug	Enable DEBUG

##### Arguments:

portPort to bind the HTTP Server

For instance, to start with verbose output and the default configfile :

```
$ $sudowebiopi-d -c /etc/webiopi/config
```

WEBIOPI is now ready.

##### Usage

If your are directly using your Raspberry Pi with keyboard/mouse/display plugged, open a browser to <http://localhost:8000/>

If your Raspberry Pi is connected to your network, you can open a browser to <http://raspberrypi:8000/> with any device of your network. *Replace raspberrypi by its IP.*

You can even add a port redirection on your router to use WebIOPI over Internet !

##### 4.2 Light Control with auto on/off

WebIOPI includes an HTTP server that provides both HTML resources and a REST API to control things. Your browser will first load a HTML file, then the included JavaScript will make Asynchronous calls to the REST API to control and update the UI. This method is very efficient, because it don't need to refresh and download the whole page.

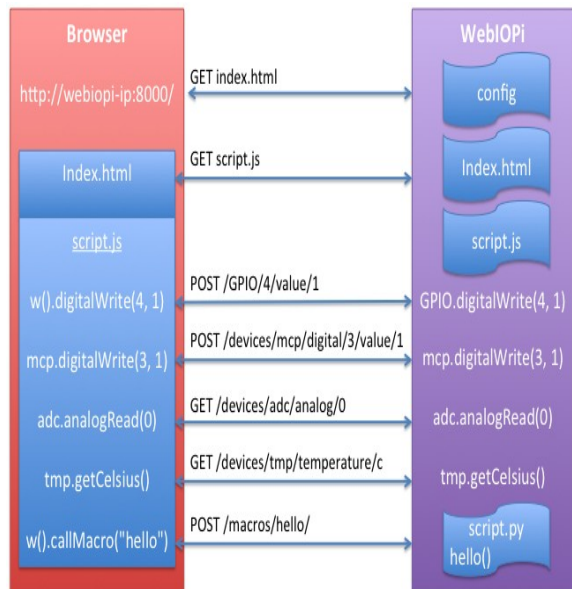


Figure 2 webiopi on/off controls

You can build your HTML / Web UI from scratch, using the WebIOPi JS library or not.

You can extend the WebIOPi behaviour by loading custom Python script using an Arduino like syntax with setup/loop functions

Assume we simply want a button to control a light, which will be automatically turned on and off. I say a light, but it can be a anything you want.

The customization process has 3 steps :

Write a simple Python script to initialize the GPIO and handle auto on/off

Write a simple HTML/JavaScript page

Configure WebIOPi Server

This python script and HTML page coding you can download from my website site.

[http://blogpacific.com/wp-content/uploads/2014/08/source\\_code\\_homeautomation.zip](http://blogpacific.com/wp-content/uploads/2014/08/source_code_homeautomation.zip)

### 5. Result

- a. mywordpress website is hosted by my own web server. So no need to purchase a external hosting plans for serve a website to internet.

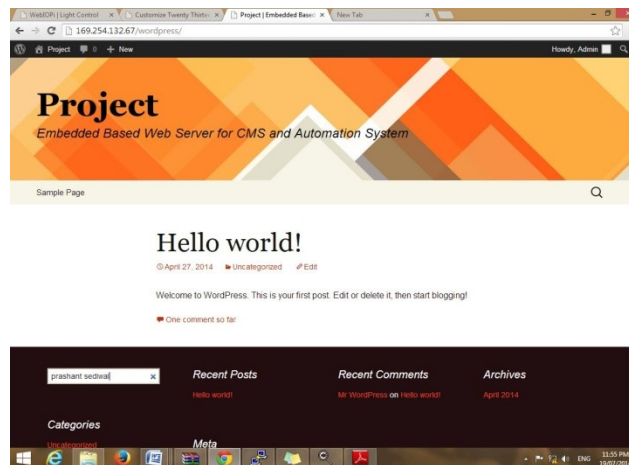


Figure 3 self hosted wordpre website

- b. Our home automation system is controlling bulb which are operate on 220v supplies by web browser. We can turn on and off this bulb by going to a unique URL and by click on light 1 or light 2 or light 3.

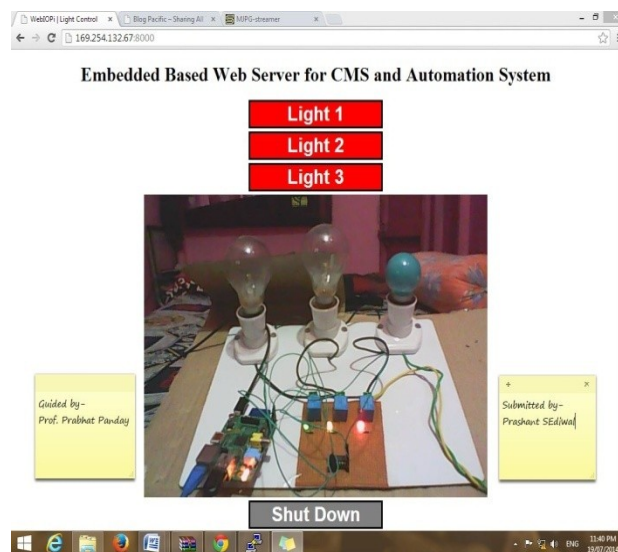


Figure 4 control home automation through web page

### 6. Application

- Through this web server we can serve many websites to internet.

- It's provides boost speed to websites execution because of ARM 11.
- Better security because all our confidential files are saved on our own server not to depend on others server.
- This web-server is best for organizations, institutions for share their data to internet.
- The cost of this webserver is equivalent to one year hosting plans for one website.
- This webserver support zig-zip conversion which compress webpage and server faster page load time upto 70%.
- This webserver support wordpress, joomla, and drupal and almost all CMS.
- This is multipurpose web server. We can use this as a hosting provider or as a automation syatem.
- We can control over all electrical and electronics devices from anywhere in the world. By browse the web and control it.
- We can also use this webserver as a CCTV.

## 7. Conclusion and future works

In this Project Report an Embedded Based Web Server for CMS and Automation System design is present through which remote devices can be controlled. The project presents system by which it is possible to controlling any electrical device remotely, via clicking on the server page devices can be switched on or off. One most important task is done by this project is that this project act as a website hosting platform. We can host various website either php based or any Content Management service based website.

The design is based on ARM11,Raspbian operating system, wordpress etc. The developed system is applicable in Industries, Research, and Educational Institution etc. The whole system has low-cost, good openness and portability, and is easy to maintain and upgrade.

The design will overcome the drawbacks of its limitation currently we can serve a website and control the electronic devices online, but we are not monitor and diagnose the website or a electronic device.

In Future my aim is to modify this system to enhance the extra services like monitoring the website and electronic device. By which we can monitor all the activity of our server and also diagnoses the error and

we can solved it easily. So that after this modification User can control, monitor, and diagnose the problem with the remote devices and website hosting.

## 8. References

1. Design and implementation of embedded Web server – IEEE2012 (ARM9)
2. Design of intelligent home appliance control system based on ARM and ZigBee - **IEEE2012** (ARM9/ARM11)
3. Design and Development of ARM Processor Based Web Server, International Journal of Recent Trends in Engineering, Vol. 1, No. 4, May 2009
4. Srinivas Raja, G. Srinivas Babu, "Design of Web based Remote Embedded Monitoring system" International Journal of Technology and Engineering system(IJTES),Jan-March 2011-Vol.2,No.2.
5. Guoling Liu, Xiaozhu Wang, He Jiang, RunianGeng , "Research on Embedded Remote Communication Mode", 978-1-4244-6349-7/10, IEEE 2010.
6. Zhan mei-qiong ,Jichang-peng "Research and Implementation of Embedded Web Server", International Conference on Multimedia and Information Technology, 2008.
7. Peng D.G, Zhang.H, Jiang.J.N. (2008) Design and Realization of Embedded Web Server Based on ARM and Linux. Mechatronics, Vol.14(10):37-40.

Authors-

<b>Name</b>	: Mr. Prashant sedawal
<b>Qualification</b>	: BE , M.Tech
<b>Area of Interest</b>	: Advanced embedded systems

mob : 9993024220,9700563692



**Name** : Mr. Prabhat Pandey  
**Qualification** : BE , ME  
**Designation** : Assistant Professor  
**Work Experience** : 3.6 years  
**Area of Interest** : Embeded System and VHDL  
**Publication** : National Journals- 02

