

Raspberry Pi Based Server, Monitoring Virtual Library System

Authors

Millind Deshkar¹, Akshay Pathade², Chetan Mendhe³, Sayyam Shambhuwani⁴,
Vaibhav Pande⁵

¹Assistant Professor, Dept. Of IT, YCCE NAGPUR, Maharashtra, India

^{2,3,4,5}Student, Dept. Of IT, YCCE NAGPUR, Maharashtra, India

Abstract

Often a central server is very much expensive and has enormous electric power requirement. The recent introduction of the Raspberry Pi, a low-cost, low-power single-board computer, has made the construction of a miniature Cloud Data Centers more affordable. The Pocket Cloud based on Raspberry pi will be a single-board computing based server which will enable user to create plug and play cloud in any given network. Pocket Cloud will act as server to the web application "Virtual Librarian". This application deals with keeping record of the books in departmental library, handling user transactions like issuing book/books, returning book/books etc.

Index Terms— Raspberry pi, Pocket cloud, Virtual librarian, server, transactions, cloud

INTRODUCTION

Pocket cloud will be a low-end affordable server which expects low traffic and will enable a cheap alternative to small management as well as record keeping applications which are usually required to run 24x7 and occupies a costly server which is never optimally used by it. Affordable server offer hardware and software based customized solution, giving complete flexibility to the clients to customize RAM, hard drive, processor, and application version as per their requirements. Pocket cloud will be a dedicated server only for the web application 'Virtual librarian'.

A "Virtual Library" is a collection of resources available on one or more computer systems, where a single interface or entry point to the collections is provided. A dedicated server only for a single web application offers a very high level of security since access to it will be limited. Another plus point of having a dedicated server instead of a shared server is that you can configure it with software program to meet your specific needs. You can also benefit from scalability since it will be very easy for you to

increase the space used by your software. If you use a shared server then you will have to keep changing plans to accommodate your increasing needs.

I. Hardware

A. Raspberry pi

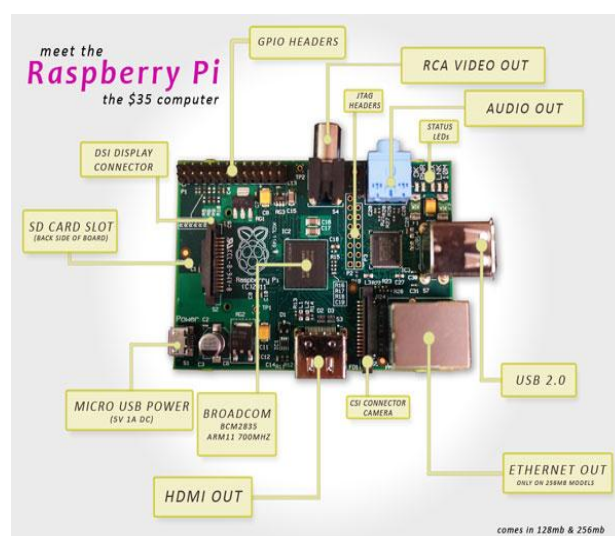


Fig. 1 Raspberry Pi Model B

B. Processor

The Broadcom Sock used in the Raspberry Pi is equivalent to a chip used in an old smart phone (Android). While operating at 700 MHz by default, the Raspberry Pi provides a real world performance roughly equivalent to the 0.041 GFLOPS. On the CPU level the performance is similar to a 300 MHz Pentium II of 1997-1999. The GPU provides 1 Gpixel/s or 1.5 Gtexel/s of graphics processing or 24 GFLOPS of general purpose computing performance. The Raspberry Pi chip, operating at 700 MHz by default, will not become hot enough to need a heat sink or special cooling. The Sock is stacked underneath the RAM chip, so only its edge is visible.

The LINPACK single node compute benchmark results in a mean single precision performance of 0.065 GFLOPS and a mean double precision performance of 0.041 GFLOPS for one Raspberry Pi Model-B board. A cluster of 64 Raspberry Pi Model-B computers, labelled "Iridis-pi", achieved a LINPACK HPL suite result of 1.14 GFLOPS (n=10240) at 216 watts for c. US\$4,000.

C. Over clocking

Most Raspberry Pi devices can be over clocked to 800 MHz and some even higher to 1000 MHz. In the Raspbian Linux distro the over clocking options on boot can be done by a software command running "sudoraspiconfig" without voiding the warranty. The Pi automatically shuttles the over clocking down in case the chip reaches 85 °C (185 °F), but it is possible to overrule automatic over voltage and over clocking settings (voiding the warranty). In that case, one can try putting an appropriately sized heat sink on it to keep the chip from heating up far above 85 °C.

Newer versions of the firmware contain the option to choose between five over clock ("turbo") presets that when turned on try to get the most performance out of the Sock without impairing the lifetime of the Pi. The five settings are:

- None; 700 MHz ARM, 250 MHz core, 400 MHz SDRAM, 0 overvolt

- Modest; 800 MHz ARM, 250 MHz core, 400 MHz SDRAM, 0 overvolt,
- Medium; 900 MHz ARM, 250 MHz core, 450 MHz SDRAM, 2 overvolt,
- High; 950 MHz ARM, 250 MHz core, 450 MHz SDRAM, 6 overvolt,
- Turbo; 1000 MHz ARM, 500 MHz core, 600 MHz SDRAM, 6 overvolt.

In the highest (turbo) preset the SDRAM clock was originally 500 MHz, but this was later changed to 600 MHz because 500 MHz sometimes causes SD card corruption. Simultaneously in high mode the core clock speed was lowered from 450 to 250 MHz, and in medium mode from 333 to 250 MHz.

D. RAM

On the older beta model B boards, 128 MB was allocated by default to the GPU, leaving 128 MB for the CPU. On the first 256 MB release model B (and Model A), three different splits were possible. The default split was 192 MB (CPU RAM), which should be sufficient for standalone 1080p video decoding, or for simple 3D, but probably not for both together. 224 MB was for Linux only, with just a 1080p frame buffer, and was likely to fail for any video or 3D. 128 MB was for heavy 3D, possibly also with video decoding (e.g. XBMC). Comparatively the Nokia 701 uses 128 MB for the Broadcom Video Core IV. For the new model B with 512 MB RAM initially there were new standard memory split files released (arm256_start.elf, arm384_start.elf, arm496_start.elf) for 256 MB, 384 MB and 496 MB CPU RAM (and 256 MB, 128 MB and 16 MB video RAM). But a week or so later the RPF released a new version of start.elf that could read a new entry in config.txt (gpu_mem=xx) and could dynamically assign an amount of RAM (from 16 to 256 MB in 8 MB steps) to the GPU, so the older method of memory splits became obsolete, and a single start.elf worked the same for 256 and 512 MB/S.

E. Networking

Though the Model A does not have an 8P8C ("RJ45") Ethernet port, it can connect to a

network by using an external user-supplied USB Ethernet or Wi-Fi adapter. On the model B the Ethernet port is provided by a built-in USB Ethernet adapter.

Peripherals Generic USB keyboards and mice are compatible with the Raspberry Pi.

F. Real-time Clock

The Raspberry Pi does not come with a real time clock, which means it cannot keep track of the time of day while it is not running.

As alternatives, a program running on the Pi can get the time from a network time server or user input at boot time.

A real-time clock (such as the DS1307) with battery backup can be added via the IC interface

II. Software

A. Raspbian

Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi. The initial build of over 35,000 Raspbian packages, optimized for best performance on the Raspberry Pi, was completed in June of 2012. However, Raspbian is still under active development with an emphasis on improving the stability and performance of as many Debian packages as possible. The Raspbian Server Edition is a stripped version with other software packages bundled as compared to the usual desktop computer oriented Raspbian. PiBang Linux is derived from Raspbian. A Raspbian image is a file that you can download onto an SD card which in turn can be used to boot your Raspberry Pi and Via APC into the Raspbian operating system. Using a Raspbian image is the easiest way for a new user to get started with Raspbian.

B. Apache Server

The Apache HTTP Server, colloquially called Apache, is the world's most widely-used web server software. This will be used as a web server to host virtual library.

C. Virtual Library

Virtual librarian system focuses on storing and managing records of departmental library books. It keeps track of the transactions of books. It provides user interface through which a user who may be a professor or a student is able to find the books available in library and the books which are issued for their account, fine etc.

III. Implementation

Raspbian which Debian optimized for Raspberry Pi hardware is deployed on it with optimum configurations required for Virtual Library application. Raspberry Pi was connected to a network using Ethernet port and accessed through SSH. Apache, PHP, MySQL, FileZilla FTP Server were installed through SSH access to the Raspbian on Pi to make a platform available for Virtual Library application. Once it was ready as a web server, all source files and media files were uploaded to htdocs using FTP access. MySQL database were exported from local development machine using php MyAdmin and uploaded to Pi's MySQL database and configuration file of Virtual Library was changed to work with new MySQL database credentials and new URL.

Working

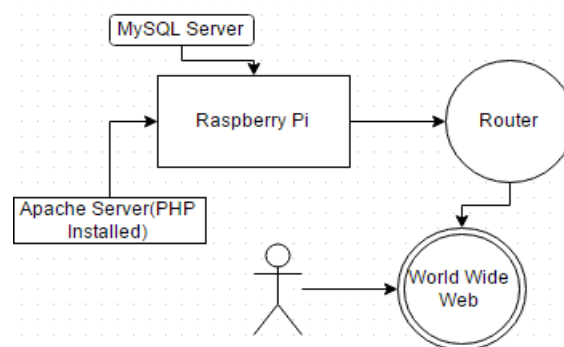


Fig. 2 Working of Pocket Cloud

MySQL database consists of the tables which hold the records of books, subscription and users. The PHP rests on Apache server and interact with user and MySQL database to retrieve, save and update data. Router assigns unique static IP address to Raspberry Pi and is connected to World Wide Web. Any user (Administrator, Subscriber, Librarian) access this portal on internet enabled machine using a browser. After the authentication process subscriber can,

- Search books and e-books from the inventory.
- Check availability
- Request to issue a book or download e-book.

The search engine functionally uses deep search methodology, the context of the books is scanned and along with title, topic names can also be used to search books. This enable book discovery through topic name to the user.

IV. RESULT/ CONCLUSION

The virtual Library can be accessed and modified from anywhere in the world using internet through personal computer or Smartphone. The system is being connected via Ethernet port to network successfully. Which provide low cost alternative to the traditional server at very low price. Work can be extended by using this pocket cloud to host other useful applications too

REFERENCES

1. N. Suresh, E.Balaji, K.JeffryAnto, J.Jenith "RASPBERRY PI BASED LIQUID FLOW MONITORING AND CONTROL" International Journal of Research in Engineering and Technology Volume: 03 Issue: 07 | Jul-2014, Available @ <http://www.ijret.org> 122
2. Md.Kawser Jahan Raihan ,Mohammad Saifur Rahaman, Mohammad KaiumSarkar & Sekh Mahfuz Global "Raspberry Pi Image Processing Based Economical Automated Toll System" Journal of Researches in Engineering Electrical and Electronics Engineering Volume 13 Issue 13 Version 1.0 Year 2013, Publisher: Global Journals Inc. (USA)
3. Shoaib Rehman Soomro, Mohammad Arslan Javed Fahad Ahmed Memon, VEHICLE NUMBER RECOGNITION SYSTEM FOR AUTOMATIC TOLL TAX COLLECTION, Robotics and Artificial Intelligence (ICRAI), 2012 International Conference on, 22-23 Oct. 2012, p.g-125 – 129
4. Ahmad Zubair, Sourav Mahmood Sagar, Pran Kanai Saha, and ShaikhAnowarul Fattah, A Design for Low Cost Electronic Toll Collection System with Secure Data Communication, 6th International Conference on Electrical and Computer Engineering ICECE 2010, 18-20 December 2010, Dhaka, Bangladesh
5. Raspberry Pi based interactive home automation system through E-MAIL, Jain, S. et al, IEEE Conference Publications,2014.
6. An arduino based system provided with GPS/GPRS shield for real time monitoring of traffic flows, Costanzo, A. IEEE Conference Publications,2013
7. Fung Po,"The Glasgow Raspberry Pi Cloud: A Scale Model for Cloud Computing Infrastructures", ICACT 2010
8. Raspberry Pi based interactive home automation system through E-MAIL, Jain, S. et al, IEEE Conference Publications,2014.
9. Documentation and downloads for the RASPBERRY [Online]. <http://www.Raspberryypi.org/>
10. Pi-Face Interface. [Online]. <http://pi.c.s.man.ac.uk>
11. About the Apache HTTP Server Project". Apache Software Foundation. Archived from the original on 7 June 2008.Retrieved 2008-06-25.
12. <http://www.mathworks.com/hardware-support/raspberry-pi.html>
13. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 1997.