Aalto University
School of Science
Degree Programme of Mobile Computing

Bo Pang

Energy Consumption Analysis of ARM-based System

Master's Thesis
Espoo, Augest 31, 2011

Supervisor:	Professor Antti Ylä-Jääski, Aalto University
Instructor:	Zhonghong Ou post-doc researcher

Aalto University
School of Science
Degree Programme of Mobile Computing

ABSTRACT OF
MASTER'S THESIS

Author:	Bo Pang		
Title: Energy Consumption Analysis of ARM-based System			
Date:	Augest 31, 2011	**Pages:**	68
Professorship:	Data Communications Software	**Code:**	T-110
Supervisor:	Professor Antti Ylä-Jääski		
Instructor:	Zhonghong Ou post-doc researcher		

With growing computing demand, energy consumption is standing out as a salient concern of data centers with large numbers of servers. To build high energy-efficient data centers, the low power consumption ARM processors have been investigated as an alternative solution to replace the conventional high-end server processors e.g. Intel and AMD processors. This study exploits this concept and measures how much energy saving can be achieved by adopting ARM processors. We design four different experiments representing WEB server applications, In-memory database, video transcoding applications and Hadoop to evaluate whether ARM based server is able to provide the same level of service in a more energy-efficient way, and which application is more suitable for ARM based server. Additionally, the lesson learn from this study is discussed in the end of this study.

Keywords:	ARM Processor, Energy Efficiency, WEB, In-memory Database, Video Transcoding, Power Management
Language:	English

Acknowledgements

First of all, I want to thank to my motherland, China, who gives me the opportunity to study abroad. Without the reform and opening-up policy, nothing I have done today would be possible. I would like to thank Aalto University for its two years high quality master education, from which I will benefit for my whole life. I also would like to express my sincere gratitude to my supervisor Professor Antti Ylä-Jääski and instructor Zhonghong Ou, who not only patiently guide my thesis writing but also teach me how to become a qualified scientific researcher. After many years, I may forget what this thesis is about, but the research attitude he taught me will always be remembered. At last I would like to thank my parents and my girlfriend, their consistent support helps me get through all these two years' study. Without the faith of going home to reunite with them, I would not graduate on time. In the end, I want to thank all my friends, especially my good brother Horoscope Zhu who guided me in Matlab and Dota earnestly, accompanied me to pass rememberable days in Otaniemi.

Espoo, Augest 31, 2011

Bo Pang

Abbreviations and Acronyms

HDD	Hard Disk Drive
EE	Energy Efficiency
LVS	Linux Virtual Server
ARP	Address Resolution Protocol
MAC	Media Access Control
HDFS	Hadoop Distributed File System
RAID	Redundant Array of Independent Disks
OS	Operating System
HTTP	Hypertext Transfer Protocol

Contents

Abbreviations and Acronyms **4**

1 Introduction **7**
- 1.1 Motivation and Challenge 7
- 1.2 Technology Trends 8
- 1.3 Problem Statement 10
- 1.4 Scope of the Thesis 10
- 1.5 Structure of the Thesis 10

2 Background **12**
- 2.1 ARM Structure 12
- 2.2 Web Server Technologies 13
  - 2.2.1 Apache 13
  - 2.2.2 Nginx 13
  - 2.2.3 LVS 14
- 2.3 Multimedia Transcoding Libraries on Linux 15
- 2.4 In-memory Database 15
- 2.5 Hadoop 16

3 Experimental Environment **21**
- 3.1 Pandaboard Specifications 21
- 3.2 Intel Workstation 21
- 3.3 Test Methodology 22

4 Experimental Design **26**
- 4.1 HTTP Server 26
- 4.2 In-memory Database 30
- 4.3 Video Transcoding 32
- 4.4 Hadoop Map-Reduce Task 34

5 Result Analysis **36**
5.1 HTTP Server Experiments 36
5.2 In-memory Database Experiment 54
5.3 Video Transcoding Experiment 56
5.4 Hadoop Experiment 57

6 Discussion **59**

7 Conclusions **62**
7.1 Key Findings 62
7.2 Limitations of Study 62
7.3 Future Work 63

Chapter 1

Introduction

1.1 Motivation and Challenge

Along with information technologies playing more and more important roles in modern society and economic, information processing becomes a common need for a variety of industries. New Internet services such as video streaming, social network sites, data mining and business intelligence and Internet search optimization require a huge amount of computing resource. Normally the scale of computing can not be achieved by workstations or personal computers, thus distributed servers cluster system built of thousands of servers in data center is a smart choice for massive-data processing, some Internet giant companies such as Google, possessing more than 200,000 servers from estimation in 2005, use a great amount of servers to build a distributed system to meet the big computing demand. However the significant growing number of servers makes power consumption stand out as a salient issue. The sum of electricity spending on servers' computing, data processing, data storage and device cooling is becoming a big concern for world-spread data center owners. From Figure 1.1 below, the money spent on power supply and cooling is going to overtake the upfront investment on servers' hardware, hence the reduction of power consumption on server is becoming a new concern of data center investors.

Nowadays data center electricity consumption becomes a big challenge for the world. From Jonathan study[15], power consumption used by servers doubled from 2000 to 2005. And in 2005 approximately 0.6% electronic power was consumed by servers of all kinds in US, which is equal to the amount of electricity consumed by all the color televisions in US. In addition to the power directly consumed by servers, the same amount of power was used by auxiliary and cooling systems, which provide stable power and cooling to

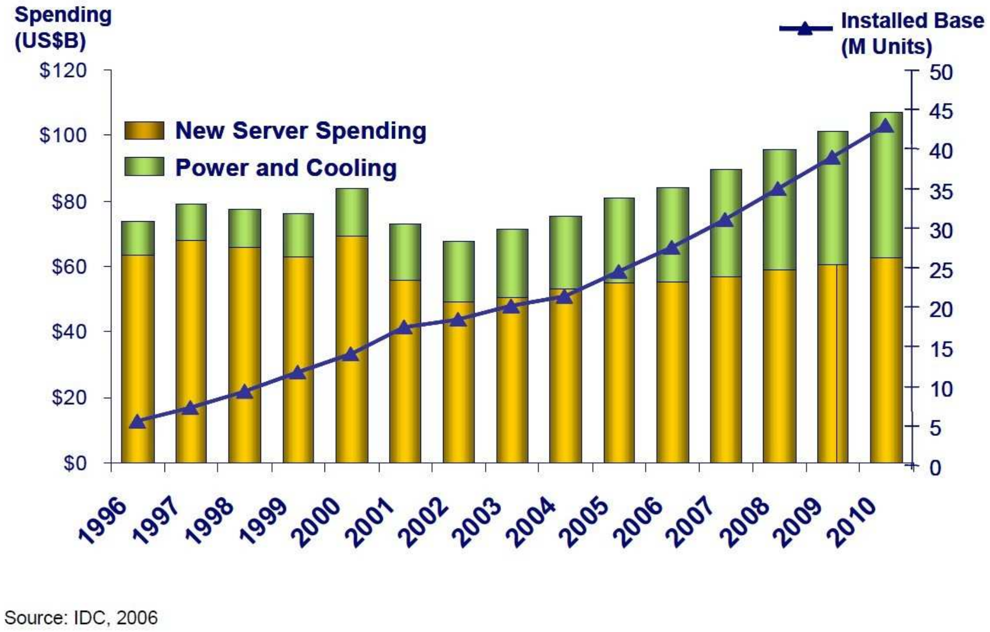


Figure 1.1: IDC 2006-2010 prediction of server spending in 2006 [14]

processors to keep the servers working stably. The total data center power demand in 2005 required about five 1000 MW power plants to be built in US, and fourteen plants in the world. Moreover IDC also predicted the total electricity consumption on servers worldwide will be $44.5 billion by 2010 which requires at least 10 gigawatt power plants to be constructed for that[5]. Thus power saving solutions for computing may bring significant economic profit to data center and also protect the environment.

1.2 Technology Trends

ARM, a micro processor architecture originally designed for mobile and embedded systems, starts to step into server market recently. Microsoft first announced ARM-based PC at the Computer Electronics Show (CES) in Las Vegas[17]. In addition, ZT Systems[51] announced a server powered by sixteen ARM cortex-A9 processor cores, which draws a maximum 80 watts system power that is even less than an Intel Xeon series processor's power consumption. Hence there is a new trend for data centers to adopt multicore ARM processors to build energy efficient and cooling efficient servers to execute computing tasks that originally executed by Intel processors based servers. Because of the recent server cost's shrink, ARM processor's notable advantage of air-cooling and low power-consumption is becoming more important to data centers' investment. And another motivation to adopt ARM processor is to solve the problem of increasing core density on processor, for accumulated devices, such as rack-mounted blade servers, the ARM processors' low heat generation characteristic and scattering layout helps decrease the complexity of motherboard integration and cooling design.

Nowadays most of data centers are adopting X86 architecture processor. From Figure 1.2, it is obvious that Intel Xeon series processor occupied roughly two thirds of server market, while the high-end server processor IBM power series penetrated 20% shares and AMD, Intel's strongest competitor, seized only 8.5%. Comparing with X86 architecture processor, the ARM based-server is still a new participator in the server market, only 2.3% users chose it. Thus replacing current high power consuming processors with ARM processors may bring significant potential power saving to data centers, ARM processor based server is a good candidate for leading a new evolution to high energy-efficiency computing.

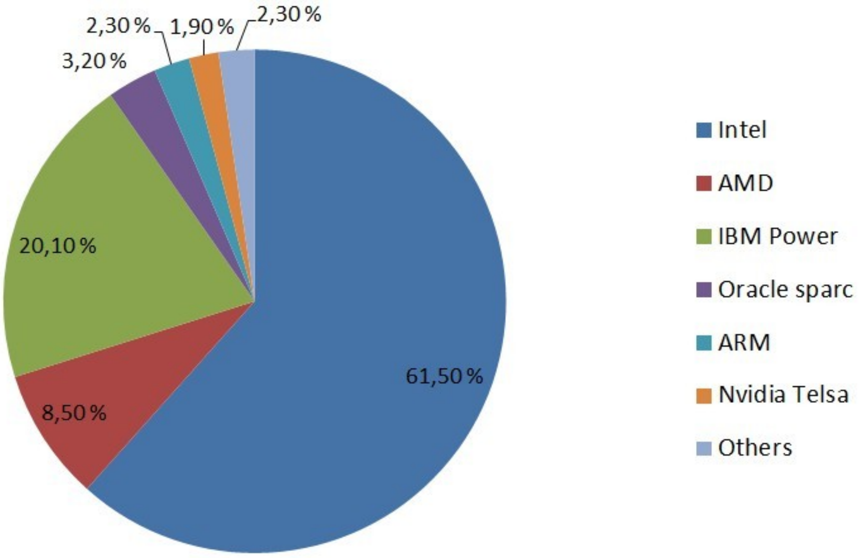


Figure 1.2: Server market share of China in 2010

1.3 Problem Statement

ARM-based server is upcoming, but there is no scientific research proved the feasibility and how power efficient ARM server would achieve. Thus the goal of this study is to answer the following questions:

- Is ARM-based servers/server cluster able to substitute Intel Xeon servers with the same performance but lower power consumption?
- Which kinds of applications are more suitable for ARM-base servers/server clusters?

For answering these two questions, a ARM processor-based testbed, consisting of four Dual-core ARM Cortex-A9 processors, is built, running the same task against an Intel Server based processor to compare their power consumption and performance.

To compare these two different platforms in a fair manner, we use the energy efficiency (EE) definition from Dimitris's paper[48]: the ratio of useful work done to the amount of used power.

$$\mathrm{EE} = \frac{\text{Work done}}{\text{Energy}} = \frac{\text{Work done}}{\text{Power} * \text{Time}} = \frac{\text{Performance}}{\text{Power}} \tag{1.1}$$

The EE will be a benchmark thoughout all the experiments in this study.

1.4 Scope of the Thesis

This thesis focuses on metering and comparing power efficiency of ARM processor based server and Intel processor based server in some specific scenarios and analyses which kinds applications or services are better fit the ARM processor based system. In this thesis four sets of experiments are conducted on two platforms, which represent the following applications: Web application, In-memory SQL database, video transcoding application and Hadoop applications.

1.5 Structure of the Thesis

This paper is organized as follow: chapter 2 introduces technologies and applications that are used as test objects in this paper, and a brief overview of their working principles are also given. And chapter 3 describes the test environment of experiments. Chapter 4 outlines the design details of each experiments with brief descriptions about the involved benchmarks. Chapter 5 presents the experiment results and analyses the energy efficiency of both platforms in each experiment. Chapter 6 makes a discuss of strategy that will further lower the power consumption of the ARM processor based server cluster. Chapter 7 draws a conclusion and discusses the limitations of this study and plans the future work for continuing this work.

Chapter 2

Background

This chapter introduces background information about technologies related to this study, including architecture of ARM processor and other application technologies which are going to be used as test tools or objects in this work.

2.1 ARM Structure

The ARM is a 32-bit reduced instruction set computer (RISC) instruction set architecture (ISA) developed by ARM Holdings company [3]. The ARM Holding company only designs the architecture of processors and licenses other manufacturers to produce processors based on its design. The licensee list includes IBM, Texas Instruments, Samsung, etc, almost every big ICT manufacture company has involved in ARM architecture processor producing. The reason why so many producers choose ARM architecture is because of its advantages: low power consuming and simplicity. ARM's RISC keeps every instruction simple and of uniform length, the special design, unlike the complex instruction, allows RISC processors demanding less registers and circuits. Moreover the uniform length instruction offers ARM processor better performance from instruction pipeline technology that exploits processing circuit at its maximum efficiency. Thus the ARM's simple architecture generates less heat than Intel X86 architecture [12] but is still able to achieve good performance. Thanks to ARM processor's unique design, ARM counts on its high energy efficiency to dominate embedded system market [4], for instance, the prevailing Apple products, Iphone and Ipad families are all carrying ARM processors, moreover many mainstream operation systems has its ARM distribution, even Microsoft announced in Jan 2011 that it planned to support ARM-based system [2]. With several years' accumulation and evolving, ARM processors gained enough computing power to participate server-end computing competition. For example, ARM cortex-A15, which contains four 2.5GHZ cores, is equal to Intel Core i7 2920XM processor at clock rate. Thus it is possible to adopt ARM-based processors to build a multi-processors server to overtake the jobs processed by Intel Xeon series servers. According to the news from industry, several manufacturers have already built their own ARM-based servers, for instance, ZT system proposed its R1801e 1U rackmount ARM server [51], which uses eight ARM Cortex-A9 processors with solid state disk. Surprisingly the total power spent on the whole server is less than 80W on 16*2GHz computing speed. All in all ARM-based server is becoming a competitive candidate for low power consumption servers [31].

2.2 Web Server Technologies

2.2.1 Apache

Apache HTTP Server ('httpd') [42] is a project under Apache Software Foundation, it is one of most popular Web server in the world and served for more than 224 million websites by June of 2011 [28]. Moreover it can be installed on a variety of operating systems, such as Linux, Solaris, Mac OS X and Microsoft Windows. What is more, Apache also provides an open interface for plug-ins, which enhances Apache with useful features. Installing the PHP plug-in, Apache obtains the ability to host PHP Web applications.

2.2.2 Nginx

Nginx [29] is an open-source high-performance HTTP server and reverse proxy, which was first developed by Igor Sysoev in 2002. Nginx has a stunning performance efficiency when handling static-resource requests, and it is also able to be used as a load balancer. Since Nginx belongs to message-oriented asynchronous servers, unlike httpd, Nginx does not spawn new processes for each incoming connection, but rather uses a threads pool to handle incoming request events. This design uses a fixed number of worker threads and thus consumes limited memory resource when handling massive http connections. According to an experiment made in 2008 [13], Nginx defeated Apache at processor load and response time respectively.

2.2.3 LVS

Linux Virtual Server (LVS) [21] is an advanced Linux load balancing solution, which provides server cluster high-scalability, high-performance and high-availability load balance service. LVS is an IP layer load balance software, thus it is able to serve any application beyond IP layer. LVS supports three load balance modes and eight schedule algorithms. The three load balance modes include Virtual Server via Network Address Translation (VS/NAT), Virtual Server via IP Tunneling (VS/TUN) and Virtual Server via Direct Routing(VS/DR). The VS/NAT mode is the simplest. Application servers are assigned private IP addresses, and LVS in this mode plays a role as a Network Address Translation server and also the only gateway between the server cluster and clients. When clients send requests to the LVS, it will look up in its server address table to find a real server in the cluster and map the destination public address to the private address of the real server, and then sends the request to the real server directly. And when the real server replies to the client, the LVS will translate the address back and deliver the reply package back to the client outside the private subnet. The VS/TUN mode works similar to virtual private network (VPN) tunnel mode. The LVS adds a new IP destination header, containing the real server's IP address, in front of the incoming IP package. Through the LVS tunnel the new header will be removed by the tunneling protocol on the real server, and after unpacking the payload of the request, the real server processes the request and replies clients directly. And the last mode is VS/DR which, unlike the former two modes, works at data-link layer, in this mode all the real servers and LVS share the same IP address, but only LVS replies the Address Resolution Protocol (ARP) requests from the gateway, so all the requests are forwarded by gateway to LVS and LVS replaces the request's destination MAC address with one certain the real server's and forward it to that real server, and the real server will reply clients directly. Corresponding to these three modes, eight schedule schemes decide which real server would be chosen to reply requests. The eight schedule schemes are listed as follows:

- Round-Robin Scheduling
- Weighted Round-Robin Scheduling
- Least-Connection Scheduling
- Weighted Least-Connection Scheduling
- Locality-Based Least Connections Scheduling
- Locality-Based Least Connections with Replication Scheduling

- Destination Hashing Scheduling
- Source Hashing Scheduling

Round-Robin schedule strategy polls real servers list and selects one fairly, this scheme is the most fair scheme, while other schemes are suitable for capability-unbalanced server cluster. In this paper's experiments, LVS in VS/NAT mode with Round-Robin schedule strategy is selected as front-end load balancer configuration.

2.3 Multimedia Transcoding Libraries on Linux

Nowadays, video broadcasting Web service is becoming an inseparable part of people's lives. Those video broadcasting websites, such as YouTube and YouKu, allow users to upload their own videos in various formats and transcode them into uniform Flash compatible format and present the videos on their pages to viewers. This thesis will study the transcoding process as a benchmark for energy efficiency comparison. On Linux, there are two popular multimedia transcoding libraries/tools. The first is MEncoder [47], which is included in MPlayer project. MEncoder, as MPlayer's codec library, supports diversity of video and audio formats covering most common-use video formats. The other one is FFmpeg [46] that is also an open-source project providing library and program for multimedia data processing. FFmpeg is famous for its library named Libavcode that is a video/audio codec used by several other projects. FFmpeg supports many video and audio formats as well, and many hardware platforms are on its support list, which includes x86(IA-32 and x86-64), PPC (PowerPC), ARM, DEC, SPARC, and MIPS architecture. Besides these two tools, Java has its own media library called Java Media Framework, which enables audio, video and other time-based media to be added to applications and applets built on Java technology. This framework can capture, playback, stream, and transcoding multiple media formats and has cross-platform feature [30].

In this thesis, the MEncode is responsible for splitting and merging the video files and the video slice will be transcoded by the FFmpeg.

2.4 In-memory Database

In this study, SQLite [38] is used as an in-memory database. The in-memory database is response-speed-oriented database that relies on main memory rather than hard disk. Because all the data are stored in main memory, there is no I/O operation needed, therefore time critical tasks and high frequency tasks can be completed in high speed by in-memory database. SQLite is a popular in-memory database, of which the biggest advantage is simplicity, but SQLite supports ACID (atomicity, consistency, isolation, durability) properties as well, as traditional database does. Although the SQLite is normally used in embedded systems and in small or middle website projects, with Partition and Shard technologies getting mature, some SQLite databases' cluster may outperform a single industrial database server.

2.5 Hadoop

Hadoop is an open-source project under Apache foundation for distributed, reliable and scalable computing. Hadoop, inspired by the Google's Google File System (GFS) [11] and MapReduce [8], implements most functions of GFS and Google's MapReduce. Its goal is to conduct data intensive computing on large cluster of commodity hardwares. This framework is spreading around several giant IT companies, such as China Mobile, TaoBao and Yahoo, which adopt this software for analyzing data for their business and also contribute back to the Hadoop project's source code [49]. The latest stable version is 0.20.203.0 [43], which is used as test benchmark in this thesis. Hadoop is comprised of three parts: Hadoop common project, Hadoop Distributed File System (HDFS) and Hadoop MapReduce.

The common project [39] is common utilities for supporting other Hadoop-related projects. This package includes library jar files and scripts to manipulate Hadoop instances and also other materials, such as source codes, documents and examples.

HDFS [40] is the foundation for Hadoop Map-reduce. It provides a distributed file system for upper applications. And since it is written in Java, it gains portability to run on heterogeneous operating systems, moreover its architecture's simplicity enables Hadoop's scalability.

HDFS is designed to process batch jobs rather than user interactive jobs, hence low data access latency is a less important goal for HDFS whose actual main goal is high throughput of streaming data access. Since it is not for general purpose application, HDFS only implements part of POSIX standard and ignores some requirements for better data throughput rate. The most significant feature of HDFS is its write-once-read-many file access mode, which means once a file's writing process is finished it can not be changed anymore. This feature dramatically advances the simplicity of system structure, there would not be write lock issue for designers. But in the future HDFS may support appending write operation. Thus HDFS is not suitable for storing data which is incline to changing, such as user's information. Instead of data representing in-time situation, system logs and historical data are suitable to be stored in HDFS and be analyzed later on.

HDFS file organization is very similar to traditional directory-file hierarchy, in which user can create directory and put files under it. But there is a small difference from existing file systems, for example, it does not support soft or hard link which is prevailingly supported by most file systems. Moreover files and directories on HDFS will be split into two parts: meta-data and data file, which are stored in different parts of system. Meta-data is managed on the Name node and real data file is stored as file in local file system of Data node, those two kinds of nodes will be discussed in the s subsequent paragraphs.

A node is a Java software instance running upon commodity machines. There are two types of nodes: Data node, Name node, and Name node is the core of the HDFS. Because HDFS is a star-style structure, there is a unique Name node in the center of the system, which takes charge of maintaining meta data and controlling data blocks replication (data file is comprised of series of blocks) and interacts with clients about the block locations. This design also simplifies the HDFS structure. Name node as a central server controls every movement of Data node.

Data node is the real worker, which manages file's content and replication process. In HDFS files are split into a sequence of equal-size blocks (except the last block). For fault tolerance, each block is replicated in several locations (the configuration of number of replicas is stored in the Name node), and all the replicas and the original blocks are spread across the HDFS cluster, like RAID 5 working principle. Figure 2.1 depicts the block and replica distribution across Data nodes. Data node reports its status and a block list of its repository to Name node periodically, and then the Name node will give reply with dictations about hierarchy changes, such as directory changing and files deleting, etc.

Figure 2.2 depicts the architecture of HDFS and communication transaction process. There are two communication protocols in this diagram, one is from client to nodes, and the other is from Data node to Name node. Both these two protocols are based on TCP/IP protocol and support remote procedure call (RPC). The protocol between Data node and Name node is used for periodically reporting. In addition to this internal protocol, as data keeper, Data node is queried directly by clients to read and write file data, and clients also need to change meta data on the Name node to update file system information, thus the client protocol is adopted for file operations.

To tackle with hardware failure, HDFS's data block replication and check point mechanism helps data availability and robustness. But the Name node is a single point of failure, and its failure may cause the whole HDFS fails to carry on any functions. Thus a secondary Name node is essential as a backup of Name node, and manual intervention is needed as well.

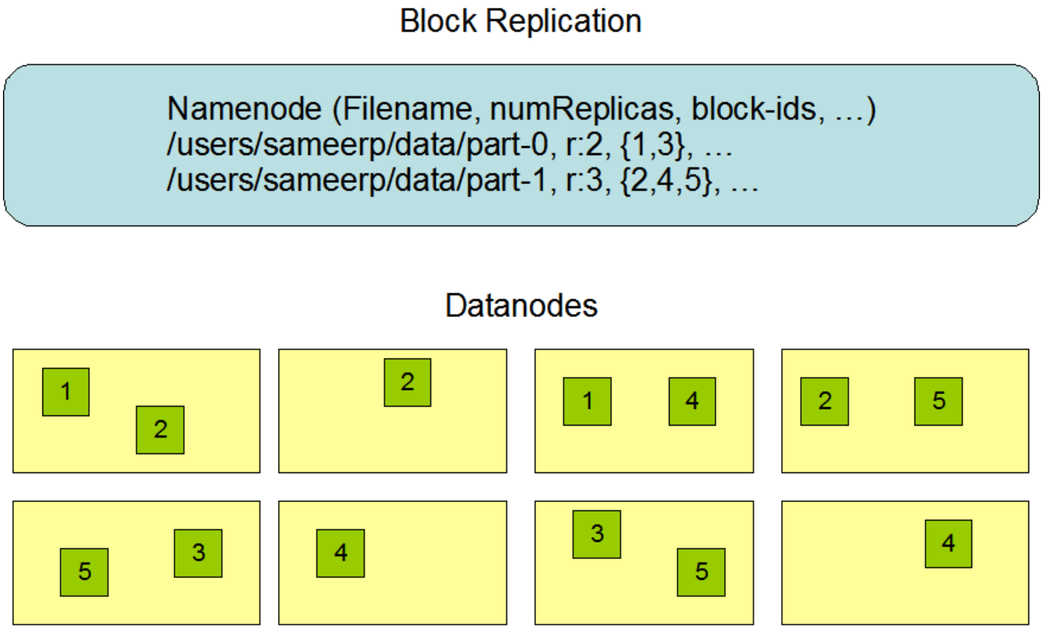


Figure 2.1: Distribution of replicas of data in HDFS [45]

Running on the HDFS, the Map-Reduce programming paradigm [41] is the last component of Hadoop. Similar to Google's MapReduce, Hadoop MapReduce programming model is also for parallelized computing of big set of data. This programming model has three phrases: map, shuffle and reduce. In map phrase, some processes, so called mappers, will analyze raw data and output in key-value pair style into intermediate files. The second phrase is called combine or shuffle, which sorts the key-value pairs in the intermediate files and gathers the data with the same key into one node via HTTP for the following reduce phrase. Data in the reduce phrase will go through some processes, such as summing up or counting number, and output in key-value style to the final output file.

From the Figure 2.3, it is obvious that one job contains several map tasks and these tasks are running on separated nodes and each one processes a piece of input file (normally a HDFS file block) and output intermediate file on the local file system. The framework will collect those intermediate files and distribute them into different nodes by keys. After that reducer will fetch the intermediate files as input, and output result into output file. Got

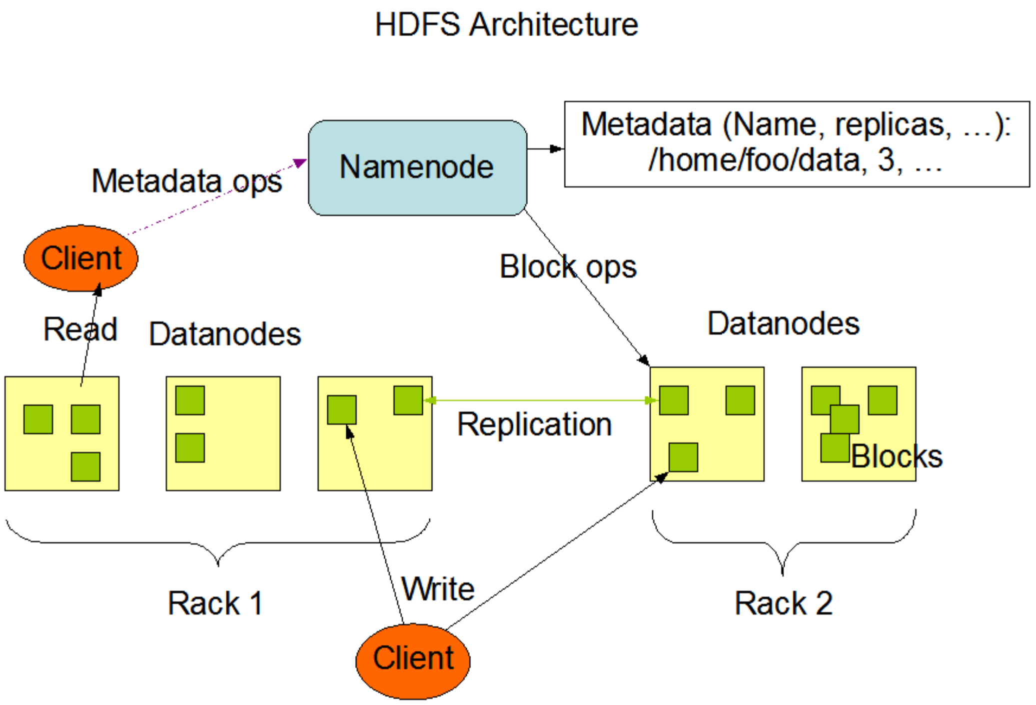


Figure 2.2: Architecture of HDFS [44]

the job finished signal, client can fetch its job's result file from HDFS.

Hadoop implements Map-reduce by two kinds of nodes, called Job Tracker and Task Tracker. Like HDFS structure, Job Tracker manages job information and assigns tasks to Task Tracker. The design principle of Hadoop is moving computing rather than data, thus Job Track will get data file block information from Name node and assign task to the Data node, which holds the file block or the closest to the data file block node, for saving backbone network traffic. Figure 2.4 depicts the classical deployment of Hadoop: Job Tracker and Name node are deployed on one master node, and Task Trackers and Data nodes are deployed on several slave nodes. This design takes advantage that map task may compute local data block instead of fetching blocks from other slave nodes [8].

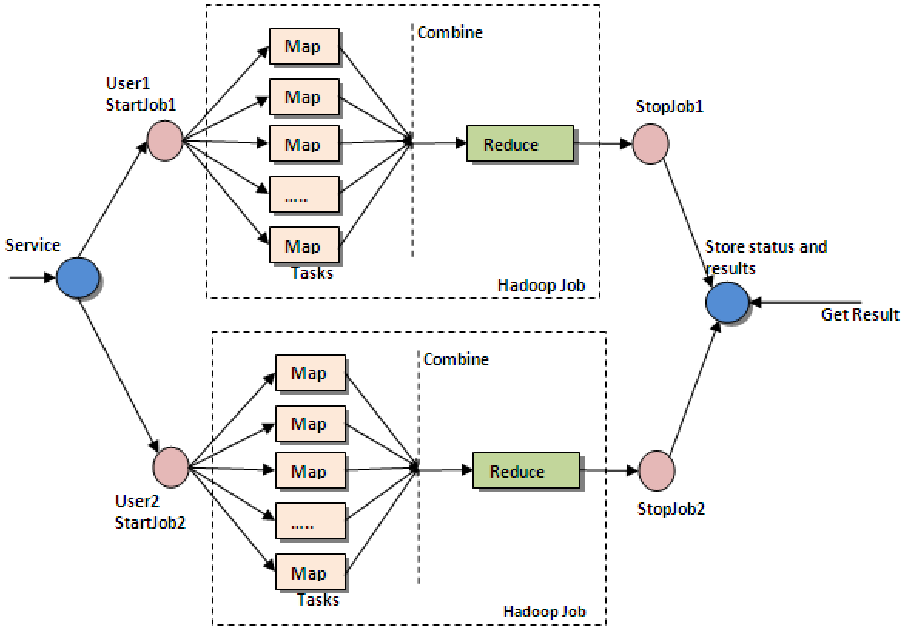


Figure 2.3: Map-reduce paradigm [1]

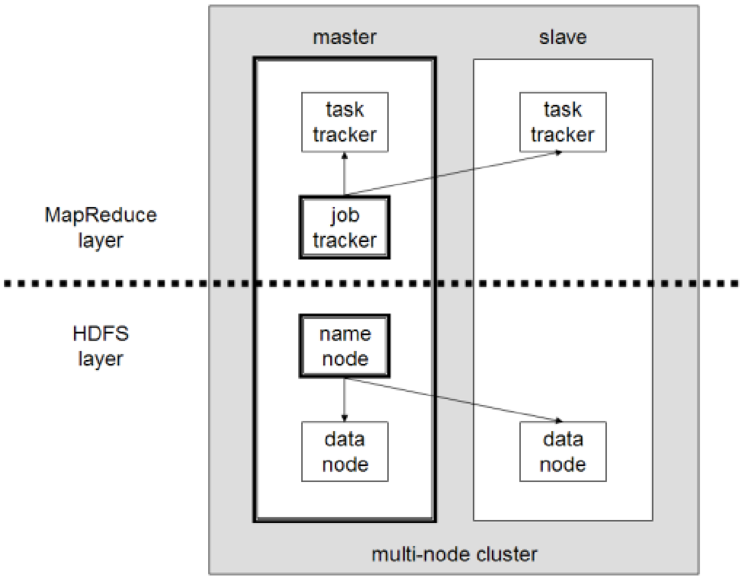


Figure 2.4: Architecture of HDFS [24]

Chapter 3

Experimental Environment

This thesis focuses on comparing performance and power consumption of ARM processors and Intel processors at various application domains. For building up ARM-base server cluster, four Pandaboards are used to set up a test platform. A workstation carrying Intel processor is deployed as its adversary. This chapter is about the configurations of these two test platforms and each one's power consumption metering methods.

3.1 Pandaboard Specifications

The ARM test platform is based on Pandaboard which includes one OMAP4430 processor, 1 GB low power DDR2 RAM and a 8 GB Kingston SD card. The OMAP4430, as the core logic unit of Pandaboard, contains Dual-core ARM Cortex-A9 MPCore with Symmetric Multiprocessing (SMP) at 1 GHz each, which allowing for 150% performance increase over previous ARM Cortex-A8 cores [33]. What's more, OMAP4430 also supports several image and video processing technology. Figure 3.1 depicts Pandaboard's layout and its periphery components.

3.2 Intel Workstation

As the counterpart of ARM test platform, we deploy a Intel processor based workstation as the control group. The processor inside the Intel workstation is Core2-Q9400 of Xeon series. This CPU is quad-core, thus this Intel CPU could run four processes simultaneously.

For reducing the difference between two platforms, the Intel workstation installs the same version of operating system and software stack to make sure

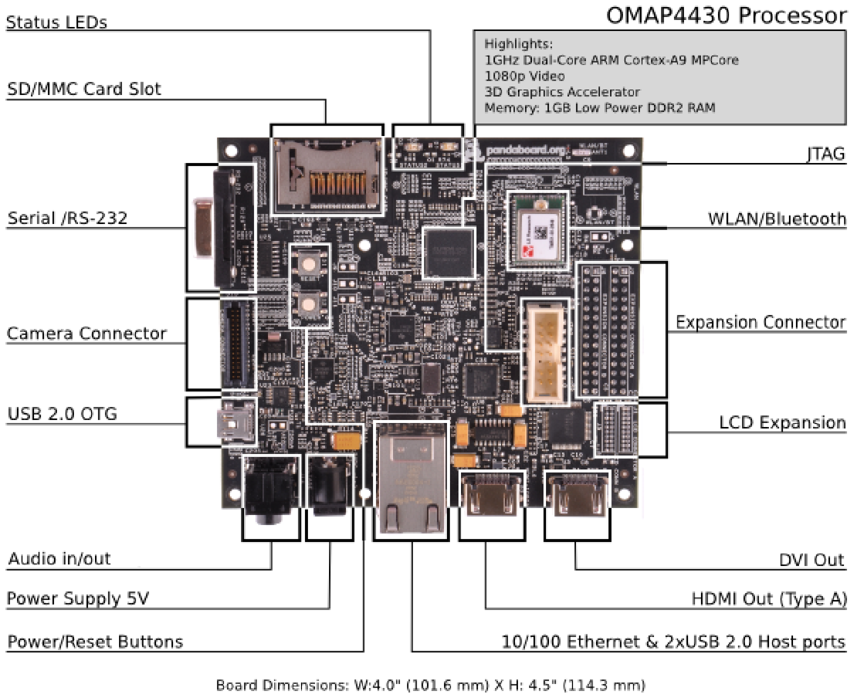


Figure 3.1: PandaBoard Platform [34]

both test platforms have the same application foundations. The configurations of two test beds, Pandaboard and Intel workstation, are compared in Table 3.1.

3.3 Test Methodology

For metering ARM platform, Mosoon Power Monitor [26] is used for measuring the power consumption. Figure 3.2 shows the power monitor and a self-made power supply plug compatible with Pandaboard's socket. This device is able to supply power to Pandaboard meanwhile recording power and reporting the average value in a real time fashion.

For Intel workstation power measurement, we use Mastech product: MS2102 AC/DC clamp meter. The MS2102 clamp meter is able to measure maximum 200A current with 2.5% accuracy. To isolate the power used by hard disk devices (HDD) and other periphery devices, the clamp meter attaches the 5V (red) and 12V (yellow) lines, which mainly supply CPU and its cooling fans power, to record the currents, as showed in Figure 3.3. And then

Name	**Pandaboard**	**Intel workstation**
CPU	ARM Cortex-A9 (45nm technology)	Intel Core2-Q9400 (45nm technology)
Core number	2	4
Clock speed each core	1 GHZ	2.66 GHZ
Memory	1G DDR2 RAM	8G DDR2
Storage	16GB SD card	248GB hard disk
Network card	10/100 Ethernet	10/100/1000 Ethernet
OS	Ubuntu Linux 2.6.35 omap4	Ubuntu Linux 2.6.35 Intel X86
Thermal design power	1.9 watts	95 watts

Table 3.1: Comparison between ARM and Intel

we multiply the currents measured with its corresponding line voltages and sum them up to get the final power. To increase the resolution of the clamp meter, as Dimitris mentioned in his paper [48], the power supply lines are wrapped around the clamp three times, and the final record is from the total sum divided by the number of loops.

To better understand the relationship between processor computing used capability and power consumption, Dstat [7], a Linux system monitoring software, is used to record the CPU usage ratio when test application is running. After the application is finished, the CPU usage ratio is calculated by averaging the records of CPU usage ratio of that particular application.

Furthermore to obtain convincing data, every test case is sampled over 60 seconds after its value is stable and there is a reboot between exhausting tests(test during which CPU usage ratio reached 100%) for cooling and resetting system. Each trial repeats three times (once in the morning, once in the afternoon and once in the evening) to ensure there is no circumstance factors (humidity and temperature) which might affect the final experiment results.

Figure 3.2: Mosoon power monitor with a self-made plug

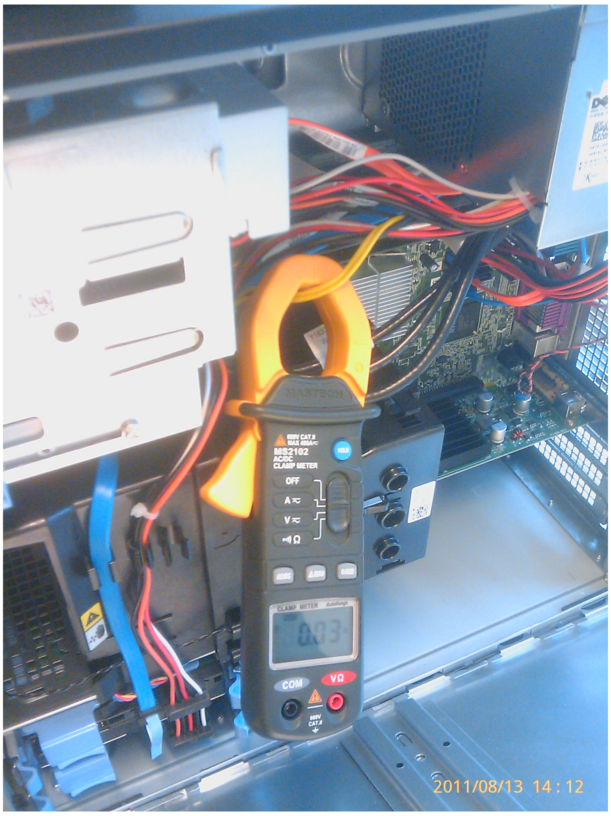


Figure 3.3: MS2102 AC/DC clamp meter

Chapter 4

Experimental Design

The experiments in this paper are about power efficiency, thus power consumption and performance of system will be measured and compared between Intel X86 processors and ARM processors in various scenarios. The purpose of the study is try to reach a conclusion that which processor gets a better power efficiency result in a particular application domain.

The experiments include several different scenarios, which contain some popular applications on server-side, will be explained in the following sections respectively.

First of all, a basic processor capability test is conducted using a hardware benchmark tool, i.e. LMbench [20]. This tool is able to use some basic processor operations to test the processor time spent on each kind of operations. The less time spent on the operations, the better capability the testing processor possesses. From this experiment, the basic performance difference between these two processors is clear. The following 4.1 shows the result of this benchmark.

From these data, it is obvious that the Intel processor makes a great advantage over ARM processor. In brief, Intel CPU executes simple operation, such as add operation, around four times faster than ARM processor. It dominates at some complex operations, for instance dividing operation conducted by Intel CPU is 14 times faster than ARM processor.

4.1 HTTP Server

Web is becoming the core application for e-business and Web server's performance is critical to end user experience, so it is an important performance index for servers. This experiment is to test power efficiency of Pandaboard and Intel workstation when they are acting as Web servers. For testing Web

Name	**Intel processor**	**ARM processor**
Integrate bit operation	0.3800	1.4000
Integrate add operation	0.1900	1.0200
Integrate multiple operation	0.1000	0.4600
Integrate dividing operation	7.7400	108.4
UINT64 bit operation	0.3800	1.4000
UINT64 add operation	0.1900	1.0200
UINT64 multiple operation	0.1000	0.4600
UINT64 dividing operation	7.7400	108.4
UINT64 mod operation	7.5600	23.4

Table 4.1: Comparison between ARM and Intel processors

server, Httperf [25], a Web benchmark testing software, is used on a client computer. At the server side, LVS is used as front-end load balancer and Nginx works as static resource web server, while httpd is adopted for dynamic Web pages requests testing.

The goal of this experiment is to reveal the EE of two platforms through revealing the relations of work load, power and Web server's handling HTTP requests capability.

The experiment is divided into two parts to examine the Web server performances of both static Web and dynamic Web. For testing dynamic Web performance, httpd and PHP5 are installed on each Pandaboard, and PHP scripts are deployed for dynamic Web page test, in which case every response HTML file is generated by the PHP script and a certain level of CPU computing is involved in. Three workload levels of test script are designed, each of them contains a word seeking task. The low, medium and high load level tasks are seeking one word in one hundred, one thousand and ten thousand words respectively, and each request reaches the server invoking the seeking task once and gets response with non-cached result.

In addition to dynamic Web page, another kind of Web server is for serving static resources including music files, video files, static Web pages and picture files. In modern Web service framework, this kind of server is always in front of dynamic Web page server that controls the business logic and generates response depending on request parameters. For measuring the static Web servers, some sample files are deployed on the server as test files, and the server responses a test file to the client. The test file sizes includes 1KB, 4KB, 10KB, 30KB, 50KB and 100KB, these files represent most sizes of static file (Logo picture file, CSS file and JavaScript file) elements on Web pages.

On the client side, a Httperf installed workstation with Ubuntu works as client computer. Httperf is a Web server benchmark tool which is able to generate Web browser's http requests and to check the response time for evaluating the Web server's paralleled requests handling capability. The test command is below:

httperf –server (hostname) –uri (url) –num-conn N (total connection number) –num-call Y (request number per connection) –rate X (connection number per second) –timeout T (timeout time in second)

This command instructs Httperf to build up N connections to the hostname/url in a speed of X connections per second, and Y requests in each connection. Thus the requests would be sent out at speed of X*Y per second. The last parameter is the timeout limit, after which the delayed responses would be labeled an error responses.

In this experiment, the configuration for Httperf command is based on practical usage experience [16]. Because most current browsers and Web servers support HTTP 1.1, which allows several requests using one keep-alive TCP connection, there should be normally more than one request in each connection. The power consumption difference between varieties of request number per connection is tested by using medium work load test case in various numbers of requests per connection. From the result depicted in Figure 4.1, the TCP connection overhead affects little on power consumption when more than 10 requests share one connection, thus we use 10 requests per connection as the parameter of Httperf to conduct this experiment.

8-seconds is used as the timeout parameter for dynamic page server, according to Zona Research's "8-second rule" [50], because normally a web page should be loaded no more than 15 seconds, after which half of users lose theirs patience and give up browsing, and sometimes a dynamic web page may require another dynamic page request (Iframe tag) to show the whole page, thus 8 second is a modest parameter for time-out setting. Moreover we use 15-seconds for static file tests, because some delays for picture and music files are acceptable when users browse web pages.

Some supportive result parameters, e.g. response number per second, request success ratio and network I/O traffic can be obtained from Httperf running on the testing workstation after tests finished. *When the request success ratio is below 100%, server reaches its saturation point stuck by CPU usage or network bandwidth.* The Network I/O is the average number of byte sent and received excluding the package headers through the TCP connection built up between Httperf and Web server, hence the number is less than the actual data traffic on the cable, because it does not account for the TCP header and the retransmissions overhead that may already occurred [25].

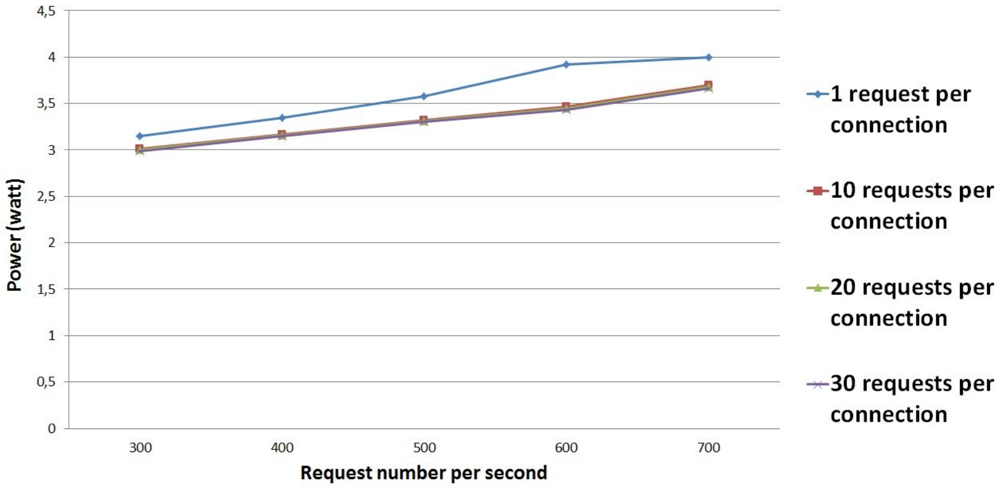


Figure 4.1: Connection affection on power consumption

A factor that may affect experiment results is the test platforms' stability. To test Pandaboard's stability, a stress test, consuming more than 90 percents of server ARM processor capability and lasting for just 3 minutes, repeats for 10 times to check whether the records of 10 times test vary a lot. Figure 4.2 shows that variance of power consumption is less than 2%. Thus the Pandaboard is stable enough to get reliable data even it has to run exhausting tasks for more than 30 minutes.

Thanks to HTTP protocol's stateless feature, a web server cluster with poor hardware may catch a chance to defeat a single top Web server with fancy hardware. Hence after finishing experiment on a single Pandaboard, four Pandaboards with an extra load balancer machine sets up a Web server cluster for testing the ARM-base servers' scalability. The load balancer machine uses LVS as the load balancer software, distributing HTTP requests fairly to individual Web servers in the same local network. Figure 4.3 reveals the architecture of Pandaboard cluster structure.

Last, to assure all these four Pandaboards have similar general capability on executing tasks, we make a stress test on these four boards. From the result in Table 4.2, there is no huge difference between these boards' power consumption when running the same tasks, but the board number 2 and number 4 have around 10% advantage at performance.

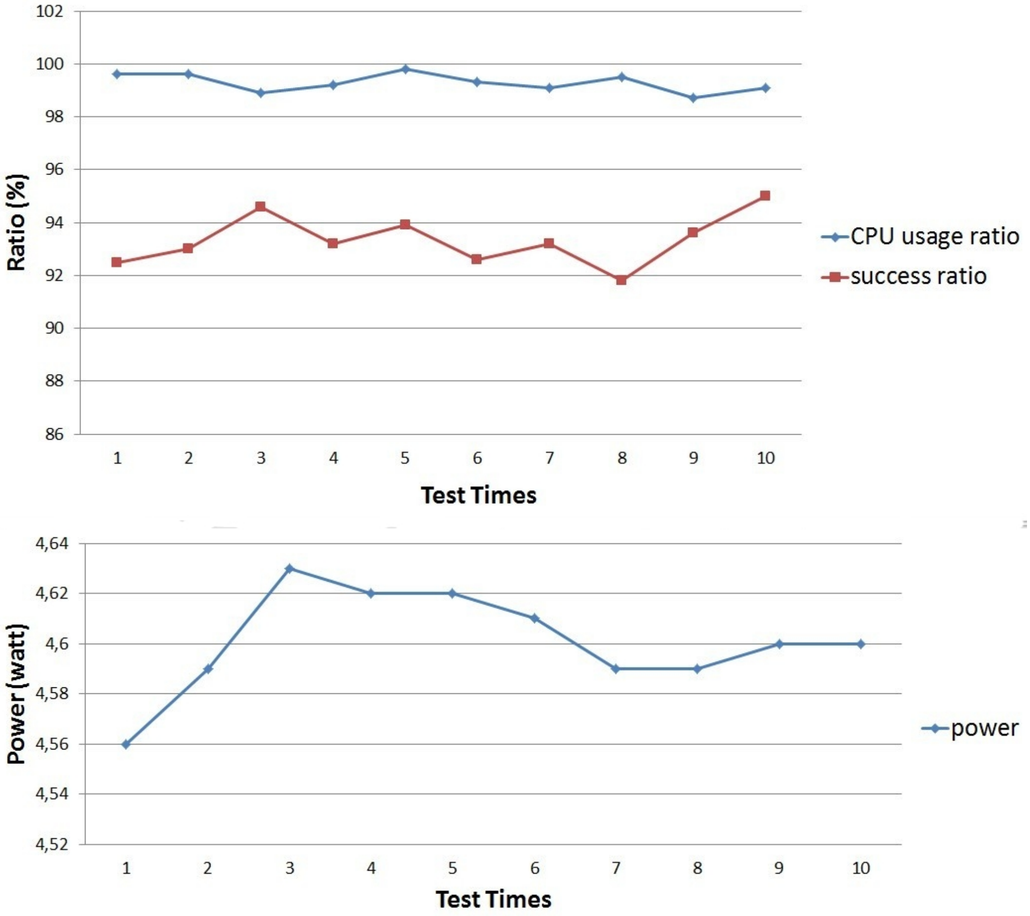


Figure 4.2: 10 times stress test on Pandaboard

4.2 In-memory Database

As the foundation of modern e-commerce and e-business solution, Databases are undertaking increasingly stricter application requirements: the data-centered applications require faster response, higher throughput and allowing bigger number of concurrent access. Thus databases need high performance CPU to meet those requirements. This experiment is targeting database's energy efficiency. Since the Pandaboard platform does not contain hard disk, this experiment only measures in-memory database to exclude the performance difference between hard-disk and SD card. We choose SQLite 3.07 [36] as the testing application. SQLite is a popular memory database, of which the biggest advantage is simplicity. Although the SQLite is normally used

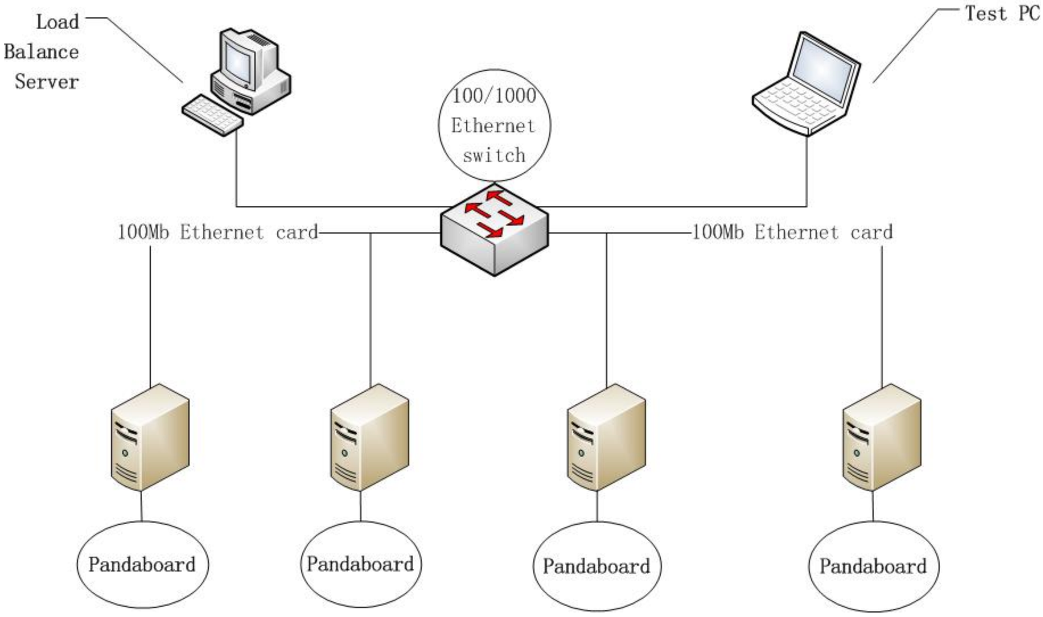


Figure 4.3: Pandaboard cluster structure

Board number	**1**	**2**	**3**	**4**
Avg power (Watt)	4.49	4.54	4.49	4.46
CPU load (%)	98.8	99	99.1	98.6
Success ratio (%)	100	100	100	100
Network I/O (KB/s)	93.4	102.1	95.8	105.6

Table 4.2: Difference between each board

in embedded system and small or middle websites, with partition and Shard technologies getting mature, the SQLite databases cluster may outperform a single industrial database server. In this experiment, SQLite is installed on both Pandaboard and Intel server to compare performance/power consumption difference between them. The benchmark test is derived from official benchmark methods on SQLite website. This benchmark includes common operations, but does not measure multi-user performance or optimization of complex queries involving multiple joins and subqueries [37]. The experiment includes six test cases, which test the basic operations of adding new records, building index, updating record, scanning records and deleting records. All SQL queries in one test case are organized in one SQL transaction to optimize the execution time. The detail of experiment is listed as follows:

- 10000 entries insert;
- 5000 times full table scan with string comparison;
- set up an index on the table;
- 2000 times update with full table string comparison;
- 5000 times insert from a result of full table scan;
- 10000 records delete with full table string comparison;

4.3 Video Transcoding

Nowadays, video websites, such as YouTube, occupy a large number of Internet entertainment market share. These kinds of websites always provide functions that allow users to upload various types of video and transcodes them into one standard encoded format, i.e. flv [9]. Hence in video transmission area, ARM based server should be tested as well. Although a single Pandaboard is slow at multi-media computing speed, when several ARM processors transferring video clips concurrently, the total speed could be comparable to one Intel server. The experiment conducted by Rafael Pereira proved the total encoding duration can be dramatically reduced, if the multi-media file is divided into pieces and processed in parallel. In his experiment, he used Amazon EC2 cloud servers to build a platform which splits a high-definition video into pieces and executes encoding process and merges them together into one encoded video. The total time this platform used is less than 10% of the total time spent on traditional way. Thus the split& merge method can accelerate the ARM-based server's video processing speed in the same way to make the ARM-based server equivalent to Intel X86 server in video process capability [35] [10].

Before conducting the experiment, a performance comparison is made using a video transcoding benchmark, HD-VideoBench [22]. This benchmark is aiming at comparing different codec high definition video digital video processing efficiency. It contains three sets of video encoders and decoders , so-called codecs, for the MPEG-2, MPEG-4 and H.264/AVC video coding standards, and a series of videos, in different sizes, that test the performance of these three sets of codec. But in this paper, HD-BideoBench is used to compare the performance between Intel workstation and Pandaboard by running the same codec and the same test video.

The H.264 codec of this benchmark is selected to be a test engine to compare video processing capability between these two platforms. From Figure 4.4, it is clear that the ratio of the time ARM processor spent to Intel

CPU spent is from 7 to 14. The fact that the ratio trough occurs at the riverbed sequence is because this sequence is a special limited video which is hard to be coded when motion compensation algorithms can not work in this case. Thus normally the Intel CPU is 12-14 times as fast as ARM-based platform at video process area.

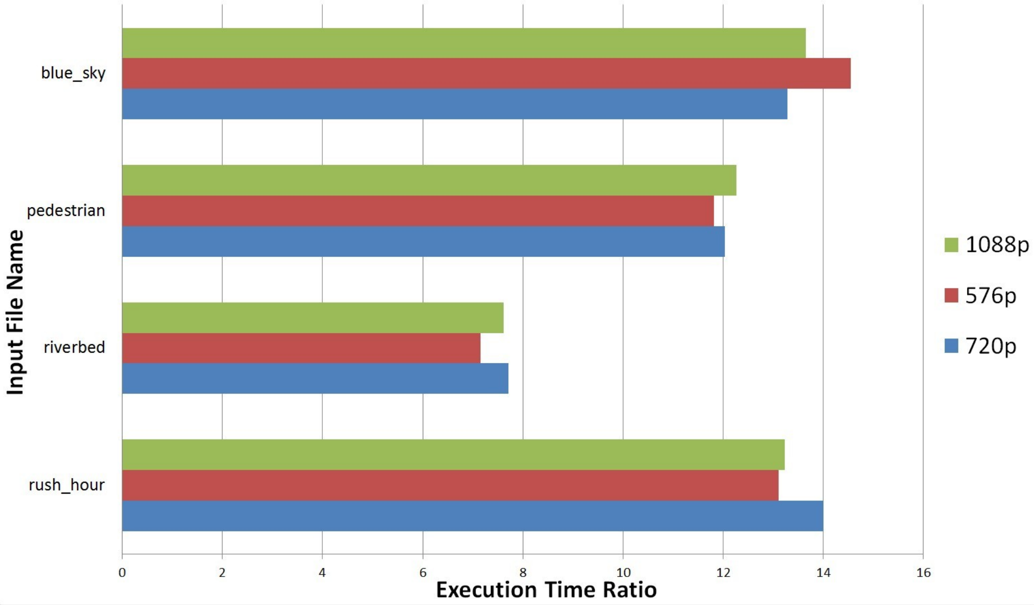


Figure 4.4: Result of HD-VideoBench

In this experiment, an original high quality file is evenly split by the video process software MEncoder into eight parts. The original test video is an 86 MB AVI file encoded by H.264 / AVC in 24 frame/sec with the resolution of 1920 x 1080. The splitting command is 'MEncoder -ss starttime -endpos endtime -oac copy -ovc copy inputfile.avi -o outputfile.avi'. After splitting, each testing video clip is exactly one eighth of the whole video file. And then codec software FFmpeg is used to transcode the testing video clip from AVI format to FLV format and also compress its resolution from 1920 x 1080 to 640 x 480. The command that transfers the test video is following: FFmpeg -i test.mov -ab 64000 -ar 44100 -f flv -s 640*480 -r 25 output.flv For fully exploiting processor's computing capability, more than one FFmpeg transcoding processes are running on ARM processor and Intel workstation in parallel, because FFmpeg has not been optimized for multi-core processor, of which one core is occupied by one process. Hence two processes are running on ARM processor and Intel CPU keeps four processes at the same time. The time spent on video files transferring is recorded and compared between two platforms.

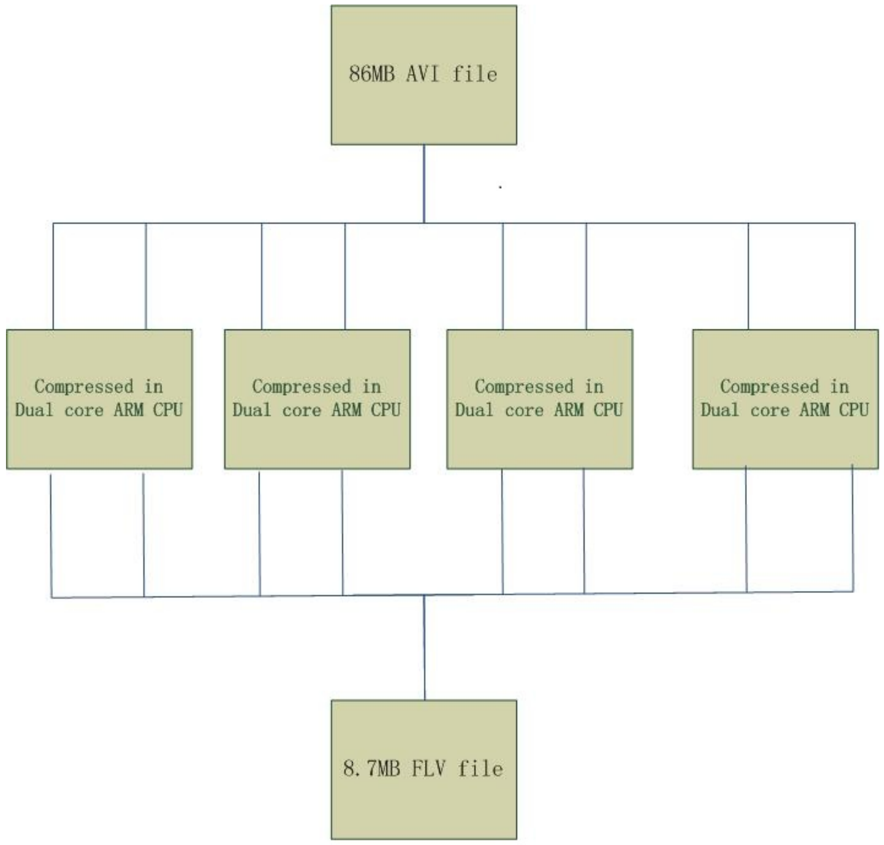


Figure 4.5: Pandaboard video transferring experiment structure

Figure 4.5 shows this experiment's process. One 86MB AVI video file is split into eight video clips, then each two of them are transferred and compressed at one Pandaboard and four Pandaboard with eight cores works simultaneously to output 8 FLV files, and then those 8 files are merged by Mencoder back to a 8.7MB FLV file.

4.4 Hadoop Map-Reduce Task

For Hadoop part, an official benchmark program is adopted to test the whole map-reduce system performance by running a Hadoop map-reduce tasks [6]. The benchmark program is packed in the Example jar file shipped with

Hadoop release (version 2.023). The benchmark is comprised of three phases: data generation, data sorting and data validating. First phrase is to generate N(N is multiple of 64) MB random data on each data node by invoking RandomWriter class, and the command is as following: "hadoop jar hadoop-*examples.jar randomwriter random-data" The "random-data" is the output file for the random data. And in the second phrase all the random data will be sorted in order by map-reduce task invoked by command: "hadoop jar hadoop*examples.jar sort random-data sorted-data", and the "sorted-data" is the output file for sorted data. The last phrase is validating the result. A program packed in the Test jar packet is aiming to commit this work. The command, "hadoop jar hadoop*test.jar testmapredsort -sortInput random-data -sortOutput sorted-data", tests the result of data sorting process is correct or not. Four Pandaboards are set up as a Hadoop cluster, in which each board works as a Data node and also a task tracker, and the Name node and job tracker are overtaken by a extra computer outside the cluster. The Intel workstation works as a Hadoop cluster itself, and one extra Name node and job tracker manages its tasks execution.

Chapter 5

Result Analysis

The previous Chapter describes the details of experimental design. This Chapter consists of four sections, each of them analyses the result of one sub-experiment and evaluates those results by EE index put forward in Chapter 1. Because we can not measure ARM processor's power accurately, the total Pandaboard power is roughly seem as ARM processor's power, but ARM processor's power should be much lower than the Pandaboard's power, because the Cortex-A9 CPU's nominal power is only 1.9 watts. Thus the real ARM processor's EE should be higher than the EE we measured in these experiments.

5.1 HTTP Server Experiments

This section discusses the results of 27 Web tests in groups of different test cases to compare the power consumption between Intel workstation and single Pandaboard and Pandaboard cluster.

The first comparison is about the low workload Web request tests, as mentioned in section 4.1. From Figure 5.1, Pandboard's power increases linearly with CPU usage from 3.11 watts to 3.96 watts. The saturation point appears at 1400 requests/sec due to the CPU usage saturation. Figure 5.2 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is around 25 watts for 2000 requests/sec and 45.5 watts for 7000 requests/sec while the CPU usage rises from 15.6% to 42.4%. Figure 5.3 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is around 5000 requests/sec due to the CPU usage saturation.

The second comparison is about the medium workload Web request tests. From Figure 5.4 Pandboard's power increases linearly with CPU usage from

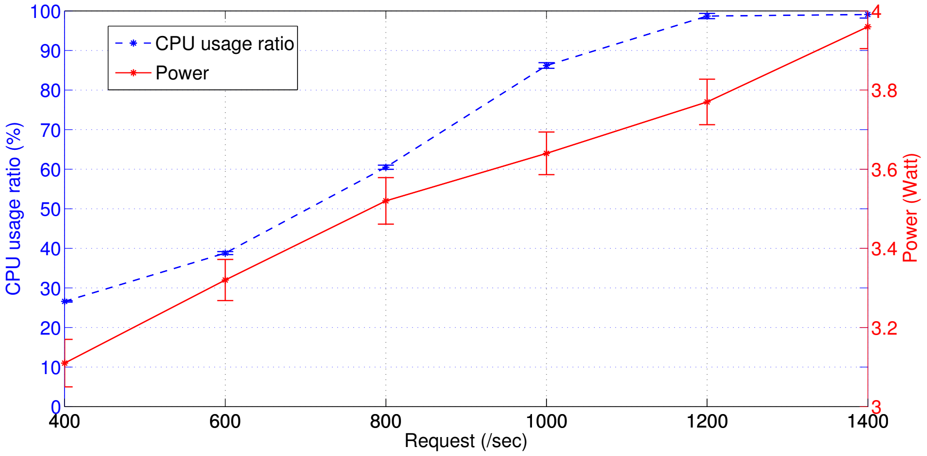


Figure 5.1: Low workload Web test on ARM processor

2.95 watts to 4.19 watts and the saturation point appears between 800 and 1000 requests/sec due to the CPU usage saturation. Figure 5.5 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 27.5 watts for 2000 requests/sec to 53.9 watts for 7000 requests/sec while the CPU usage rises from 21.0% to 60.0%. Figure 5.6 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is between 3000 and 4000 requests/sec due to the CPU usage saturation.

The third comparison is about the high workload Web request tests. From Figure 5.7 Pandboard's power increases linearly with CPU usage from 3.28 watts to 4.57 watts and the saturation point appears between 300 and 350 requests/sec due to the CPU usage saturation. Figure 5.8 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 27.8 watts for 500 requests/sec to 74.5 watts for 3500 requests/sec while the CPU usage rises from 18.8% to 99.9%, and the saturation point is around 3500 requests/sec due to the CPU usage saturation. Figure 5.9 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is between 1000 and 1500 requests/sec due to the CPU usage saturation.

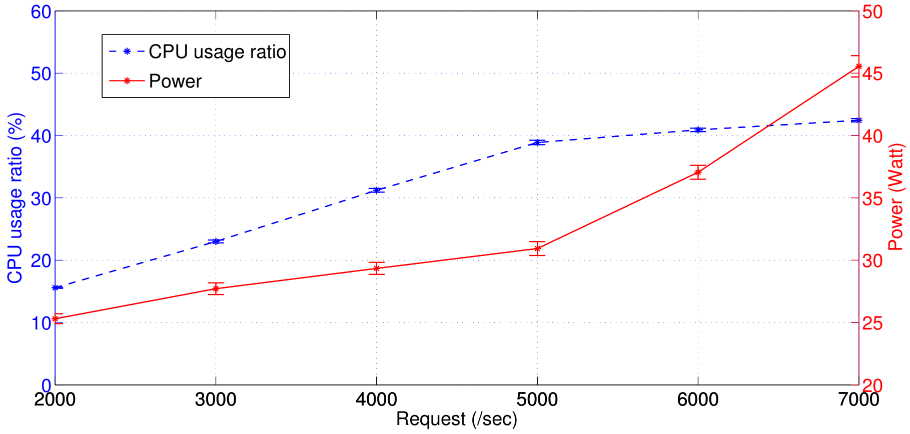


Figure 5.2: Low workload Web test on Intel CPU

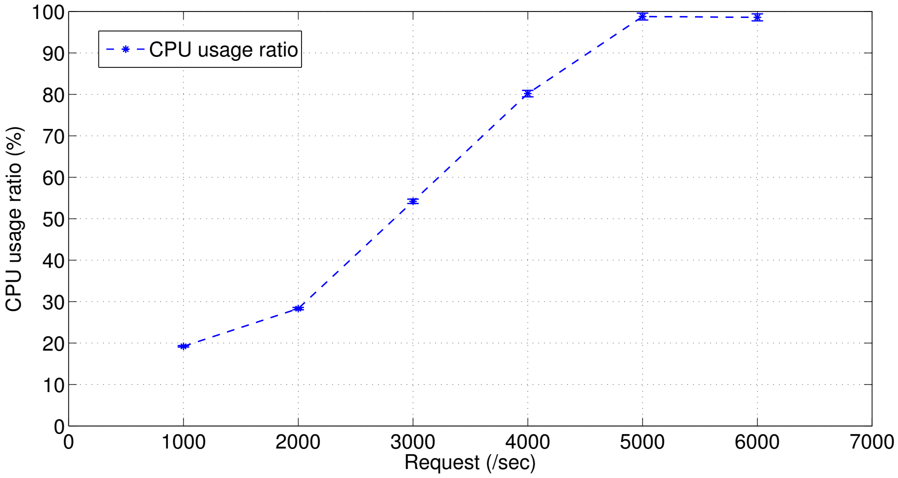


Figure 5.3: Low workload Web test on ARM-based cluster

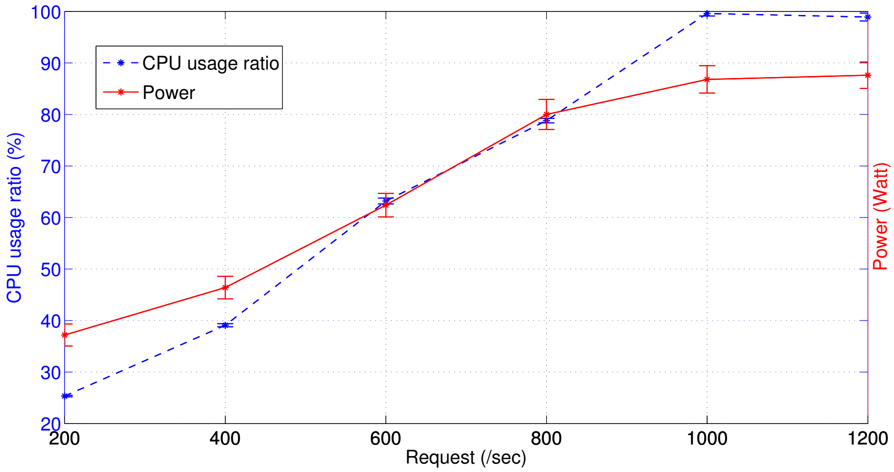


Figure 5.4: Medium workload Web test on ARM processor

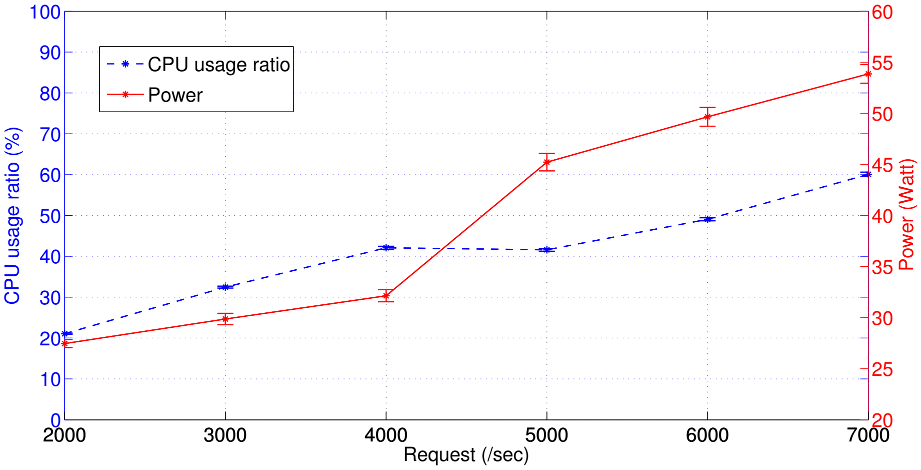


Figure 5.5: Medium workload Web test on Intel CPU

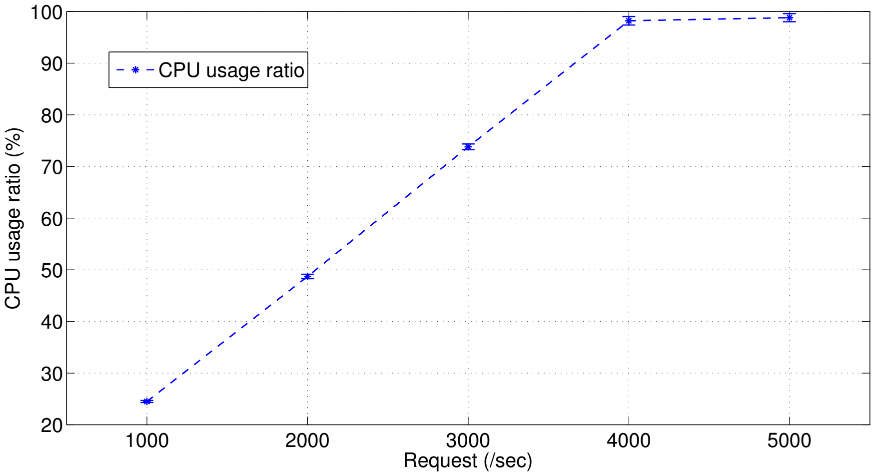


Figure 5.6: Medium workload Web test on ARM-based cluster

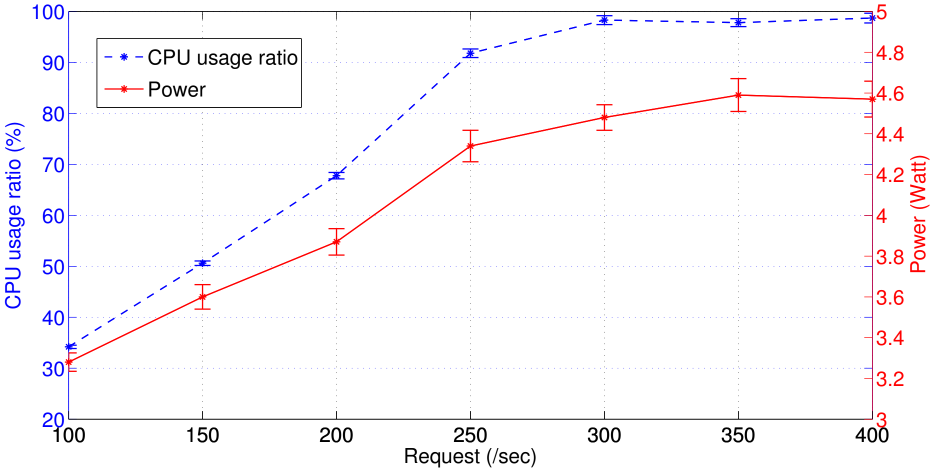


Figure 5.7: High workload Web test on ARM processor

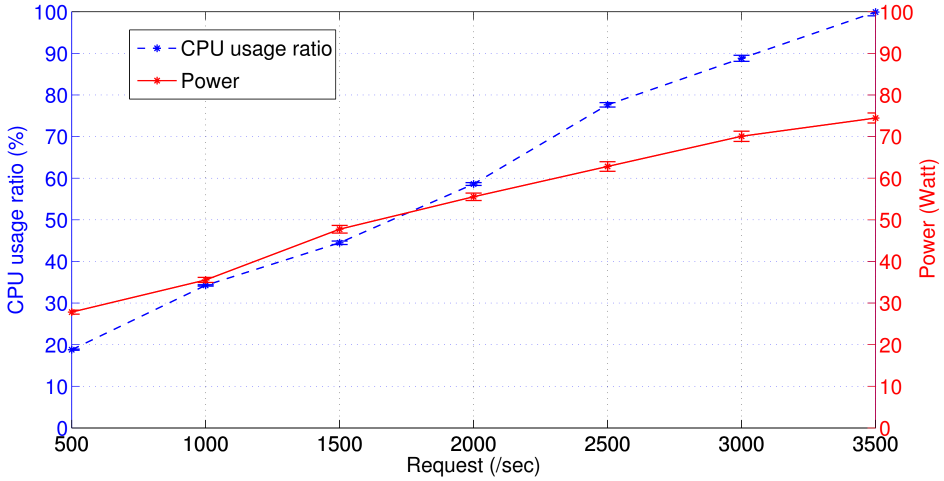


Figure 5.8: High workload Web test on Intel CPU

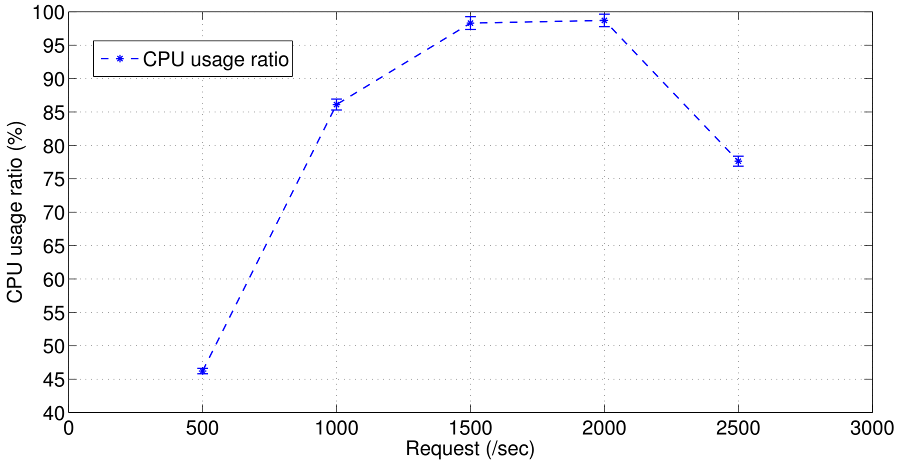


Figure 5.9: High workload Web test on ARM-based cluster

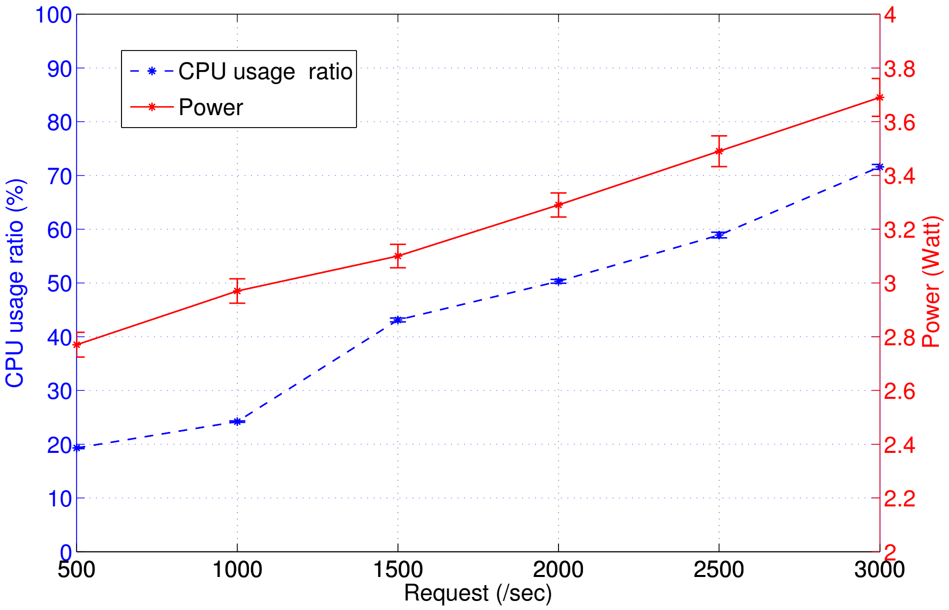


Figure 5.10: 1KB file fetch test on ARM processor

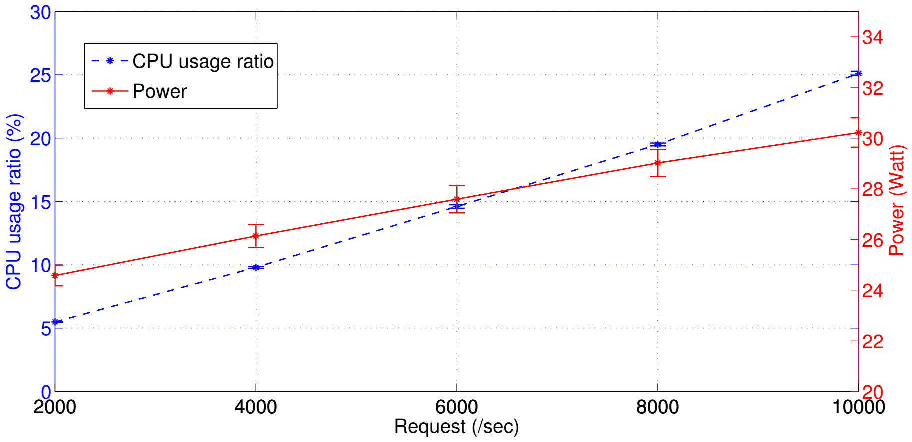


Figure 5.11: 1KB file fetch test on Intel CPU

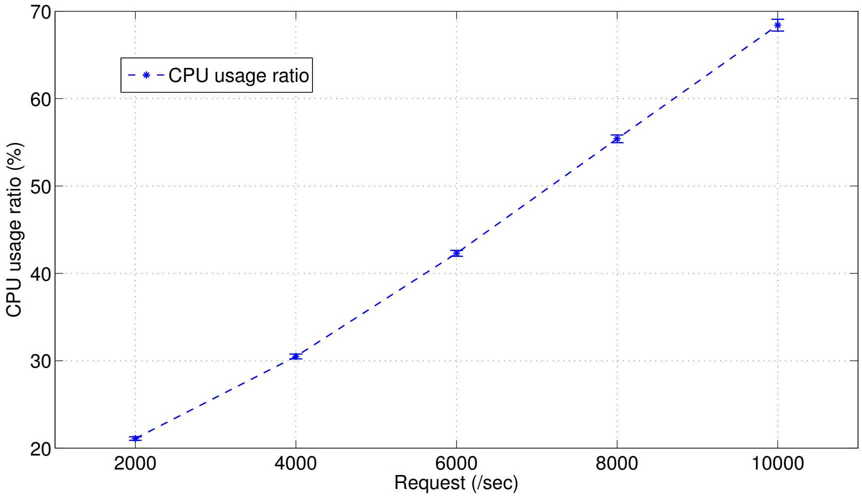


Figure 5.12: 1KB file fetch test on ARM processor server cluster

The fourth comparison is about the 1KB static file fetch request test. From Figure 5.10 Pandboard's power increases linearly with CPU usage from 2.77 watts to 3.69 watts due to the client testing machine's limitation. Figure 5.11 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 24.58 watts for 2000 requests/sec to 30.22 watts for 10000 requests/sec while the CPU usage rises from 5.5% to 25.1%. Figure 5.12 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards.

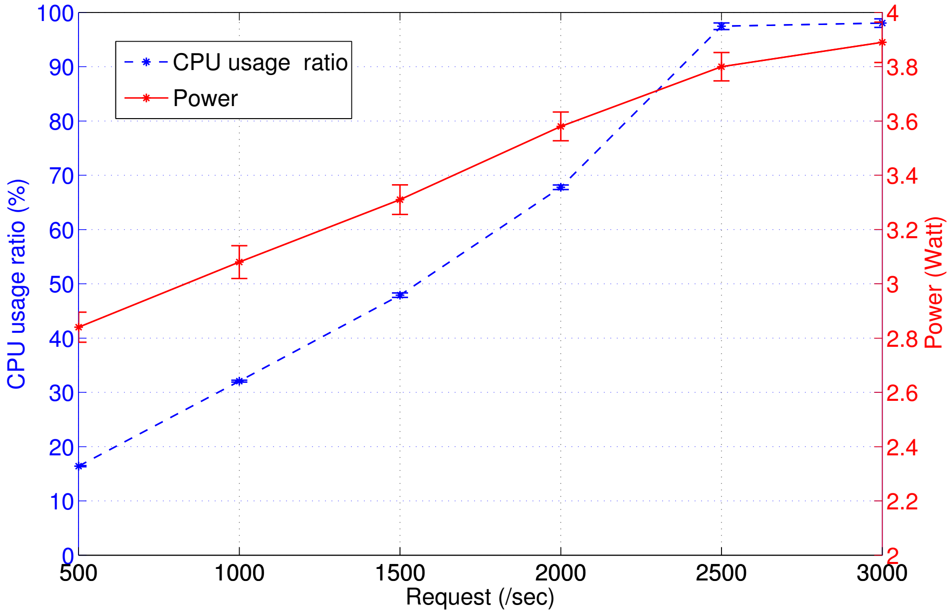


Figure 5.13: 4KB file fetch test on ARM processor

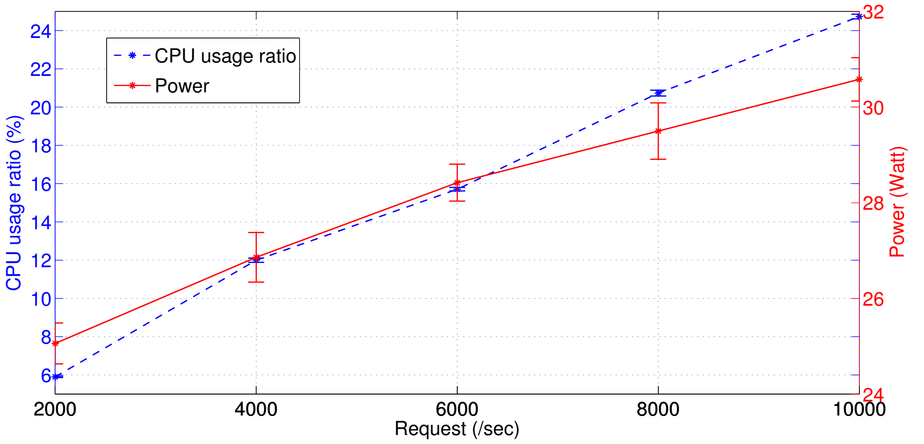


Figure 5.14: 4KB file fetch test on Intel CPU

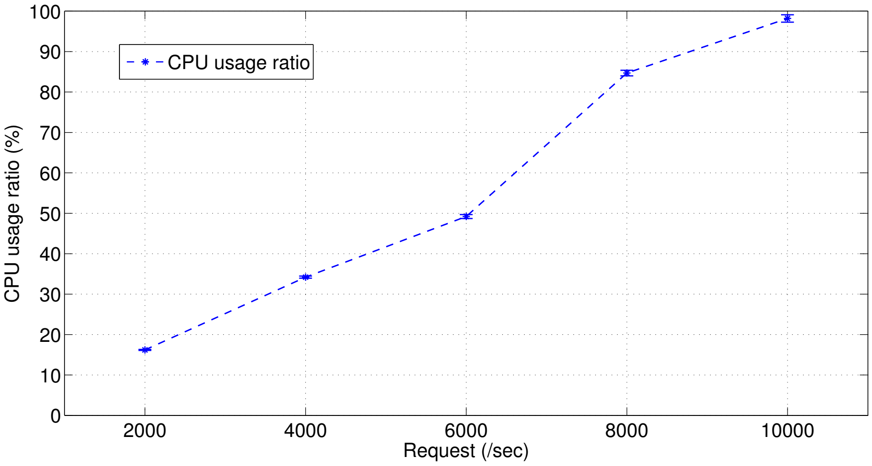


Figure 5.15: 4KB file fetch test on ARM processor server cluster

The fifth comparison is about the 4KB static file fetch request test. From Figure 5.13 Pandboard's power increases linearly with CPU usage from 2.84 watts to 3.89 watts and the saturation point appears around 2500 requests/sec due to the CPU usage saturation. Figure 5.14 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 25.1 watts for 2000 requests/sec to 30.6 watts for 10000 requests/sec while the CPU usage rises from 5.9% to 24.7%. Figure 5.15 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is around 10000 requests/sec due to the CPU usage saturation.

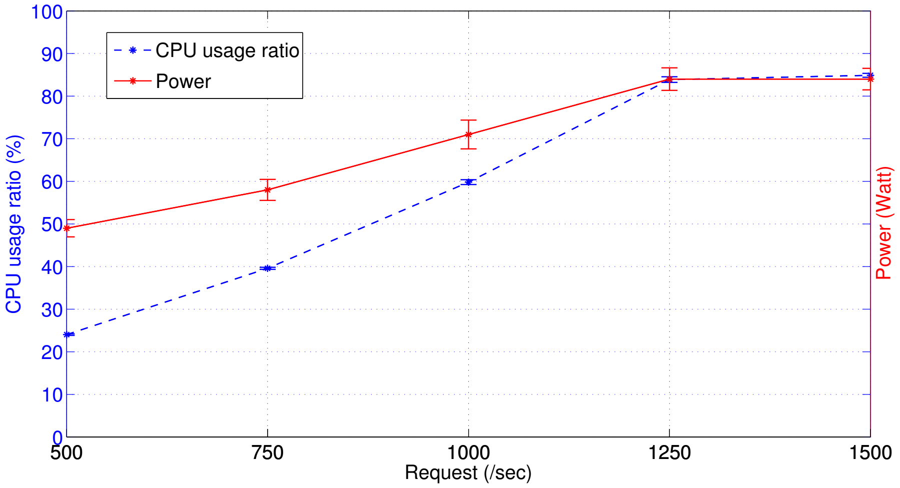


Figure 5.16: 10KB file fetch test on ARM processor

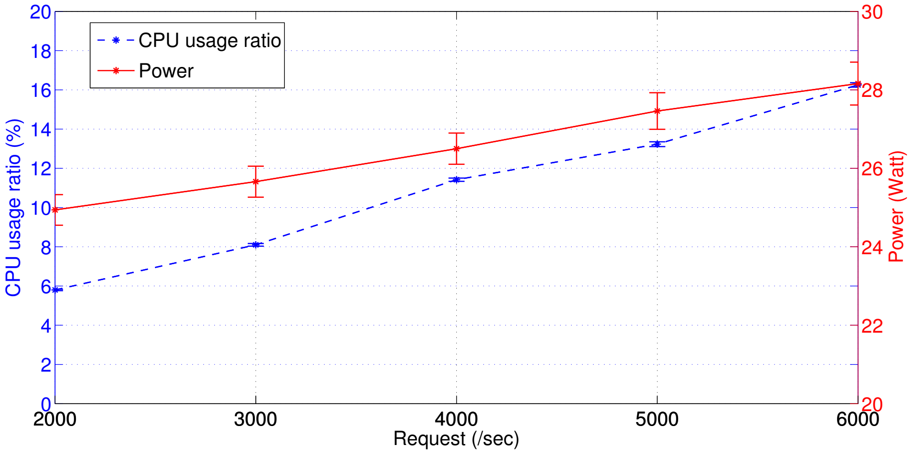


Figure 5.17: 10KB file fetch test on Intel CPU

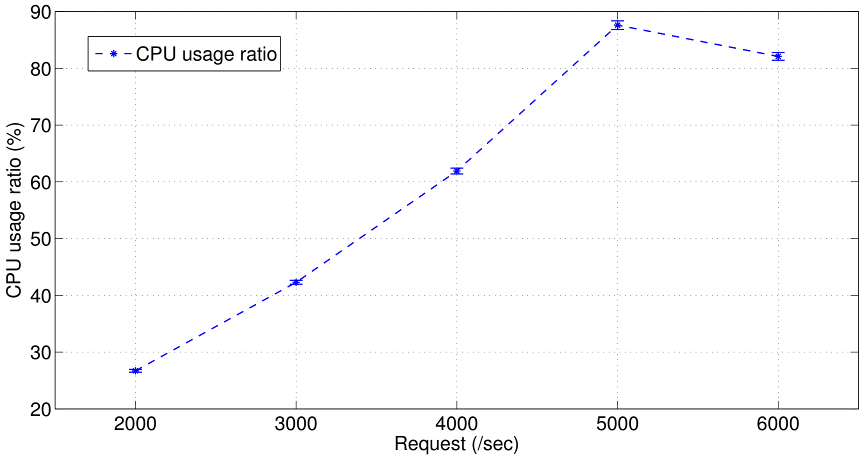


Figure 5.18: 10KB file fetch test on ARM processor server cluster

The sixth comparison is about the 10KB static file fetch request test. From Figure 5.16 Pandboard's power increases linearly with CPU usage from 2.98 watts to 3.68 watts and the saturation point appears around 1500 requests/sec due to the network bandwidth saturation (100 Mb/sec). Figure 5.17 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 24.9 watts for 2000 requests/sec to 28.2 watts for 6000 requests/sec while the CPU usage rises from 5.8% to 16.25%. Figure 5.18 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is around 6000 requests/sec due to the network bandwidth saturation.

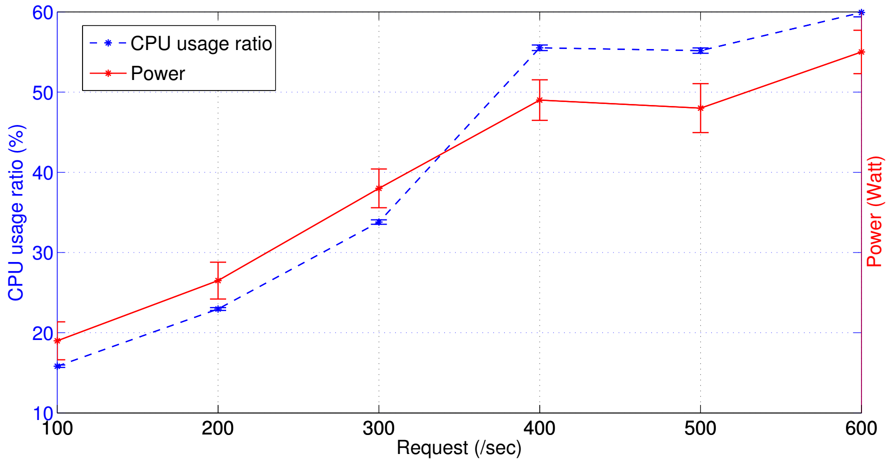


Figure 5.19: 30KB file fetch test on ARM processor

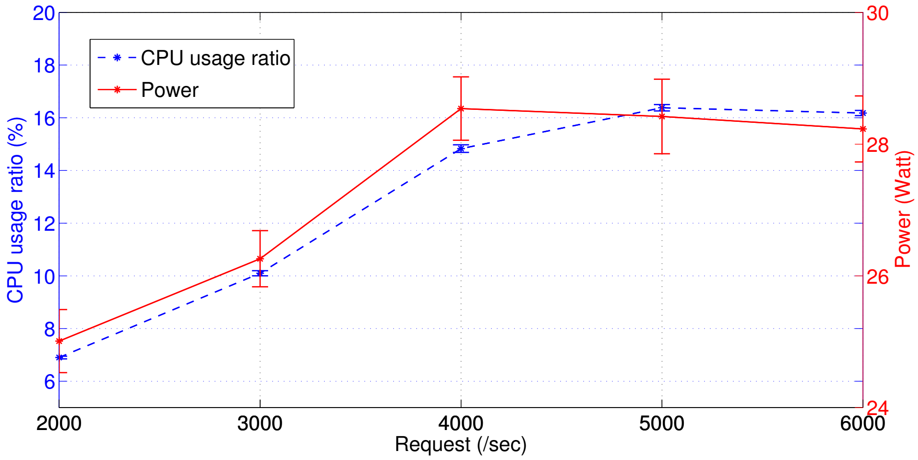


Figure 5.20: 30KB file fetch test on Intel CPU

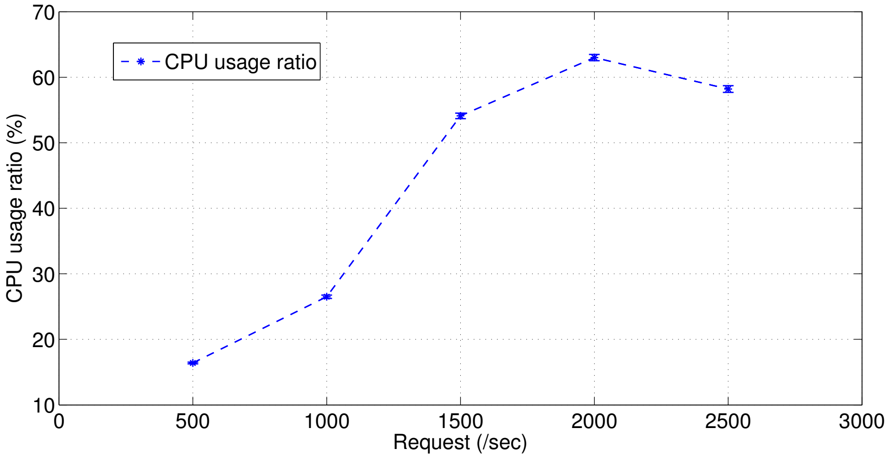


Figure 5.21: 30KB file fetch test on ARM processor server cluster

The seventh comparison is about the 30KB static file fetch request test. From Figure 5.19 Pandboard's power increases linearly with CPU usage from 2.78 watts to 3.5 watts and the saturation point appears around 600 requests/sec due to the network bandwidth saturation. Figure 5.20 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 25.1 watts for 2000 requests/sec to 28.8 watts for 5000 requests/sec while the CPU usage rises from 6.9% to 16.4% and the saturation point is around 5000 requests/sec due to the network bandwidth saturation. Figure 5.21 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is around 2500 requests/sec due to the network bandwidth saturation.

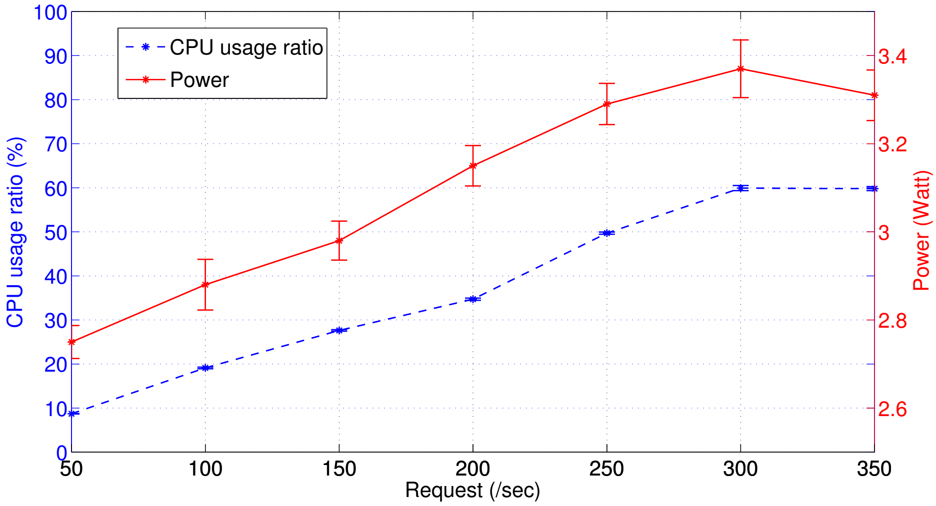


Figure 5.22: 50KB file fetch test on ARM processor

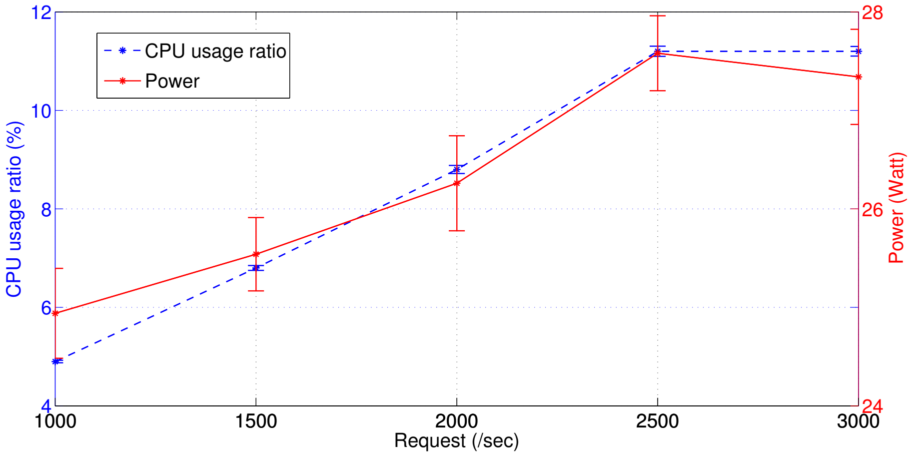


Figure 5.23: 50KB file fetch test on Intel CPU

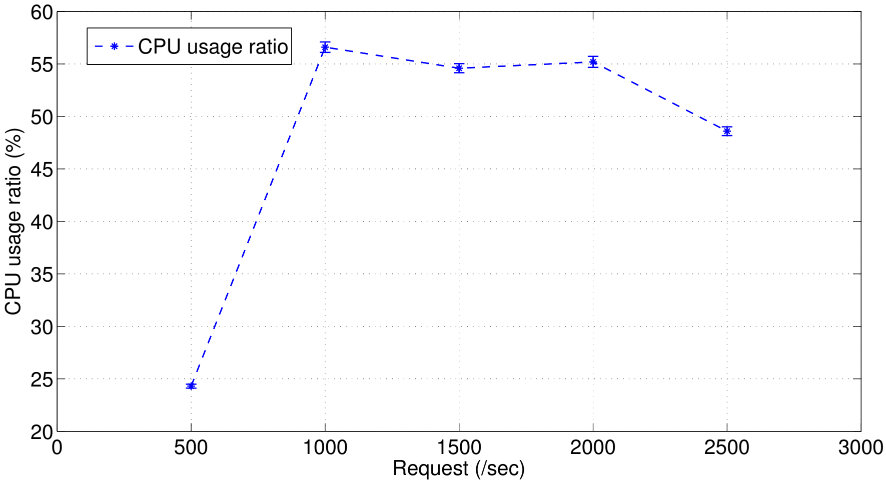


Figure 5.24: 50KB file fetch test on ARM processor server cluster

The eighth comparison is about the 50KB static file fetch request test. From Figure 5.22 Pandboard's power increases linearly with CPU usage from 2.75 watts to 3.37 watts and the saturation point appears around 350 requests/sec due to the network bandwidth saturation. Figure 5.23 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 24.9 watts for 1000 requests/sec to 27.3 watts for 3000 requests/sec while the CPU usage rises from 4.9% to 11.2% and the saturation point is around 2500 requests/sec due to the network bandwidth saturation. Figure 5.24 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is around 1500 requests/sec due to the network bandwidth saturation.

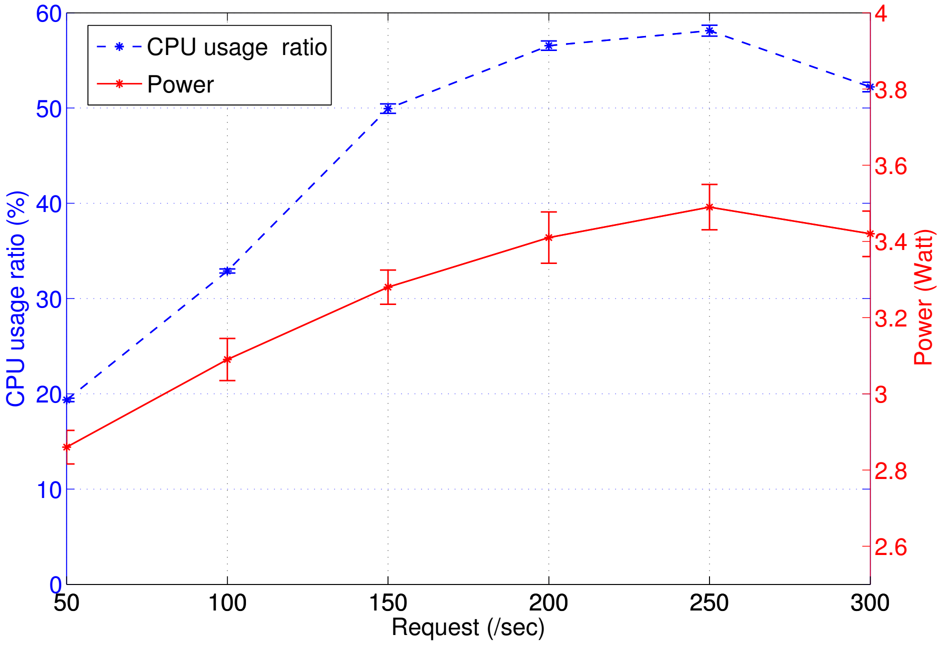


Figure 5.25: 100KB file fetch test on ARM processor

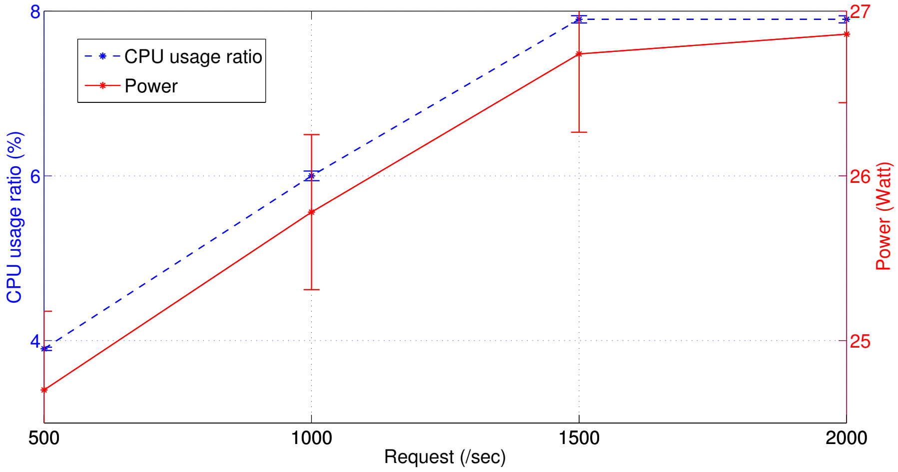


Figure 5.26: 100KB file fetch test on Intel CPU

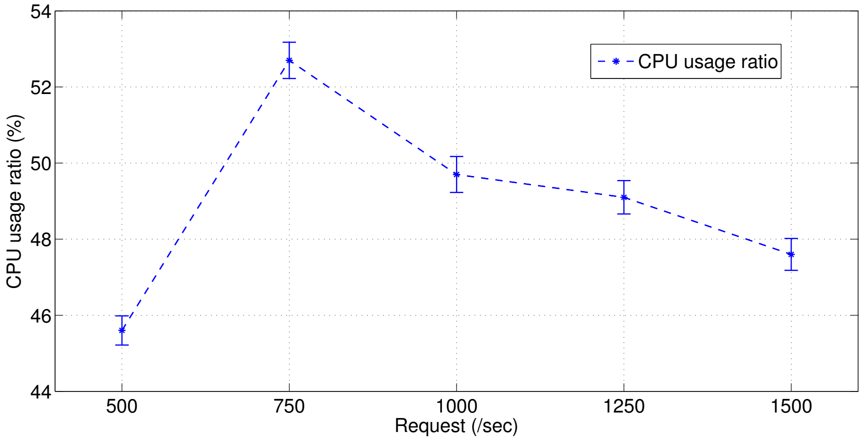


Figure 5.27: 100KB file fetch test on ARM processor server cluster

The last but not the least comparison is about the 100KB static file fetch request test. From Figure 5.25 Pandboard's power increases linearly with CPU usage from 2.86 watts to 3.48 watts and the saturation point appears around 250 requests/sec due to the network bandwidth saturation. Figure 5.26 demonstrates that the Intel workstation's power consumption is also following the CPU usage rising, and the power is from around 24.7 watts for 500 requests/sec to 26.8 watts for 2000 requests/sec while the CPU usage rises from 3.9% to 7.9% and the saturation point is around 1500 requests/sec due to the network bandwidth saturation. Figure 5.27 demonstrates that the performance of the cluster is close to the sum of four single Pandaboards and the saturation point is around 1000 requests/sec due to the network bandwidth saturation.

The performance is the number of request processed by both platforms per second and the power is the power recorded for Pandaboard and Intel CPU. Since the cluster performance is almost the sum of four single Pandaboards' performance and each Pandaboard consumes similar power from the Table 4.2, the EE of Pandaboard is approximately equal to EE of single Pandaboard, additionally, we use cluster saturation point to compare EE between Pandaboard and Intel CPU.

The following Table 5.1 shows the EE index of each experiment. The first column is the test case's name, and the second column is the number of testing requests per second, which is normally the saturation point of the Pandaboard cluster in the corresponding test. Moreover the third and the fourth column are EE index of Pandaboard cluster and EE index of

Test case	**Requests/sec**	**EE of Pandaboard cluster**	**EE of Intel CPU**	**Ratio of EE**
Low work load	5000	329	161	2.0
Medium work load	3600	226	115	2.0
High work load	1200	67.2	29.8	2.2
1KB	10000	714	331	2.2
4KB	10000	658	327	2.0
10KB	6000	407	213	1.9
30KB	2400	171	93.7	1.8
50KB	1400	106	56.8	1.9
100KB	1000	71.6	38.7	1.9

Table 5.1: Web test results

Intel processor respectively, while the last column is the ratio of the EE of Pandaboard cluster to the EE of Intel CPU. In this table, the ratios of EE are around 2 in all tests, thus the Pandaboard cluster is as two times energy efficient as the Intel processor in Web application domain.

From this experiment, it is clear that ARM processor energy efficiency is at least double of Intel CPU. But four-Pandaboard cluster's capabilities, including network bandwidth and computing capability, are still lower than a single Intel workstation. However relying on Web structure's high scalability, with a high performance load balancer, ARM processor based server cluster is possible to provide the same level Web service as Intel server cluster in double energy efficiency.

5.2 In-memory Database Experiment

The in-memory database experiment demonstrates the SQL engine speed and power consumption on both platforms. From Table 5.2, the speed of Intel workstation is approximately 4 times as fast as Pandaboard on Create table and insert test, Update test, Index test and Insert from a select result compound test, additionally, in Delete test Intel workstation is 4 times faster than Pandaboard. However, in the Full scan test, the Pandaboard is at the same level as Intel workstation which is only 30% faster than Pandaboard. To conclude, ratio of power of Intel workstation to Pandaboard's is around 11.4 in all tests, except of Create table and insert query test and Index query test,

Test cases	**Intel executing time (ms)**	**Intel power consumption(w)**	**ARM executing(ms)**	**power consumption(w)**
Create table and insert	128	n/a	532	n/a
Full table scan	88270	40.13	119345	3.56
Index	37	n/a	140	n/a
Update	14421	39.35	60244	3.50
Insert from a select result	12258	39.22	51389	3.47
Delete	8732	38.98	43231	3.48

Table 5.2: In-memory database experiment result

test cases	**Intel EE**	**ARM EE**	**Ratio ARM/Intel**
Full table scan	283	2354	8.3
Update	1762	4743	2.7
Insert from a select result	2080	5608	2.7
Delete	2938	6647	2.3

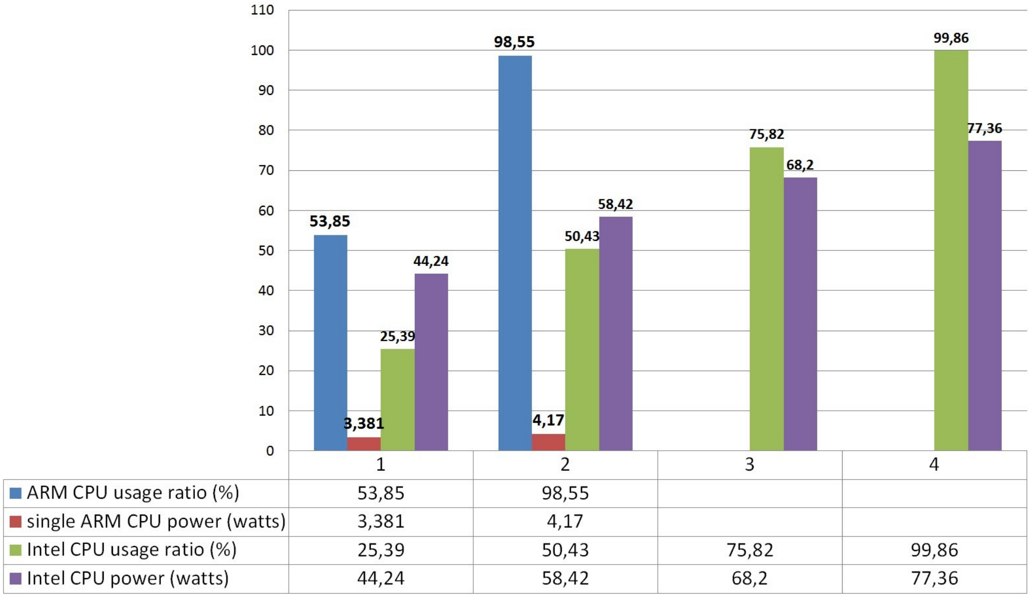
Table 5.3: In-memory database's energy efficiency

of which the executing time is too short to record power in our experimental environment.

The in-memory database experiment contains 6 query tests, but only four tests last more than 1 second, the other two queries' power consumption is negligible. Since both two platforms conduct the same tasks, we define the Work done as 100,000,000 units, and the energy is power multiplying executing time. The table 5.3 shows the EEs for each test and EE ratios of ARM processor to Intel CPU. In this table, the write operations(update) EE ratio are around 2.7 and read operation(scan) is 8.3, for most real cases, the in-memory database is used for reading data speedily and the executing time of full table scan is not much on ARM processor comparing with Intel CPU, thus ARM processor has a big advantage at this area. For further optimizing the executing speed, ARM processor could adopt partition technology, in which each ARM processor only processes one physical subtable from a big logic table, and the total speed would increase fast.

5.3 Video Transcoding Experiment

This section discusses the video transcoding experiment. In this experiment, each Pandaboard processes two 10MB video clips and the Intel workstation processes four 87MB videos. During the test we find both platforms finish their tasks in around 100 seconds, thus the processing speeds of two platform are almost the same and the ratio of throughput is 10 to 172. Figure 5.28 showes the power consumption of Pandabaord and Intel platform, in which the red bars represent power consumption of Pandaboard at the statuses of running one clip and two clips, whereas the purple bars depict Intel CPU's power. Moreover, blue bars and green bars represent Pandaboard CPU usage ratio and Intel CPU usage ratio respectively. From this chart, the Intel CPU power consumption grows around 10 watts as the CPU usage rise 25%, hence Intel CPU power consumption is not linear with, actually lower than, its usage ratio. When one 86MB video is need to processed, four Pandaboards consume total 16.68 watts (4.17 watts for each boards) to process while Intel CPU demands 44.24 watts power. But when four 86MB videos is to be processed, the Intel CPU uses 77.36 watts while 16 Pandaboards require 67.72 watts to complete the task with similar time spent. In this experiment, the energy consumption and time spend on merging and splitting video are omitted, because they are too trivial to be measured.



	1	2	3	4
🟦 ARM CPU usage ratio (%)	53,85	98,55		
🟥 single ARM CPU power (watts)	3,381	4,17		
🟩 Intel CPU usage ratio (%)	25,39	50,43	75,82	99,86
🟪 Intel CPU power (watts)	44,24	58,42	68,2	77,36

Figure 5.28: Video experiment result

Since the difference between Pandaboards is small, we can seem the EE of four-Pandaboard cluster equals to one single Pandaboard. When the total four-Pandaboard cluster transcodes the 86MB test video file as Intel workstation does, the EE ratio is 2.65, but when the Intel CPU transforming four test video at the same time, the EE ratio is 1.16. Thus the ARM processor has less advantage at this CPU-intensive application. However, according to Pereira's paper [35], the distributed ARM based server system may own advantage from its concurrency structure to accelerate the video transforming speed to reduce user response time, but this is out of the scope of this thesis.

5.4 Hadoop Experiment

The last experiment is about energy efficiency on map-reduce task. The power consumption is stable, for Pandaboard it is around 3.6 watts and for Intel CPU is 60 watts. Because of the stable ratio of time spent on shuffle phrase which does not cost computing capability to time spent on reduce phrase which exhausts CPU resource, hence the file size does not affect the average power.

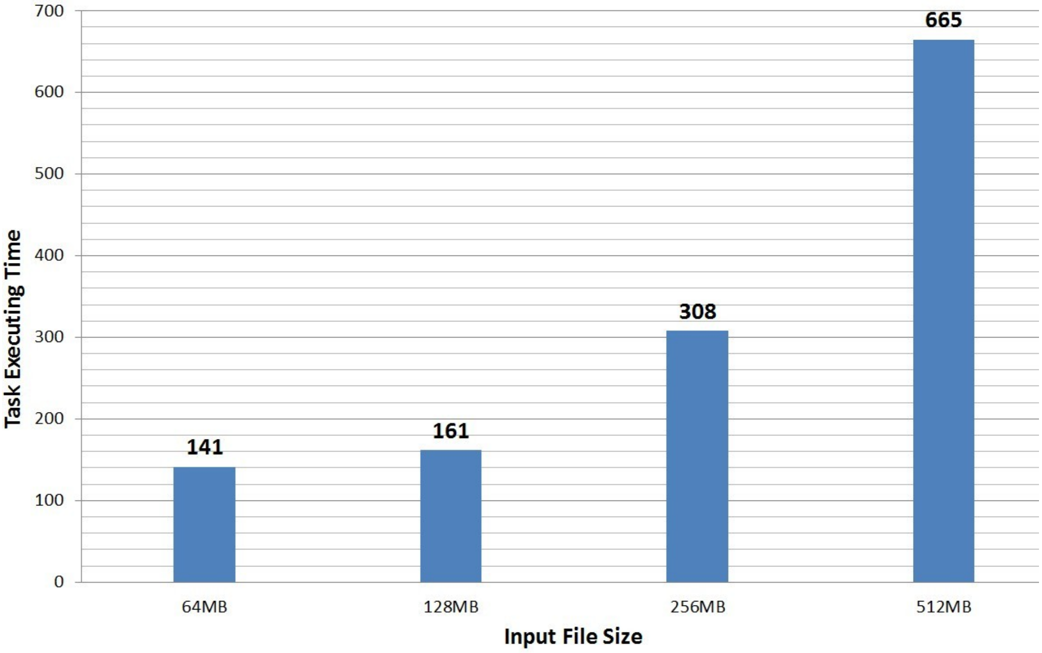


Figure 5.29: Pandaboard map-reduce executing time (second)

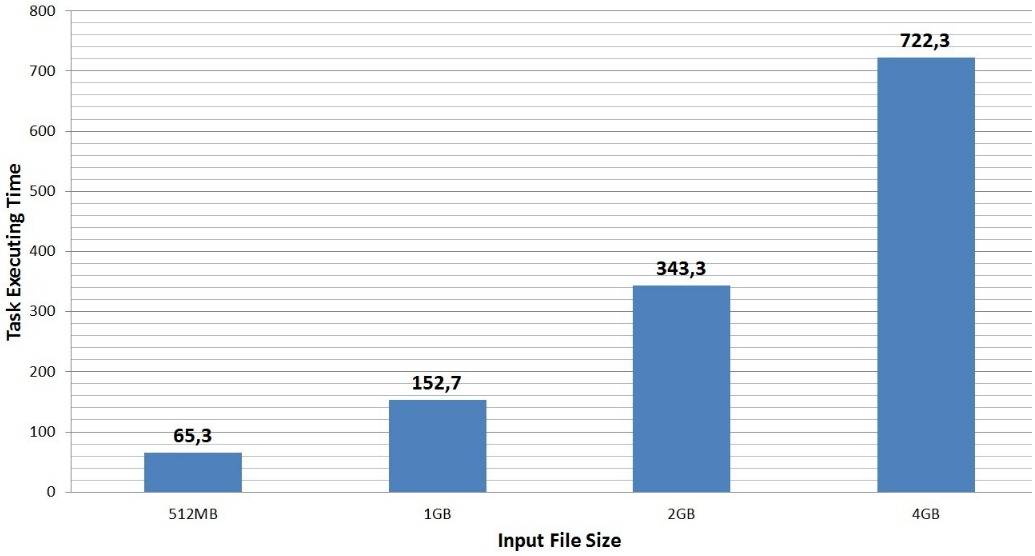


Figure 5.30: Intel map-reduce executing time (second)

When the work done, in this case, the input test file size, is 512 MB, the EE of ARM processor is 0.05 while EE of Intel CPU is 0.135. The ratio is as high as 27 and the Hadoop is very fault-prone on Pandaboard cluster, which can not even process file bigger than 512 MB successfully due to I/O errors frequently occur, thus the Pandaboard cluster is not suitable to replace Intel workstation in Hadoop application area. Moreover Figure 5.30 illustrates that the working time on Intel CPU is linear to the input size, but the curve of Pandaboard cluster is not smooth, that is because the experiment on Pandaboard cluster is not proceeding well, at shuffle phrase, Datanodes on Pandaboards always encounter I/O exceptions, which may be caused by using SD-card as storage media, and restart tasks many times, and the consequence is the record can not reflect the ARM processor's capability. The future work is to find out a platform with HDD storage to conduct this experiment again to get reliable outcome of energy efficiency of ARM processor.

Chapter 6

Discussion

From the previous experiments, ARM processors show some advantages at power saving when processors are used at an exhausting manner. Furthermore, in real enterprise applications, the processor utilization is not always 100% [27] and it fluctuates greatly with the time of a day or seasonally. For example, the Web site usually has much more visitors at daytime than at midnight, and the university course system is much busier at the beginning of semester than in normal days, hence at most of the time some processor is idle and reserved for usage peaks. But the power consumption of processor is not proportionate to processor usage ratio, the processor spends some energy on OS when no or few task is running. This static power consumption may decrease processor's energy efficiency index dramatically. For example, in low workload Web test, from Figure 6.1, the EE of Intel processor at 3500 requests/second is 47 and drops to 18 at 500 requests/second, hence fully using processor computing resource brings best energy efficiency to processors.

For Intel processors, the best energy efficiency is not a long-lasting status in reality, because dropping incoming requests will happen when the processor is fully used, thus IT department normally keeps server processor usage ratio under 80% to keep applications work stably and serve all requests. In addition, the Intel processors at idle status are also consuming considerable energy, even SpeedStep technology, which decreases processors' frequency automatically when work load is low, is applied. For the Intel platform in this study the idle power is around 25 watts, which is equal to the power of 5.5 exhaustively working Pandaboards.

However the ARM processor's low computing capability becomes an advantage in the periodical fluctuating work load situation, because, when the number of incoming requests is under certain thresholds,a smart load balancer can redistribute the requests forwarding to fewer processors, and the idle ARM processors can be turned off, while the alive ARM processors share

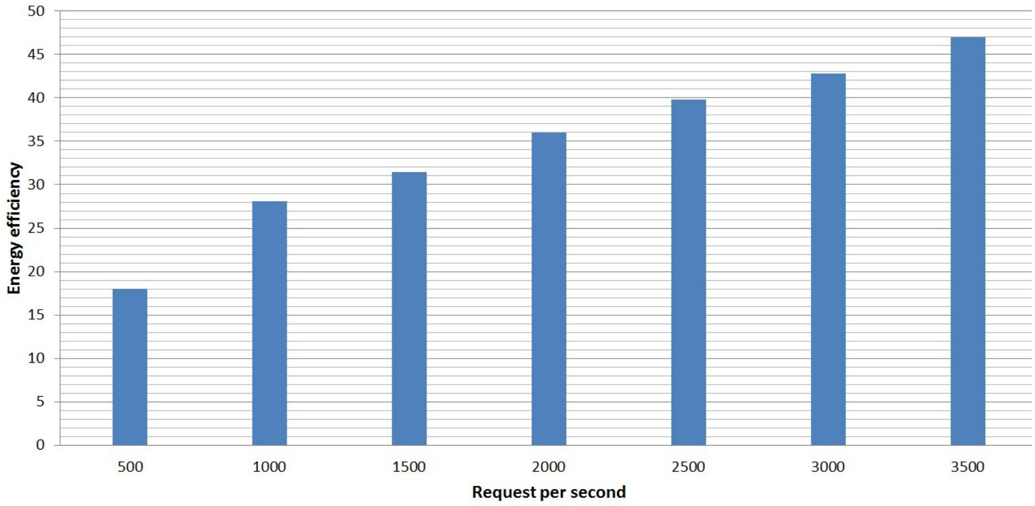


Figure 6.1: EE of Intel processor for high workload Web experiment

rest tasks and keep processors running exhaustively [23]. For example, on Figure 6.2, several ARM processors are deployed on one motherboard, and they all work in the working hours but only three of them are awake and take the night work shift in the midnight. In this manner, the energy is fine-grained managed in the processor cluster. For Web server cluster and video transcoding server cluster, a good load balance system combining a power management system could make this energy saving management possible. For Hadoop, a paper from Stanford University [19] depicts a method, in which a working covering subset of nodes that contains at least one replica of every file's blocks is alive and serving while the rest nodes can be shut down when task demand is low. And this energy saving principle is the same [18] as what discussed here. But for in-memory database, because SQL database is not good at scalability, shutdown even one server in cluster would impact the total availability, thus the SQLite in-memory database can not benefit from this scheme, however some distributed NoSQL database may take this advantage [32].

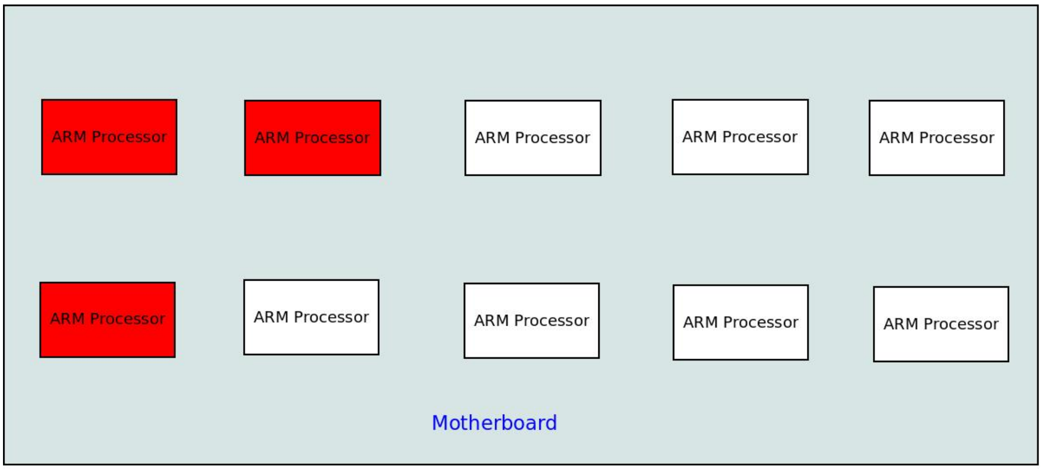


Figure 6.2: Energy management system

Chapter 7

Conclusions

7.1 Key Findings

The aim of this paper is to verify the feasibility of replacing Intel CPU with low energy consumption ARM processor, and compare the energy efficiency between them in a variety of applications scenarios to filter out which kinds of applications ARM processors are more suitable. A cluster built up by four Pandaboards is utilized to conduct the comparison with a Intel CPU based workstation in four different application scenarios: Web server, in-memory database, video transcoding server and Hadoop cluster. Through the four experiments, we found Pandaboard cluster has potential to execute the same amount of task that Intel CPU workstation executed in the same time, moreover its energy efficiency is at least one time higher than Intel CPU when running Web server, in-memory database and video transforming server, and in in-memory database reading test, the ARM processor's energy efficiency is as eight times efficient as Intel CPU, hence the multi ARM processor based server/server cluster is capable to substitute the Intel CPU server/cluster and save at least half of Intel CPU energy consumption. Another finding is that the low CPU utilization and high scalability application is more suitable for ARM processor server cluster with power management system, which is able to shut down idle servers and keep the rest fully used to reduce power consumed by operating systems and to restart servers for demanding changes.

7.2 Limitations of Study

First of all, we did not find an easy way to measure the power consumption of ARM processor. Thus we have to use the power consumption of the whole Pandaboard to represent the ARM processor consumption, which decreases the real energy efficiency of ARM processor. Because the nominal maximum power of ARM processor is around 2 watts, in my opinion, the real energy efficiency should be at least double of current value, because the measured board power is double of ARM processors' nominal power. Another limitation is data storage device. Pandaboard utilizes SD card as its data storage, which can not provide the same magnitude of random access speed and stability. Hence the first three experiments are all memory based, with thise experiments the difference between HDD and memory is avoided. But the Hadoop is aiming to process big data file, thus we have to use SD card as storage with no option, because the memory of Pandaboard is less than 700 Megabyte. The Hadoop experiment is, as we speculated, not successful, and the data can not represent the ARM processor's performance at all. The last but not the least is that the Pandaboard platform can not really stand for the trend of industry. To compensate the low capability of ARM processor, engineers design many ARM processors on one board and share the same memory and other devices, the multi-processor motherboard technology combines many ARM processors together in a more efficient and economic fashion. The cluster we designed in this thesis, in which each node is connected by Ethernet, is lame at communications delay and can not reach the best performance of the ARM processors set.

7.3 Future Work

This paper only makes a very basic verification of the potential that ARM processor is able to replace Intel CPU in an energy efficient way. In the future, an ARM multi-processor motherboard with the same peripheral devices as a Intel CPU server, should be acquired, and a fine-grain computing capability and power management system should be developed and deployed, and then some experiments based on those platforms in larger application coverage will give a more industrial indicating conclusion.

Bibliography

[1] ANONYM. Map/Reduce Structure, 2011. `http://map-reduce.wikispaces.asu.edu/file/view/figure5_%281%29.png`. Accessed 25.7.2011.

[2] ANONYMOUS. Microsoft Announces Support of System on a Chip Architectures From Intel, AMD, and ARM for Next Version of Windows, 2011. `http://www.microsoft.com/presspass/press/2011/jan11/01-05socsupport.mspx`. Accessed 25.8.2011.

[3] ARM COOPERATION. ARM the architecture for digital world, 2011. `http://www.arm.com/`. Accessed 25.8.2011.

[4] BYRAM, S. an interview with steve furber. *CommunICatIons of the aCm 54*, 5 (2011).

[5] CALXEDA. About Calxeda, 2011. `http://www.calxeda.com/about.php`. Accessed 25.7.2011.

[6] CHEN, Y., KEYS, L., AND KATZ, R. Towards energy efficient mapreduce. *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-109, Aug* (2009).

[7] DAG WIEERS. Dstat: Versatile resource statistics tool, 2011. `http://dag.wieers.com/home-made/dstat/`. Accessed 25.8.2011.

[8] DEAN, J., AND GHEMAWAT, S. Mapreduce: Simplified data processing on large clusters. *Communications of the ACM 51*, 1 (2008), 107–113.

[9] GARCIA, A., AND KALVA, H. Cloud transcoding for mobile video content delivery. In *Consumer Electronics (ICCE), 2011 IEEE International Conference on*, IEEE, pp. 379–380.

[10] GARCIA, A., KALVA, H., AND FURHT, B. A study of transcoding on cloud environments for video content delivery. In *Proceedings of*

the 2010 ACM multimedia workshop on Mobile cloud media computing (2010), ACM, pp. 13–18.

[11] GHEMAWAT, S., GOBIOFF, H., AND LEUNG, S. The google file system. In *ACM SIGOPS Operating Systems Review* (2003), vol. 37, ACM, pp. 29–43.

[12] JAGGAR, D., ET AL. Arm architecture and systems. *IEEE micro 17*, 4 (1997), 9–11.

[13] JOE. Apache vs Nginx : Web Server Performance Deathmatch, 2008. `http://www.joeandmotorboat.com/2008/02/28/apache-vs-nginx-web-server-performance-deathmatch/`. Accessed 25.8.2011.

[14] KOOMEY, J. Estimating total power consumption by servers in the us and the world, 2007.

[15] KOOMEY, J. Estimating total power consumption by servers in the us and the world, 2007.

[16] KRISHNAMURTHY, B., ET AL. Analyzing factors that influence end-to-end web performance. *Computer Networks 33*, 1-6 (2000), 17–32.

[17] KURT MACKIE. Are ARM-based servers the next big trend, 2011. `http://gcn.com/Articles/2011/01/28/ECG-Will-ARM-Based-Servers-Lift-Microsofts-Cloud.aspx?Page=1`. Accessed 25.6.2011.

[18] LANG, W., AND PATEL, J. Energy management for mapreduce clusters. *Proceedings of the VLDB Endowment 3*, 1-2 (2010), 129–139.

[19] LEVERICH, J., AND KOZYRAKIS, C. On the energy (in) efficiency of hadoop clusters. *ACM SIGOPS Operating Systems Review 44*, 1 (2010), 61–65.

[20] LMBENCH COMMUNITY. LMbench - Tools for Performance Analysis , 2011. `http://www.bitmover.com/lmbench/`. Accessed 25.8.2011.

[21] LVS COMMUNITY. LVS, 2011. `http://www.linuxvirtualserver.org/`. Accessed 25.8.2011.

[22] MAURICIO ALVAREZ. HD-VideoBench. A Benchmark for Evaluating High Definition Digital Video Applications, 2008. `http://alvarez.site.ac.upc.edu/hdvideobench/index.html`. Accessed 25.6.2011.

[23] MEISNER, D., GOLD, B., AND WENISCH, T. Powernap: eliminating server idle power. *ACM SIGPLAN Notices 44*, 3 (2009), 205–216.

[24] MICHAEL G. NOLL. How the final multi-node cluster will look like, 2007. `http://www.michael-noll.com/blog/uploads/Hadoop-multi-node-cluster-overview.png`. Accessed 25.7.2011.

[25] MICHAEL PETERS. httperf manual, 1998. `http://www.hpl.hp.com/research/linux/httperf/httperf-man.txt`. Accessed 25.8.2011.

[26] MOSOON SOLUTION INC. Power Monitor, 2011. `http://www.msoon.com/LabEquipment/PowerMonitor/`. Accessed 25.8.2011.

[27] NAGOTHU, K., KELLEY, B., PREVOST, J., AND JAMSHIDI, M. On prediction to dynamically assign heterogeneous microprocessors to the minimum joint power state to achieve ultra low power cloud computing. In *Signals, Systems and Computers (ASILOMAR), 2010 Conference Record of the Forty Fourth Asilomar Conference on*, IEEE, pp. 1269–1273.

[28] NETCRAFT. Market Share for Top Servers Across All Domains August 1995 - September 2011, 2011. `http://news.netcraft.com/`. Accessed 20.9.2011.

[29] NGINX COMMUNITY. Nginx, 2011. `http://wiki.nginx.org/`. Accessed 25.8.2011.

[30] ORACLE CORPORATION. Java SE Desktop Technologies, 2005. `http://www.oracle.com/technetwork/java/javase/tech/index-jsp-140239.html`. Accessed 25.8.2011.

[31] ORTIZ JR, S. Chipmakers arm for battle in traditional computing market. *Computer 44*, 4 (2011), 14–17.

[32] OZSU, M., ĂZSU, M., AND VALDURIEZ, P. *Principles of distributed database systems.* Springer, 2011.

[33] PANDABOARD COMMUNITY. PandaBoard Platform Specifications, 2011. `http://pandaboard.org/content/platform`. Accessed 25.7.2011.

[34] PANDABOARD COMMUNITY. PandaBoard Set-up, 2011. `http://pandaboard.org/sites/default/files/board_setup_v3.png`. Accessed 25.7.2011.

[35] Pereira, R., Azambuja, M., Breitman, K., and Endler, M. An architecture for distributed high performance video processing in the cloud. In *2010 IEEE 3rd International Conference on Cloud Computing* (2010), IEEE, pp. 482–489.

[36] sqlite Community. SQLite Download Page, 1998. `http://www.sqlite.org/download.html`. Accessed 25.7.2011.

[37] sqlite Community. sqlite performance benchmark, 1998. `http://www.sqlite.org/speed.html`. Accessed 25.5.2011.

[38] sqlite Community. sqlite, 2011. `http://www.sqlite.org/`. Accessed 25.8.2011.

[39] The Apache Software Foundation. Hadoop Common Releases, 2007. `WelcometoHadoopCommon`. Accessed 25.8.2011.

[40] The Apache Software Foundation. Welcome to Hadoop Distributed File System!, 2007. `http://hadoop.apache.org/hdfs/`. Accessed 25.8.2011.

[41] The Apache Software Foundation. Welcome to Hadoop MapReduce!, 2007. `http://hadoop.apache.org/mapreduce/`. Accessed 25.8.2011.

[42] The Apache Software Foundation. Apache HTTP Server, 2011. `http://httpd.apache.org/`. Accessed 25.8.2011.

[43] The Apache Software Foundation. Hadoop Common Releases, 2011. `http://hadoop.apache.org/common/releases.html#11+May%2C+2011%3A+release+0.20.203.0+available`. Accessed 25.8.2011.

[44] The Apache Software Foundation. HDFS architecture, 2011. `http://hadoop.apache.org/common/docs/current/images/hdfsarchitecture.gif`. Accessed 25.7.2011.

[45] The Apache Software Foundation. HDFS Data Nodes Structure, 2011. `http://hadoop.apache.org/common/docs/current/images/hdfsdatanodes.gif`. Accessed 25.7.2011.

[46] The ffmpeg Project. ffmpeg offical site, 2011. `http://ffmpeg.org/`. Accessed 25.8.2011.

[47] The MPlayer Project. Mplayer offical site, 2011. `http://www.mplayerhq.hu/design7/news.html`. Accessed 25.8.2011.

[48] Tsirogiannis, D., Harizopoulos, S., and Shah, M. Analyzing the energy efficiency of a database server. In *Proceedings of the 2010 international conference on Management of data* (2010), ACM, pp. 231–242.

[49] Yahoo cooperation. Hadoop at Yahoo!, 2011. `http://developer.yahoo.com/hadoop/`. Accessed 25.8.2011.

[50] ZONA RESEARCH. THE NEED FOR SPEED II, 2001. `http://www.keynote.com/downloads/Zona_Need_For_Speed.pdf`. Accessed 25.8.2011.

[51] ztsystems.com. ZT Systems Announces ARM-based Server Solution for Breakthrough Energy Efficiency and Density in Hyperscale Data Centers, 2011. `http://www.ztsystems.com/Default.aspx?tabid=1484`. Accessed 25.7.2011.