Raspberry Pi Clustering for VoIP Communication

M.W.P Maduranga[1], H.M Manjula Perera [2]

[1] *Department of Electrical and Electronic Technology, University of Vocational Technology, Sri Lanka*
[2] *Department of Networks and System Provisioning , Dialog Axiata PLC , Sri Lanka*
[1] pasan@univotec.ac.lk

***Abstract*—**

In this article, we examined a Raspberry Pi cluster for VoIP applications. The VoIP application is run in two different experimental setups. One uses a single Raspberry Pi board and the other uses a Raspberry Pi cluster with three Raspberry Pi boards as nodes. In this cluster, you set up a parallel processing algorithm called Massage Transfer Interface (MPI) to share the load within the node. Both settings generate simultaneous voice calls. CPU utilization and call count (flow) performance were analyzed for both settings. Clustering performance has been significantly improved.

Keywords— Clustering, Massage Passing Interface (MPI), Raspberry Pi, Embedded Systems.

I. INTRODUCTION

The Raspberry Pi is a credit card-sized, low-cost embedded computer. Raspberry Pi is used to design several types of embedded applications, Internet of Things (IoT) applications, and more. If multiple Raspberry Pi boards are interconnected, consider using a Raspberry Pi cluster. A RaspberryPi cluster can do more calculations than a single board. In this study, we implemented a Voice over IP (VoIP) application on a Raspberry Pi cluster with one Raspberry Pi board and three interconnected Raspberry Pis. In this cluster setup, one board functions as the primary node (load balancer) and the other two nodes function as nodes. The system also includes an Ethernet switch and Internet connection. The raspberry Pi is a kernel level programmed in Linux, so it can be used as a server. The system uses free available phone software such as Asterisk and FreePBX. Synchronized voice calls were generated with two experimental settings: a single Raspberry Pi setup and a Raspberry Pi cluster. The recorded results were analyzed to observe the performance of the two experimental devices. The second section describes recent work on Raspberry Pi clustering. Section III details the design and implementation of the experimental setup. The fourth section details the experimental results, and the fifth chapter provides a summary.

II. RELATED WORKS

Several contributions are made on Raspberry Pi clustering such as applications of super computers, comparison of several load balancing methods on clusters etc [1], [2]. Few recent works reported as below.

Amjad Mahmood and Irfan Rashid have completed their research and analyzed the performance of random, round robin, weighted round robin, and minimum connection scheduling, which are well-known load balancing algorithms on clustered web servers. They are studying the performance of the four algorithms given by the simulation to evaluate performance under different workloads and conditions. The performance of the four algorithms was evaluated under different workloads and conditions [3].

An eco-friendly web server feasibility study was conducted using an ultra-low power micro-computing board to serve web content. They studied the power and performance trade-offs of microcomputing boards. They argue that a low-power computing platform can deliver enough performance for small websites run by small businesses and teams, while increasing requests per watt. Static and dynamic web content can be served using low-power computing platforms such as the Raspberry Pi platform. They tried to make 200 requests on each R-Pi board, then collected more requests and tried. The results show that clustered web servers can reach up to 17-23 times on a typical server system [4].

III. DESIGN AND IMPLIMENTATION

A. System Overview

The system consists of three Raspberry Pi boards, an Ethernet switch, an analog phone, an IP phone, a Raspberry Pi display, and a webcam. The Raspberry Pi board is interconnected with an Ethernet that contains all the IP addresses assigned to each device, as shown in Figure 1.

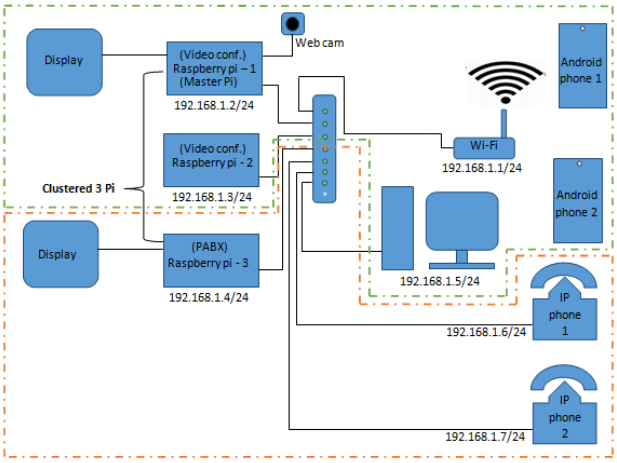

Figure 1. System Overview

B. Hardware

Raspberry Pi 3

We used three raspberry pi 3 B plus modules to build the setup.

Figure 2: Raspberry Pi 3

After the basic settings are configured, two open source operating systems (OS) are installed on the primary node of the video conferencing system and the PABX system. The Raspbian 4.14 operating system (OS) is used to connect the two Pi modules (master Pi and backup) of the video conferencing system. Raspbian comes preloaded with a number of software for education, programming, and general use.

After installing the operating system (Raspbian version 4.14) on the master Pi and backup pi, you need to install the main software. This allows you to use the processing power of all pis (master Pi and backup Pi). In this cluster, we configured a parallel computing algorithm called MPICH [9],[10]. Since Raspberry Pi supports an installed Python encoding environment, the easiest way is to install Python in the MPI interpreter. Here we use the Jitsi app for video conferencing. You can also easily install an Android phone. Jitsi Meet is a fully encrypted 100% open source video conferencing solution. In addition to video conferencing [5],[6],[7],[8]

IV. TESTING AND RESULTS

Basically, we performed performance tests in two ways. First, the setup uses a single Raspberry Pi board, generates simultaneous calls for a specific period, and analyzes the CPU performance of the processor. Next, connect the two Raspberry Pi boards in the setup and configure MPICH parallel computing on the primary node. The algorithm generates the same call as in the first case. Figures 3 and 4 illustrate the operation of the processing functions for clustered and non clustered scenarios.

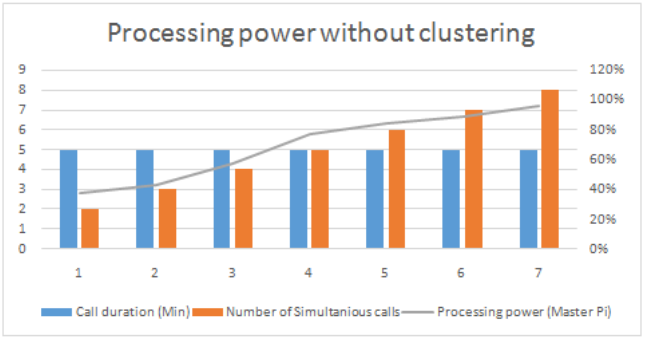

Figure 4: CPU usage when single Raspberry Pi uses

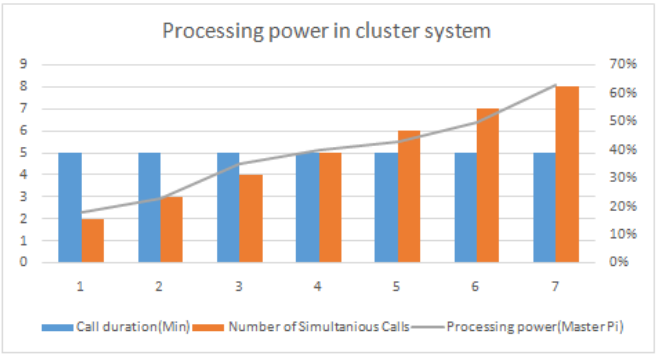

Figure 4: CPU usage in the cluster with MPICH

V. SUMMARY

We experimented how the parallel computing algorithm MPICH is used in a Raspberry Pi cluster for VoIP applications. In this experiment, we set up two experimental setups using a single Raspberry Pi board and set up a cluster using three Raspberry Pi boards. Multiple simultaneous VoIP calls were generated and CPU performance was analyzed based on the number of calls. It has been observed that CPU utilization drops dramatically when clusters are used.

VI. FUTURE WORKS

The Raspberry Cluster performs functions and computing as a supercomputer. Considering commercial video conferencing systems, PABX systems, and CCTV systems are very expensive. With this Rapberry Pi cluster, you can implement this VoIP

communication product at a very low cost while meeting sufficient system requirements. Future work in this experiment will allow you to explore and analyze other parallel processing algorithms depending on the required application.

REFERENCES

[1]apache.org (Accessed on 2016.01.12) <https://httpd.apache.org/docs/2.4/mod/mod_proxy_balancer.html>

[2]raspberrypi.org. (2015a). Raspberry Pi 2 - Model B Retrieved April 10,2016, <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>

[3]Ashwin Kumar T K, Jongyeop Kim, K M George and NohpillPark,"Dynamic data rebalancing in Hadoop", *Computer and Information Science (ICIS) 2014 IEEE/ACIS 13th International Conference on*, pp.315-320, 2014.

[4]B. Varghese, N. Carlsson, G. Jourjon, A. Mahanti and P. Shenoy," Greening web servers: A case for ultra low-power web servers," *Green Computing Conference (IGCC), 2014 International*, Dallas, TX, 2014, pp.1-8.

[5] "Real-Time Transmission of Voice over 802.11 Wireless Networks Using Raspberry Pi" International Journal of Engineering Development and Research©2014 IJERD | Volume 2, Issue 1.

[6] Lopez, "Hadoop on a Raspberry Pi," Datanami, 27-Nov-2013. [Online]. Available: [Accessed: 16-Mar-2017]. <https://www.datanami.com/2013/11/27/hadoop_on_a_raspberry_pi/.>

[7] I.-Y. Jung, K.-H. Kim, B.-J. Han, and C.-S. Jeong, "Hadoop-Based Distributed Sensor Node Management System,"Int. J. Distrib. Sens. Netw., vol. 10, no. 3, p. 601868, Mar. 2014.

[8] M. Maurya and S. Mahajan, "Performance analysis of MapReduce programs on Hadoop cluster," in 2012World Congress on Information and Communication Technologies, 2012, pp. 505–510.

[9] M. Sekiyama, B. K. Kim, S. Irie, and T. Tanikawa, "Sensor data processing based on the data log system usingthe portable IoT device and RT-Middleware," in 2015 12th International Conference on Ubiquitous Robotsand Ambient Intelligence (URAI), 2015, pp. 46–48.

[10] M. W. P. Maduranga and R. G. Ragel, "Comparison of load balancing methods for Raspberry-Pi Clustered Embedded Web Servers," *2016 International Computer Science and Engineering Conference (ICSEC)*, Chiang Mai, 2016, pp. 1-4.