An Embedded System for applying High Performance Computing in Educational Learning Activity

Irene Erlyn Wina Rachmawan, Nurul Fahmi, Edi Wahyu Widodo, Samsul Huda, M. Unggul Pamenang, M. Choirur Roziqin, Andri Permana W, Stritusta Sukaridhoto, Dadet Pramadihanto.

Graduate School of Informatics and Computer Engineering
Politeknik Elektronika Negeri Surabaya
Jl. Raya ITS Sukolilo Surabaya 60111, Indonesia
Telp: 6231 5947280 Fax: 6231 5946114
E-mail: ireneerlyn@sfc.keio.ac.jp, {wiedhodho, nurulfahmi, samsul, unggul, choiruroziqin, andripermana}@pasca.student.pens.ac.id, {dhoto, dadet}@pens.ac.id

Abstract

HPC (High Performance Computing) has become more popular in the last few years. With the benefits on high computational power, HPC has impact on industry, scientific research and educational activities. Implementing HPC as a curriculum in universities could be consuming a lot of resources because well-known HPC system are using Personal Computer or Server. By using PC as the practical moduls it is need great resources and spaces. This paper presents an innovative high performance computing cluster system to support education learning activities in HPC course with small size, low cost, and yet powerful enough. In recent years, High Performance computing usually implanted in cluster computing and require high specification computer and expensive cost. It is not efficient applying High Performance Computing in Educational research activiry such as learning in Class. Therefore, our proposed system is created with inexpensive component by using Embedded System to make High Performance Computing applicable for leaning in the class. Students involved in the construction of embedded system, built clusters from basic embedded and network components, do benchmark performance, and implement simple parallel case using the cluster. In this research we performed evaluation of embedded systems comparing with i5 PC, the results of our embedded system performance of NAS benchmark are similar with i5 PCs. We also conducted surveys about student learning satisfaction that with embedded system students are able to learn about HPC from building the system until making an application that use HPC system.

Keywords: HPC, Cluster, Education learning, Learning model.

1. INTRODUCTION

HPC (High Performance Computing) has become more popular in the last few years. With the benefits on high computational power, HPC has impact on industry, scientific research and educational activities. HPC system is a mechanism to get high speed computing for process an information that require a great ability using several processors as part of a single machine or a cluster of several computers as an individual resource. However, the high cost of HPC equipments and maintenance makes it difficult to build it. Possible sollutions include, PC's in the laboratorium room, using cluster running on virtual machines in background, but in some cases also unavailable during outside of class time. Other solutions, virtualized parallel resources on non parallel hardware are typically free or low cost, highly available and have good turnaround, but may have misleading performance characteristics due to limited physical parallelism. Therefore, cluster computing is the best alternative way to build a high performance computing platform with low cost to give HPC learning for students in university [1-3].

Clusters are cheaper to own, and can be expanded to nearly unlimited size. It also addresses the flexibility and scalability issues by allowing the upgrading of individual components in the cluster. Skills in high-performance computing are increasingly desired. Therefore, there exists a great need of skilled and educated students who are able to design, implement and operate such HPC environments. In order to offer HPC related courses such as Distributed and Parallel system, we build affordable infrastructure using other low power processor architectures such as embedded system is currently being explored since these processors have been primarily designed for the mobile and embedded device market. Thus, students can explore the HPC knowledge with hands-on experience.

In the end of 1999, High Performance Computing has been implemented for educational purpose, those methods used to support the performance of computer laboratory for processing the student learning activity in university. It is use the MPI library and has a shared memory over one laboratory [4]. The topic course about High Performance Computing started to be implemented in university in early of 20th century, where PC is used as the nodes, both for master and slaves. One of the researcher in 2001 focused on the parallel programming model for junior undergraduate student. It is aims the student to be able to develop a program that could be runs in multiple computing [5]. Course about HPC is continuing to be teached for university student as shown in 2005 research about the application of HPC system as practical courses. In those research Beowulf cluster computing by using the Pentium 3 for the master nodes and Pentium 2 for the slaves nodes, and the student learned about HPC thourgh the material provided in course [8]. The newest trend of HPC lesson is shown in 2014, where the course is likely to implement the low cost system to perform the high performance computing for solving the simple mathematical process [9][10], where these lesson use the embedded system to build a clustered computer for solving

the simple computation such as matrices calculation. The timeline of how HPC has been known and implemented shown in Figure 1.

In this paper, we presents an innovative high performance computing cluster system to support education learning activities in HPC course with small size, low cost, and yet powerful enough. The cluster consists of 32 embedded nodes, 2 network switches and 2 customized power supply. Students be involved in the construction of cluster computer, build clusters from basic computer components, benchmark performance, and implement simple parallel case using the cluster. We also design the learning model to achieve the following pedagogical goals:

- Understand basic concepts of High Performance computing.
- Provide practical experience on parallel computing based on shared memory and distributed memory.
- Motivate students for advanced topics and learning about practical, and Economics factor in Laboratorium .

This paper is organized as follows: Section 2 gives the related works as the reference of this paper. Section 3 describes the originality of this research through educational perspective. Section 4 discusses a design system of Embeddeded System in this research.In section 5, we end with some conclusion.

2. RELATED WORKS
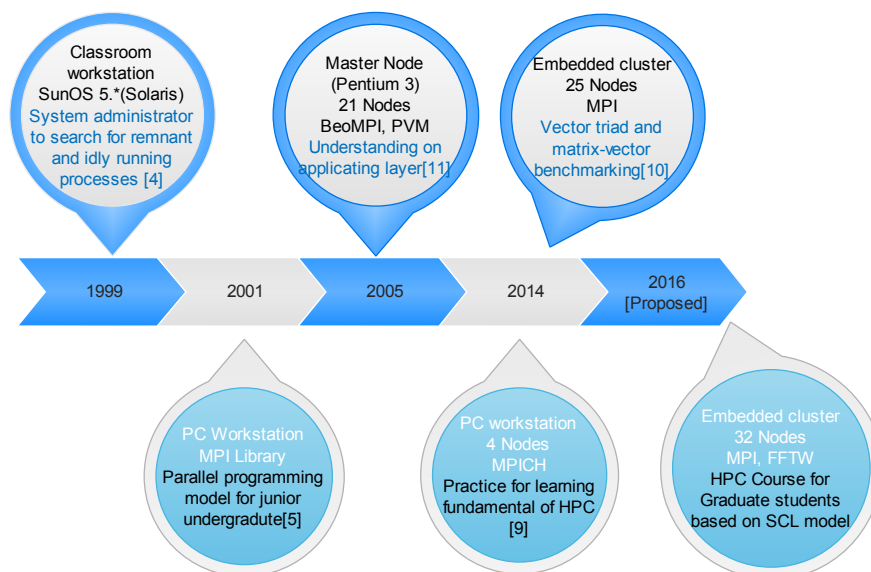
Figure 1. Timeline of HPC in education field

Leo Nagamatsu, *et al* [4], they explaint first Parallel Computing and implementation for supporting education process not for applied to embedded system. And also in the research which proposed [5], they describe Parallel Computing in the classroom with using Personal Computing

(PC) / Workstation. For the implementation programming they using MPI library.

Amit Chhabra, *et al* [6], proposed a parallel computing framework based on cluster of networked desktop PCs which called Cluster Based Parallel Computing framework (CBPCF) based master-slave. In this system, they do evaluation with experiment like parallel Matrix multiplication and Pi (π) value approximation. The result from experiment shown bandwidth requirements in problems like matrix multiplication, is a major hindrance to get good performance as major percentage of the turnaround time is consumed as communication time. Pi approximation problem has shown good amount of speedup as well as efficiency due to more computation work involved than communication in this paper.

Nanjesh B R, *et al* [7], they explained about the processing use of MPI where can be realized by implementing the parallel applications like parallel merge sorting. the aim is to evaluate statistics of parallel execution and do comparison with the time taken to solve the same problem in serial execution to demonstrate communication overhead involved in parallel computation. They have analyzed the performance of parallel method with some case. *First*, they analyze with case single node analysis to show performance dependency on RAM. *Second*, Single node analysis to show communication overhead involved in parallel computations. *Third*, Multiple nodes analysis with smaller unsorted list of elements (computation time < communication time) and the last multiple nodes analysis with larger unsorted list of elements (computation time>communication time). The result total time taken to compute the result decreases drastically when the number of nodes increases. The results obtained from the experiment, if the number of nodes meningkat then the total time for the faster processing.

Mohd Noor Ikhwan Abdul Rahim, *et al* [8], they explained performance between Control Processing Unit (CPU) cluster and General Propose Graphics Processing Unit (GPGPU) in the processing on Cluster Computing. They use 5 computer nodes and 1 head node. The head node is installed with Rock Cluster Distribution based on CentOS linux. In this system, they analyzed performance with comparing between CPU and GPGPU. The cluster's protein folding performance is measured in ns/day, average CPU utilization, average memory usage, and average network bandwidth will be taken into account in this simulation.

Mingliang Xu, *et all* [9], explains the Realization of Small Cluster Parallel Computing Environment for college Student where the student targeted to known the process of implementing the high performance computing to analyze the simple mathematical logics. While in the same year, Aaron M. et all [10] presented the new approach of using the raspberry pi to maintain the low cost of High Performance System in educational purpose. And then Ilias Stavrakes, *et al* [11], proposed Parallel Computing Course use Beowulf Clusters. In this system they using PCs with Pentium III as master node and PC Pentium II as slave nodes. So, total node which use is 21 nodes.

Sukaridhoto, *et al* [13], representing the embedded system to practice High Performance Computing using Pandaboard to compute small number of matrices.

3. ORIGINALITY

The original proposed idea of this research is for implementing and creating the High Performance Computing to giving the students the opportunity to explore advanced technologies. We proposed a new system by using Embedded computer to perform the High Performance Computing in economic way, so the application of High Performance Computing could be implemented for educational purpose. The main idea of this system prototype is to make student be able to creatively design and modify the system which need High Performance Computing according their interest problem and idea, then verified the result through actual experiment. In recent years, High Performance computing usually implanted in cluster computing and require high specification computer and expensive cost. It is not efficient applying High Performance Computing in Educational research activiry such as learning in Class. Therefore, our proposed system is created with inexpensive component by using Embedded System to make High Performance Computing applicable for leaning in the class. The implementation of embedded system aims students be able to create an innovative high-performance computing cluster system to support education learning activities by using the small size, low cost, and yet powerful High Performace Computing. Our novelty is using 32 nodes of panda boards for HPC and performance analysis. The novelty of our research also includes student curriculum for HPC and Student satisfaction survey about HPC class.
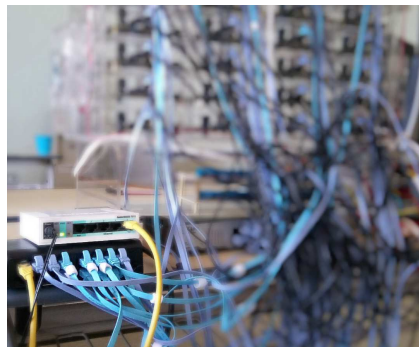
4. SYSTEM DESIGN
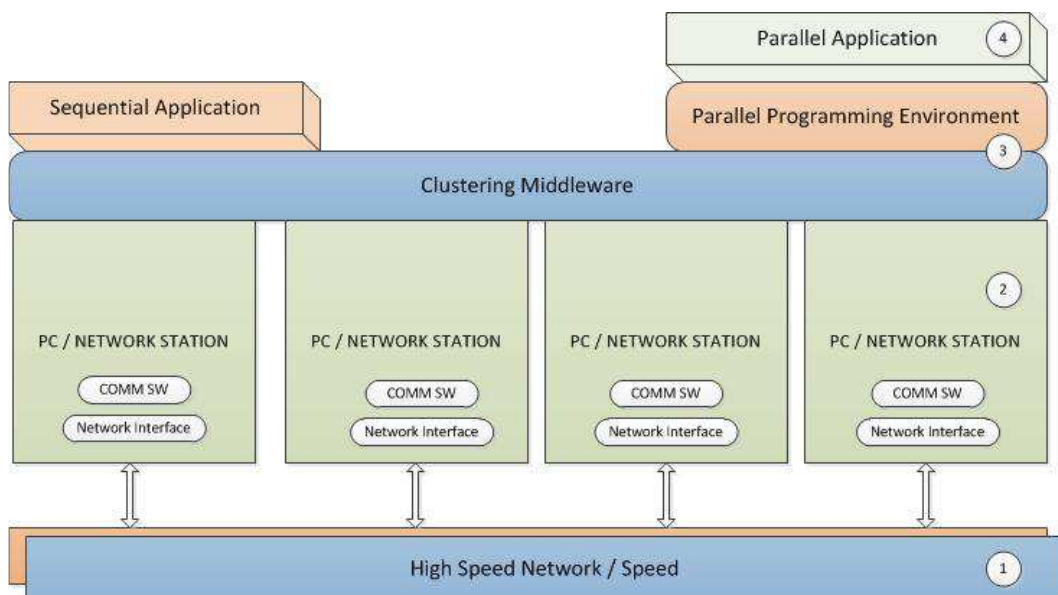
Figure 2. Overview of parallel computing system

In this session we describe all of phase the implementation of the using embedded systems on HPC learning. We design the architecture of cluster embedded system and learning model. An alternative solution to the HPC platform is the cluster-based embedded system for its flexibility and feasibility of high computing power with relatively low cost. Overview of parallel computing system described in Figure 2.

Part 1. High Speed Network / Speed :   Cluster components are usually connected to each other through a local area network (LAN) to supporting high speed network and also installed NFS to to sharing files between nodes, NIS to manage user accounts and SSH to remote system.

Figure 3. LAN to supporting the communication between nodes

Part 2. Embedded System : is a computer system designed to perform a specific function. Embedded systems are built on the basis of specific needs and run processes that have been defined, unlike the personal computer (PC) which is a general-purpose. Therefore, we design and implement an affordable high performance cluster system on the PandaBoard ES with 32 computing nodes. The power source for each computing nodes supported by 5V customized power supply.



Figure 4. Embedded System using Pandaboard

Part 3. Parallel Programming Environmental: is very useful to transfer data from machine to machine, but it lacks a simple way of creating threads. Threads are very important to clusters now. This is because almost all processors are made with more than one core. MPI works by sending messages from process to process on different machines. When running a

program then has been written with MPI, the creator of the processes (Master Node) sends the program to all the Compute Nodes. When they get the program they are given a rank. This rank is how a programmer controls the flow of the program, and all of this exists in the MPI communication world. You then simply state that if the machine has some rank or some range of ranks, then it should do some task and send the results to server.

Part 4. Parallel Computing: program to compute matrices and shows performance of embedded system for High Performance Computing, we design two testing cases, image fourier transform and benchmark testing.

4.1 System Architecture

Our proposed cluster system consists of 32 computing nodes from #1, #2, until #32, where the node #1 works as server node for various applications, such as NIS, NFS, and SSH servers, etc., and is directly accessible from terminal outside or from node #1 which connected with monitor and keyboard like shown in Figure 3. These 32 computing nodes, 2 network switches, and 5V customized power supply are mounted together. The computing nodes are connected over conventional 100Mbps Fast Ethernet. Primary persistent storage is via a SD Card slot allowing SDHC card class 10 with 8GB capacity. The board include wired 10/100 Ethernet as well as wireless Ethernet and Bluetooth connectivity. The board can output video signals via DVI and HDMI interfaces. It also has 3.5 mm audio connectors. It has two USB host ports and one USB On-The-Go port, supporting USB 2.0. The PandaBoard has a real-time clock can be synchronize with NTP server, and runs the Linux Kernel with ARM architecture.
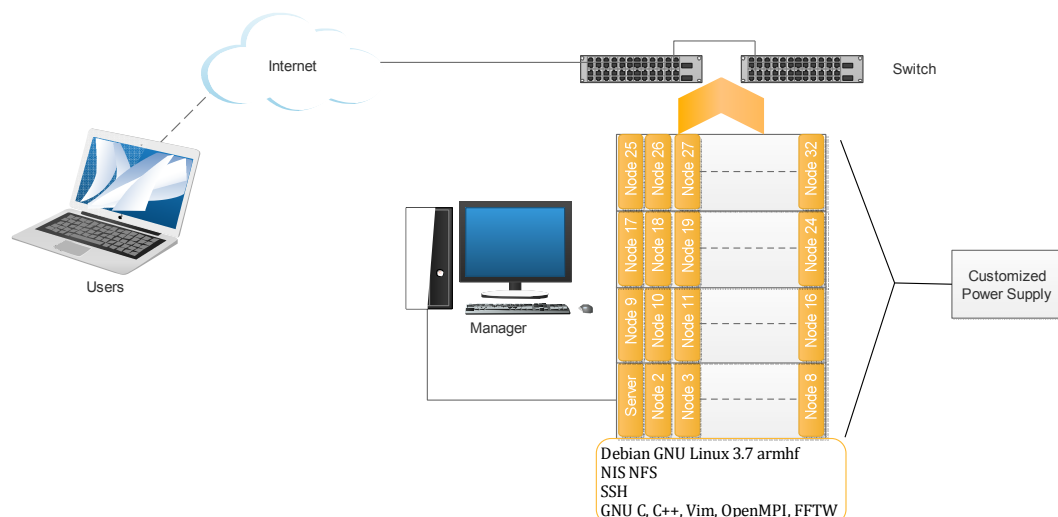
Figure 4. Cluster system architecture

The ARM Cortex-A9 in PandaBoard is a 32-bit multi-core processor which implements the ARMv7 instruction set architecture. The cortex-A9 can

have a maximum of 4 cache-coherent cores and clock frequency ranging from 800 to 2000 Mhz. Each core in the cortexA9 CPU has a 32 KB instruction and a 32 KB data cache. One of the key features of the ARM Cortex-A series processors is the option of having Advanced SIMD (NEON) extensions. NEON is a 128-bit SIMD instruction set that accelerates applications such as multimedia, signal processing, video encode/decode, gaming, image processing etc. The features of NEON include separate register files, independent execution hardware and a comprehensive instruction set. It supports 8, 16, 32 and 64 bit integer as well as single precision 32-bit floating point SIMD operations.

We used DC 5V 40A 200W Transformer Switch Power Supply with cooling fan. With this power supply we can power up 32 nodes of PandaBoard. In each PandaBoard need 5V DC with 2A minimum for requirement to run. To distribute the power supply, we made custom cables distribution. With this board also, we can also build stacking power supply to provide power supply for more PandaBoard.

The nodes run Debian GNU/Linux 3.7 configured for ARMv7 on SDHC and daemons. Figure 5 shows the overview of cluster-based embedded system. There are several applications, which can support for programming, tools and parallel computing.

1. It supports a variety of development tools and libraries for parallel programming including GNU C, C++ compiler, and Message Parsing Interface (MPI) by using OpenMPI, and FFTW.
2. It provides with a variety editor application such as Vim, nano and etc.
3. It also includes remote execution communication with secure shell (SSH).
4. It uses NFS system to share files between nodes.
5. It implements NIS to administer user accounts to be use in nodes.
6. With shell scripting provided by BASH, user can easily manage and control many nodes.
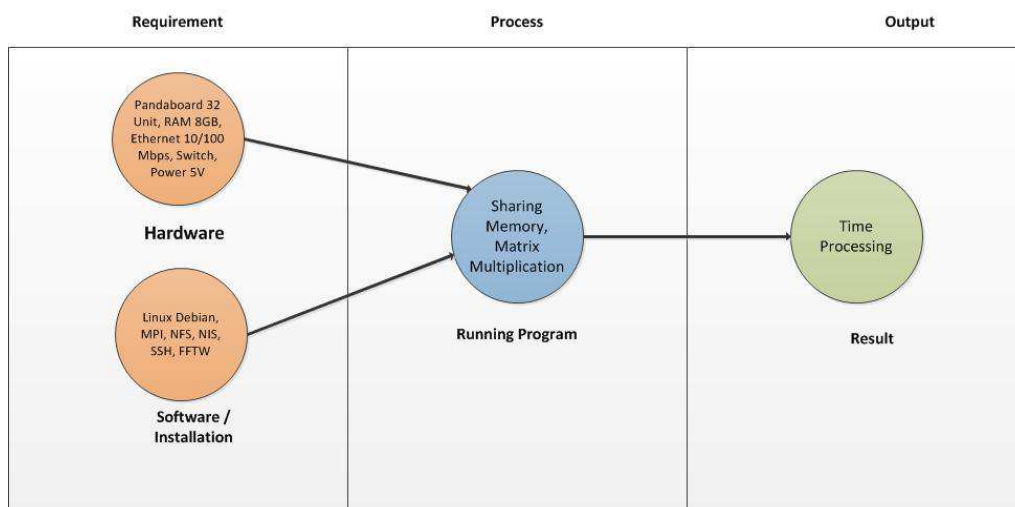
Figure 5. Overview of cluster-based embedded system

Parallel computing is calculating the computation by using 2 or more CPU / Processor in a same computer or a different computer in which case each instruction is divided into several instructions and then sent to the processor involved computing and done simultaneously. For the process of division of the computing process is performed by a software which responsible for managing. In order to support clustering programming, Message Passing Interface (MPI) standard is employed and Open MPI (version 1.4.2) is installed in the system because of its high performance message passing libraryOne of the most familiar use is the Message Parsing Interface (MPI).

4.2 MPI

MPI (Message Passing Interface) is a specification of API (Application Programming Interface) that enables communication between computers on a network in an attempt to complete a task. Paradigm Message - Passing the implementation of MPI provide a unique approach in building a software in the domain of certain functions, which in this case in an environment of distributed systems, thus providing the ability to software products built on middleware is to be able to exploit the capabilities of computer networks and computing parallel.

Figure 6 shows the mechanism of the implementation of message-passing program-based, point-to-point communication is the simplest communication which will involve a process of working together. In the API of the MPI, it can simply be used MPI_Send or MPI_Recv, which will allow the occurrence of a specific process that one can send a data message to a specific process to another like shown in Figure 6. Operating point-to-point communication are particularly useful in communications irregular or patterned.
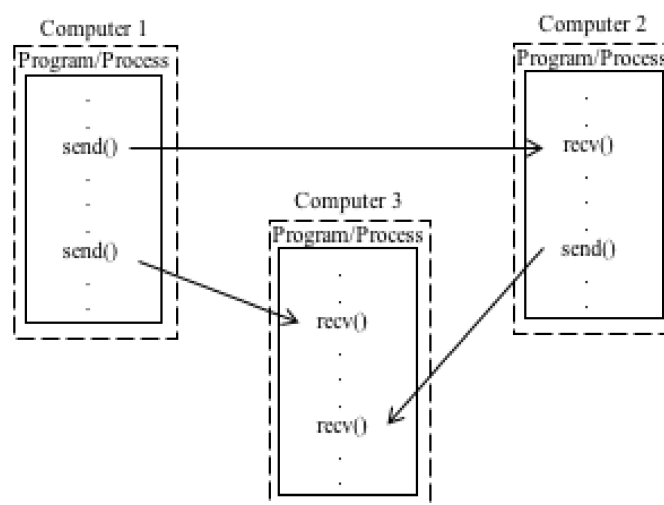

Figure 6. MPI scheme

Unlike the point-to-point communication, collective communication on the MPI API involves communication between all processes in a process group (in terms of the overall pool process or a subset defined in the program). Simple interface functions can be MPI_Bcast (broadcast) the sending of data from a process to all other processes in the group, and MPI_Reduce function to do the opposite. Type of communication gives two advantages: first, the communication operations allow programmers to express complex operations using a simple semantics, second, optimizing implementations can perform operations in a way that is not provided by the type of operating point-to-point communication.
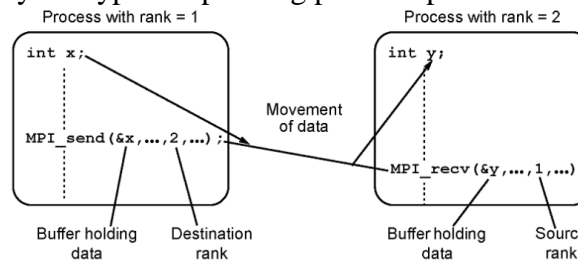
Figure 7. Delivery and Receipt Message Method

4.3 HPC Course Model

Rationale of the SCL (Student Center Learning) [13] is constructivist learning theory. The principle of constructivist theory comes from tori learning developed by Jean Piaget (1983), Jerome Breuner (1961), and John Dewey (1933), which focus the learning process on changing the behavior of learners themselves and experienced directly to establish the concept of learning and understanding. Furthermore, the concept of the learning experience of the triangle Dale prove that learning experience for themselves the real or actual condition and control the learning process is the fulfillment of a better learning experience than learning by observing. When viewed from the condition of students in receipt of learning experiences, a sense of anxiety that always burdening the learner will be reduced along with their interaction in the learning process. To be able to follow the course of this HPC, it would require some ability in other areas, this is because due to the HPC course requires participants to have knowledge of programming, networking, and operating system. The following will be described in more detail about the relevance of each of the courses: (1) Programming: Knowledge of programming is the basic needs before going into HPC material, because the success of HPC is strongly influenced by the ability of programming. Programming language that can be used on HPC can vary, for example: C, C ++, Java, Python, and many others. Performance of HPC is also determined by the style of programming from its developers. (2) Networking:In HPC, there are several techniques in the division of tasks, where one of the techniques for data exchange is through the network. Knowledge of the network is no less important than programming. Start of exchanging messages between nodes, data sharing (NFS) and permissions (NIS). (3) Operating System: It is no less important in HPC is how the division of tasks and memory. To perform the division of tasks and memory required

knowledge of operating system. Therefore, it is also taking an important part in the course HPC. The load must be able to master the ability at the end of the learning will be broken down into pieces that make up the capabilities of the ability or competence end by itself.

In the course High Performance Computing perform SCL approach to teaching methods, among others:

1. Students should be actively involved in the learning process that is triggered from intrinsic motivation.

    In the lab High Performance Computing students are directly involved actively in preparing all the needs that are used to assemble the hardware parallel computing begins installing open mpi, prepare the tools required, at the stage of testing and comparison algorithms course with the guidance of lecturer of subjects High performance Computing.

2. The topic, issue, or subject of learning should be attractive and stimulate intrinsic motivation

    Topics and issues discussed at the High Performance Computing course very interesting sparking motivation for students to complete this project. Parallel Computing is designed to facilitate a set of computers that are connected to other computers so that they can work together to operate and work closely and visible to network clients as if the computer is 1 piece unit of a computer.

3. The learning experience gained through the atmosphere of a real or actual and relevant with the knowledge and skills required and used in the workplace.

    HPC has impact on industry, scientific research and educational activities. HPC system is a mechanism to get high speed computing for process an information that require a great ability using several processors as part of a single machine or a cluster of several computers as an individual resource.

4.4. HPC Teach Model

At its core practice learning at the Polytechnic goal to give the learning experience of students to master the skills of a particular area of expertise. Good skills mastery of skills physical and intellectual activity involving students directly using equipment similar to the equipment during the actual learning process.

Table 1. Teaching Material

No.	Course Outline	Contents	Objective
1.	Cluster computer	Architecture and interconnection networks.	To understand the architecture of Cluster Computer.
2.	Parallel computing	Priciples and programming modes.	Learn and practice the principle of parallel computing.
3.	MPI	Getter and scatter techniques.	Learn and practice MPI getter and scatter techniques.
4.	FFTW	Fourier transform analysis using MPI.	Implementing and understanding several

			matematical process for parallel computing.
5.	Cluster-based embedded system	Constructing and building system	To be able to design the system of HPC.
6.	Application for Cluster-based embedded system	Designing, collecting data and experimental study.	To learn and analye the performance of Cluster computing using Embedded System.
7.	Measuring the performance of Cluster-based embedded system	Benchmarking parallel application the class A and B of IS type of NAS (NASA Advanced Supercomputing).	To analyze the performance of Cluster computing using Embedded System.

Teaching material of this course shown in Table 1. and course outline number 1 until 3 in Table 1. are implemented in the first month of HPC courses until mid term, this course is practiced in Student PC with total maximum node is four using virtual box. Adding the number of nodes in PC using virtual box will be extremely slowing the PC in processing parallel computing program. The embedded system using for cluster computing is implemented in course outline number 4 until number 7. Figure 8 shows a parallel computing hardware that students make as the implementation of the High Performance Computing courses. Parallel Computing is taking 32 units panda board that functions as a mini PC, 8 GB micro sd serves as RAM, 10/100 Mbps Ethernet LAN cable that serves as a liaison between the network of mini PCs, Switch which serves to multiply and get the Internet network and power 5v serves to connect the hardware to the mains. To use the software that is already installed Debian Linux, NFS, NIS, SSH and FFTW.
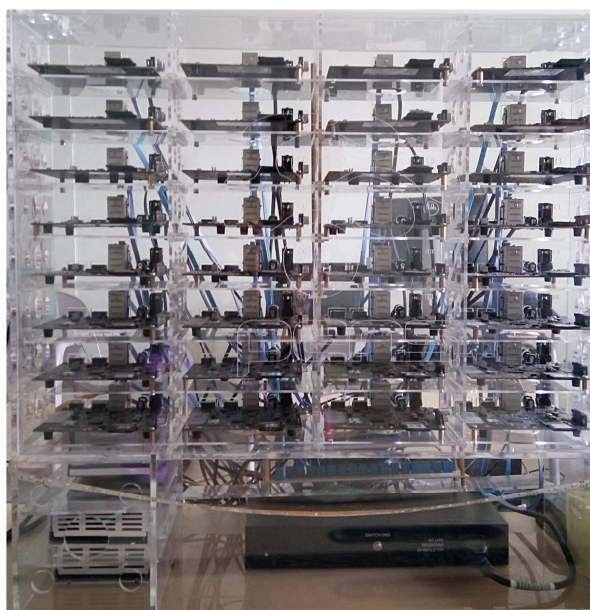
**Figure 8.** Cluster Parallel Computing

5. EXPERIMENT AND ANALYSIS

The System is split into two subsections in developing embeded system: MPI and management program and networking. The number of students assigned to each group to build embeded cluster system. Two student assigned the mpi and management program and five students are focus on networking communication. In this section the subsystem qualifications are explained followed by their evaluation procedures.

A. System Testing and Analysis

There are two testing cases within this project. The first is Image Fourier Transform, and the second case is benchmark testing to showing the performance of Embedded system for High Performance Computing. As the final evaluation at the system level, students are required to demonstrate the working system for fourier transform and to record the performance graph Fig. 10 to be analyzed. And as second case Fig. 11 shows a snapshot of the benchmark demo.

1. Image Processing testing and Analysis

In this experiment, we implement a numerical computation for testing the performance of our embedded system. In this experiment we try to implement fourier transform to an array and divide the array in to 32 process. Node who control the main process called as server and scatter the piece of data to 31 node, after the parallel fourier transform process in each parallel processor finished, the server will gather the data and construct dataset for final result from gathered data. The steps of parallel fourier transform shown in Figure 6.
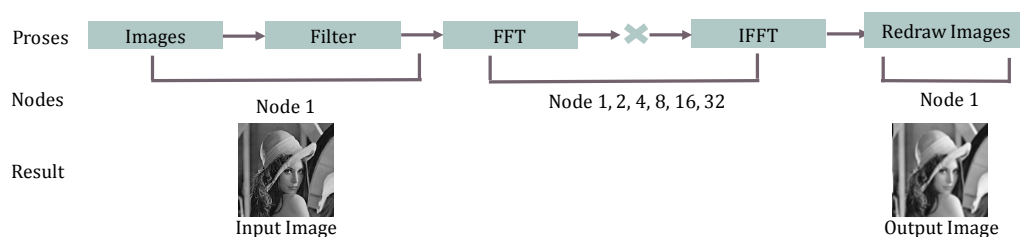

Figure 9. Image processing diagram flow

Figure 9. Shows diagram flow of image processing experiment, the process start by deviding the process for MPI. We use the dataset from a bmp images and read the RGB of input images and assign to an array. Thus array is a new dataset that will be divide into 31 of process and distribute to different processor. As for the experiment process, fourier transform runs in incremental processor, the experiment has been done in 6 times experiment. Start from process 1, 2, 4, 8, 16 and 32, where the experimental result shown in Figure 8.

The experiment has succefully shows the impact of deviding job and process in multi processor. It is shown that with the increasing of the total

number of processor could shortened the computational time. The main control process happens in server node which has a job to deviding the number of process and scatter the grayscale colour value to 2, 4, 8, 16, and 32 processors. Process node will execute the fft to those part of pixel data that sended from server node. After execute the FFT the next step is to do the filtering process in frecuency domain and then run the IFFT to return the number from FFT to its original data. After these two process, the result will be sended back to and gathered by server node. In Last step, server node will construct the data to be an intact data. Computational time is used as measurement parameter, the minimum time taken to process indicating the better performances. To compare the performance of our embedded System, we also run our program in one node using PC with i5 as the processor. And the result of the experiment could be shown in Figure 10.
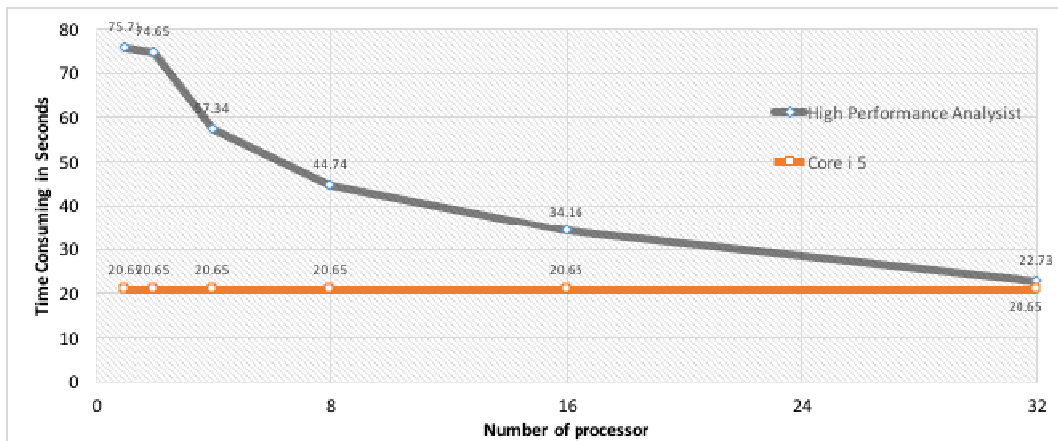
**Figure 8.** Image Processing Performance in High Performance Embedded System

2. NAS Paralel Benchmark

The NAS Parallel Benchmarks (NPB) are a set of programs designed to help evaluate the performance of parallel supercomputers. As for this embedded system we use the class A and B as for the standart test problem. Details of NAS Parallel Benchmark is represented in Table 2. This NAS parallel benchmark runs in incremental processor, the experiment has been done in 6 times experiment. Start from process 1, 2, 4, 8, 16 and 32.
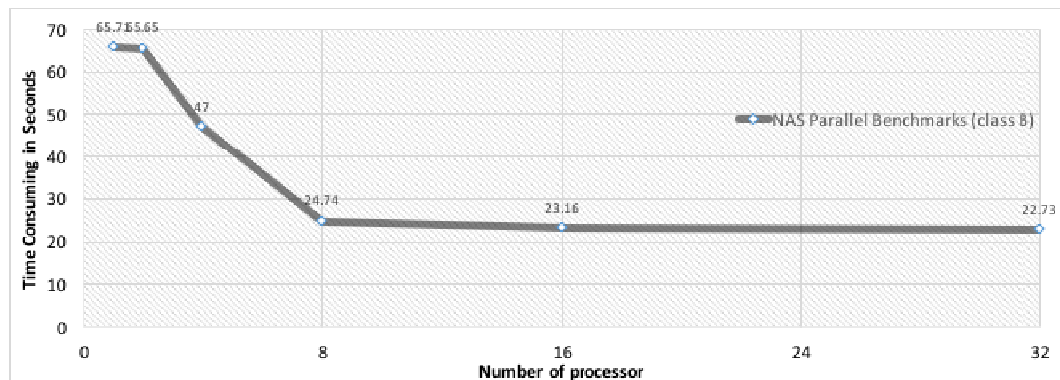
Table 2. NAS Paralel Benchmark

Version	Benchmark Included	Benchmark Classes	Programming Models Used)
Version 3.3.1	IS	A	MPI
Version 3.3.1	IS	B	MPI Serial

The performance of this benchmarks determined by the computational time process as shown in Figure 11. As for first experiment in running the NAS Parallel Benchmark, we use the IS benchmark with A classes which stated as fully parallel program. Classes A, B: standard test problems with 4X size increase going from one class to the next. Figure 11 show the performance of embedded system, the more processor used the smaller computational time will be taken. Because the class A in IS Benchmark is fully MPI, increasing the number of processor to process will be highly effect on computational time.
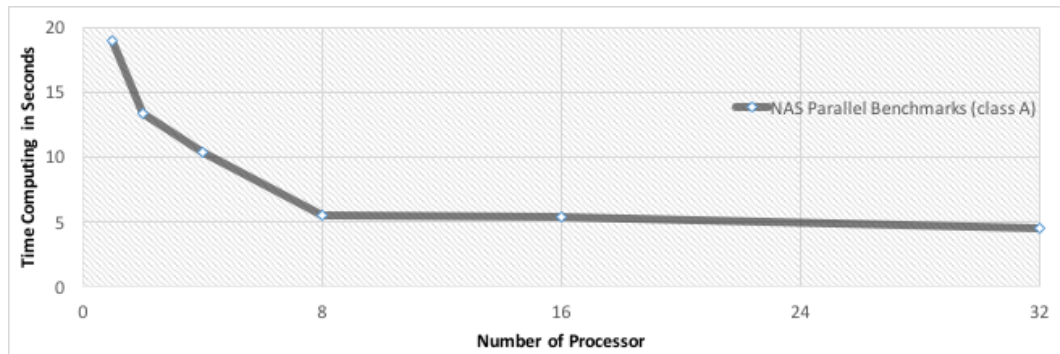
Figure 11. NAS Paralel: IS Benchmark, class A

The second benchmark testing, incremental processor also used to analyse the performance of embedded system in running parallel task. In this second classes, it is use the mechanism of semi parallel program. Figure 12. Showing that the number of processor also impact the computational time.

Figure 12. NAS Semi Paralel Benchmark: IS Benchmark, class B

B. Course Learning Objectives

All students are evaluated at the end of the semester of High Performance Computing course. This evaluation based on five qualitative criteria: course material, motivation, creativity, learning process and feedbacks. Seven students involved in this course. The student's course evaluation is gathered through an anonymous survey form. Students are asked questions related to their personal opinion and experiences. IS focused on assessing this course

effectiveness in terms of program and course outcomes. The detailed evaluation is ilustrated in Table 3.

Table 3. Course assesment criteria

No.	Questions	Score (%)
Course materials		77
Q1	The completeness of the materials.	
Q2	This course was interesting.	
Q3	The course materials were suitable for learning model.	
Q4	The evaluation model for learning model.	
Motivation		100
Q5	Student's knowledge about High Performance Computing (HPC) in the beginning of course.	
Q6	Student's knowledge about High Performance Computing (HPC) application in the beginning of course.	
Creativities		100
Q7	I will recommend this course to other students.	
Q8	I have a better understanding of the functions of each telecommunication building block and of the system as a whole.	
Learning processes		83
Q9	Lecturer knowledge on learning and answering questions from the students.	
Q10	The use of HPC modules use as a learning medium for parallel computing course.	
Feedbacks		100
Q11	The results of the development of HPC modules.	
Q12	The results of the learning process on parallel computing course.	

As the data in Table 3 is based on five qualitative criteria: course on the material students almost pick at point C which is an average of 62%, on student motivation almost pick at point A is with an average of 46%, on the creativity of the students almost choose at point C which is an average of 50%, in the learning process of students almost pick at point C which is an average of 48% and the input of students almost pick at point C which is an average of 38%, each rated at 77%, 100%, 100%, 83% and 100% of the total grade, respectively. The survey relates to every student reacts and learn lessons for High Performance Computing. Also 100% of participants reported that the High Performance Computing experience has increased their interest in clustered computing.

Althought the survey score considered to be very favorable, the result survey show that future improvements can be accomplished by considering student as center of learning by working independently in a teamwork attitude. Future projects can therefore be designed to enhance student

knowledge of these aspect. Table 4. Shows the comparison between Single PC, Server, and Embedded System as a High Performance.

Table 4. HPC Hardware Comparison

Comparison Factors	Single PC	Servers	Embedded System
Cost n node	$ 600	n x $2000	n x $200
Node	Max 4 nodes using Virtual Box	n	n
Space	narrow	wide	narrow
Power	500 W	n x 1000W	n x 15W

n = number of node

6. CONCLUSION

In this paper, we implemented High-Performance Computing (HPC) learning using of embedded systems. We design the architecture of cluster embedded system and learning model where present an innovative cluster system to support HPC education learning activities in HPC course with small size, low cost, and yet powerful enough. In this experiment, we have two testing for analysis HPC, first is Image Fourier Transform, and the second case is benchmark testing and also we evaluate all student at the end of the semester of HPC course. This evaluation based on qualitative criteria where student's course evaluation is gathered through an anonymous survey form.

This system is a combination of technical and soft skill resulting a high knowledge understanding from classroom learning. From the empirical assessment and student survey, the implemented educational activities satisfied the intended program learning outcomes and course outcomes.

REFERENCES

[1] Clay Carley, Larry Sells, Brett McKinney, Chao Zhao, and Henry Neeman,**Using a Shared, Remote Cluster for Teaching HPC**, *2013 IEEE International Conference on Cluster Computing (CLUSTER),* Indianapolis, pp. 1-6, 2013.

[2] Omar Abuzaghleh, Kathleen Goldschmidt, Yasser Elleithy, and Jeongkyu Lee, **Implementing an Affordable High-Performance Computing for Teaching-Oriented Computer Science Curriculum**, *Journal ACM Transactions on Computing Education (TOCE)*, Vol. 13(1), No. 3, 2013.

[3] Paweł Czarnul, **Teaching High Performance Computing using BeesyCluster and Relevant Usage Statistics**, *Proceedings of the 4th International Conference on Computational Science*, Cairns, Vol. 29, pp. 1458-1467, 2014.

[4] Leo Nagamatsu, Daisuke Suwa, **Searching Aimlessly Running Processes on Classroom Workstations in the University of Aizu,** International Conference Parallel Processing, Aizu-Wakamatsu, Japan, pp. 418 – 423, 1999.

[5] Amy Aporia, Rajkumar Buyya, Hai Jin, and Jens Mache, **Cluster Computing in the Classroom: Topics, Guidelines, and Experiences**, Proceedings First IEEE/ACM International Symposium on Cluster Computing and the Grid, Brisbane, Qld, pp. 476 - 483, 2001.

[6] Amit C, Gurvinder S, **A Cluster Based Parallel Computing Framework (CBPCF) for Performance Evaluation of Parallel Applications**, IEEE International Journal of Computer Theory and Engineering, Vol. 2, No. 2 April, 2010.

[7] Nanjesh B R, Madhu C K, Vinay Kumar K S, Hareesh Kumar G, **MPI Based Cluster Computing for Performance Evaluation of Parallel Applications**, Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013), JeJu Island, pp. 1123-1128, 2013.

[8] Mohd Noor Ikhwan A R, Lucyantie M, Syed Farid S A, **Analysis on Parallelism Between CPU And GPGPU Processing On Cluster Computing**, IEEE 2014 Symposium on Computer Applications & Industrial Electronics (ISCAIE), Penang, Malaysia, pp. 203-207, 2014.

[9] Mingliang Xu, Qianmin Su, **The Realization of Small Cluster Parallel Computing Environment for College Education**, The 9th International Conference on Computer Science & Education (ICCSE 2014), Vancouver, Canada, pp. 861-863, 2014.

[10] Pfalzgraf, A.M. and Driscoll, J.A, **A low-cost computer cluster for high-performance computing education**, Electro/Information Technology (EIT), 2014 IEEE International Conference, Milwaukee, WI, pp. 362 - 366, 2014.

[11] Ilias Stavrakas, Theophanis Kontogiannis, Marios Panaousis, Maria Kaitsa, Ilias Daradimos, Theodoros Georgiou, George Blathras and Dimos Triantis, **Beowuif Clusters for Parallel Programming Courses**, EUROCON 2005 - The International Conference on "Computer as a Tool", Belgrade, pp. 791 - 794, 2015.

[12] S. Sukaridhoto, D. Pramadihanto and A. S. KHalilullah, "**Further Investigation on Building and Benchmarking A Low Power Embedded Cluster for Education**", IPTEK Journal of Proceedings Series, vol. 1, 2015, pp. 311-318.

[13] Sparsh Mittal, **A survey on techniques for improving the energy efficiency of large-scale distributed systems**, *Journal ACM Computing Surveys (CSUR)*, Vol. 46(4), No. 47, 2014.

[14] M. Bruri Triyono, **Enhancing The Learning Quality through Implementing Instructional Design for Vocational Education**, *International Conference on Vocational Education and Training (ICVET)*, Yogyakarta, pp. 129-134, 2011.