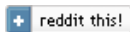


Raspberry Web Server

A Raspberry Pi web server cluster

Your message for visitors here [Learn more](#) [Close](#)

- [Home](#)
- [Linux basics](#)
- [Raspberry Pi Cluster](#)
- [HTML and CSS](#)
- [CGI scripting](#)
- [SQL Databases](#)
- [Server Administration](#)
- [GPIO](#)
- [Pyplate](#)



Like 450



+1 220

[Home](#) /

Home

A Raspberry Pi is a small computer that uses an ARM 11 processor running at 700MHz with 512MB RAM.

Setting up a Raspberry Pi as a home web server is a great way to learn about web design and server administration. A Raspberry Pi uses much less power than a PC, and takes up much less space. The fully featured Linux operating system comes with lots of software, including the Apache web server which is used to host some of the world's biggest web sites.



Latest articles

[Install Pyplate on a Raspberry Pi with Apache](#) Build a web site on a Raspberry Pi with Pyplate

[CSS - Cascading Style Sheets](#) This article is about using CSS with HTML.

[Making a complete HTML page](#) Coding a complete HTML page on a Raspberry Pi.

[HTML Basics](#) Learn how to write basic HTML on a Raspberry Pi.

[Bash basics](#) Learning how to use the Linux command line is extremely useful. Almost every aspect of the Linux operating system can be accessed from the command line.

[Comparing the performance of Nginx and Apache web servers](#) I've done a side-by-side comparison of Apache and Nginx to see which is the best choice for my web site.

[Setting up Nginx and uWSGI for CGI scripting](#) Nginx is an open source web server that's designed to handle heavy traffic efficiently and quickly. It's lightweight, so it's perfect for use with a Raspberry Pi.

Popular articles

[Install Pyplate on a Raspberry Pi with Apache](#) Build a web site on a Raspberry Pi with Pyplate

[Raspberry Pi Cluster](#) Find out how to build a cluster using Raspberry Pi computers. This site is hosted on a Raspberry Pi cluster, and contains tutorials on Linux, Python, Apache and web programming

[Linux Basics](#) Learn some of the basics of Linux

[Building a remote control vehicle using a Raspberry Pi](#) See how to control a a model tank with a Raspberry Pi. A simple example of web programming.

[Share your Raspberry Pi's files and folders across a network](#) Make your Raspberry Pi's Folders accessible on a network. Learn how to configure your Raspberry Pi

[Using MySQL on a Raspberry Pi](#) MySQL is a sophisticated database that scales easily for larger applications. This tutorial is an introduction to using MySQL on a Raspberry Pi.

[Building a web user interface for the temperature monitor](#) Learn how to build a web UI in Python. This tutorial shows how I built a web UI for the SQLite temperature logger on my Raspberry Pi.

[Connect your Raspberry Pi to a USB hard disk](#) Learn how to configure your Raspberry Pi. Connect a USB hard disk to your Raspberry Pi.

[Building an SQLite temperature logger](#) Learn how to build an SQLite based temperature logging system written in Python. This system uses a DS18B20 temperature sensor to take readings every 15 minutes, and it displays them on a web page.

[7 Comments](#)

Specs

A Raspberry Pi is a small single board computer with the kind of processor that you'd normally find in a smart phone. It's just about powerful enough to run a desktop version of Linux. It can be used as a media server and you can also use its General Purpose Input Output (GPIO) pins to interact with your own circuits. I'm using my Raspberry Pi to find out how much traffic a miniature webserver can handle.

The Raspberry Pi's specs are as follows:

- ARM11 processor running at 700MHz
- 512MB RAM (256MB in Model A)
- HDMI out
- USB host and function
- SD card slot (4GB or larger recommended)
- audio out
- RCA Video
- 10/100 ethernet (model B only)
- VideoCore 4 Graphics Processing Unit
- Wifi with USB dongle

[1 Comment](#)

Set Up

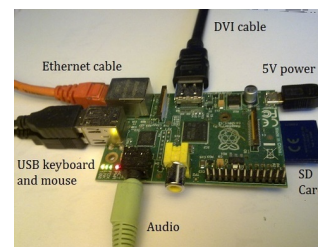
I started with the Raspberry Pi Model B Board and starter kit and set it up as described in the Raspberry Pi User Guide. You can get the Raspberry Pi as a kit with a USB wifi dongle, keyboard and mouse, or on its own if you already have a mouse and keyboard.

I configured my Pi to run at 800MHz instead of the default 700MHz. I installed the Apache web server, and copied some web pages to Apache's root directory. My Pi is connected to my router with an ethernet cable as this will provide better performance than a wifi connection.

Next, I set up port forwarding in my router so that people can connect to my Raspberry Pi from outside of my firewall. I created an account with no-ip.com, a service which redirects a web domain to my IP address.

I've set up a blog where I'll go into more detail [here](#).

[2 Comments](#)



GPIO

The Raspberry Pi's General Purpose IO Pins (GPIO) can be used to read data from sensors and send control signals to motors, lights or any circuit you can dream up.

The PiFace interface board makes it much easier to connect things like motors to your Raspberry Pi without damaging it.

You can write your own programs in Python or C to control devices connected to the PiFace board.

The wiringPi software library makes it easy to control and debug the GPIO pins. The table below is generated using the 'gpio readall' command.

```
+-----+--Rev2-+-----+-----+-----+
| wiringPi | GPIO | Phys | Name  | Mode | Value |
+-----+-----+-----+-----+-----+
|    0    | 17   | 11   | GPIO 0 | IN   | Low   |
|    1    | 18   | 12   | GPIO 1 | IN   | Low   |
|    2    | 27   | 13   | GPIO 2 | IN   | Low   |
|    3    | 22   | 15   | GPIO 3 | IN   | Low   |
|    4    | 23   | 16   | GPIO 4 | IN   | Low   |
|    5    | 24   | 18   | GPIO 5 | IN   | Low   |
|    6    | 25   | 22   | GPIO 6 | IN   | Low   |
|    7    | 4    | 7    | GPIO 7 | IN   | Low   |
|    8    | 2    | 3    | SDA    | IN   | High  |
|    9    | 3    | 5    | SCL    | IN   | High  |
|   10    | 8    | 24   | CE0    | ALT0 | High  |
|   11    | 7    | 26   | CE1    | ALT0 | High  |
|   12    | 10   | 19   | MOSI   | ALT0 | Low   |
|   13    | 9    | 21   | MISO   | ALT0 | Low   |
```

	14		11		23		SCLK		ALT0		High	
	15		14		8		TxD		ALT0		High	
	16		15		10		RxD		ALT0		High	
	17		28		3		GPIO 8		IN		Low	
	18		29		4		GPIO 9		IN		Low	
	19		30		5		GPIO10		IN		Low	
	20		31		6		GPIO11		IN		Low	
+-----+-----+-----+-----+-----+												

[0 Comments](#)

Follow me



Home

Home
Specs
Set Up
GPIO

Linux basics

What is Linux?
Files and directories in Linux
Users and Permissions

Bash

Bash basics

Raspberry Pi Cluster

Raspberry Pi Cluster

Raspberry Pi Server Cluster Tests

Improving cluster performance by tuning Apache

On Site Optimization

Adding more nodes to the cluster

Comparing the performance of Nginx and Apache web servers

HTML and CSS

HTML Basics

Making a complete HTML page

CSS - Cascading Style Sheets

CGI scripting

Writing CGI scripts on a Raspberry Pi

Writing CGI scripts in Python

Using Google charts in Python CGI scripts

Sending data to an HTTP server - get and post methods

Web Forms with Python

Using Python to set, retrieve and clear cookies

Setting up Nginx and uWSGI for CGI scripting

Raspberry Pi Temperature Logger

Building an SQLite temperature logger

Building a web user interface for the temperature monitor

SQL Databases

Set up an SQLite database on a Raspberry Pi

Accessing an SQLite database with Python

Using MySQL on a Raspberry Pi

Server Administration

Connect your Raspberry Pi to a USB hard disk

Change your Raspberry Pi's hostname

Share your Raspberry Pi's files and folders across a network

Back up your Pi to your Google drive

Get your Raspberry Pi web site on-line

Run a script on start up

GPIO

Raspberry Pi GPIO

Controlling GPIO with wiringPi

Controlling GPIO with wiringPi

Connecting your Pi's GPIO to a breadboard

Detecting a button press through GPIO

Using Interrupt Driven GPIO

Use GPIO to turn on an LED

Using Python to control GPIO pins

Connecting a temperature sensor to GPIO

Pi Face

Pi Face

Building a remote control vehicle using a Raspberry Pi

Pyplate

Install Pyplate on a Raspberry Pi with Apache
