PRODUCTIVE CO PROCESSOR DESIGN BASED ON PROGRAM BENCHMARK

[1] P. S. BALAMURUGAN & [2] Dr. K.THANUSHKODI,

[1]RESEARCH SCHOLAR, ANNA UNIVERSITY, COIMBATORE
[2] DIRECTOR AKSHAYA COLLEGE OF ENGINEERING AND TECHNOLOGY, COIMBATORE

.

Abstract -**The objective of this paper is to design a methodology where many co-processors are accessed by the processor in array mode. By using co processor, the work on the multi core processor gets reduced by accessing it in array manner. A multi core processor is an efficient processor which can enable parallel processing and perform multi threading effectively. In this paper, in order to improve the performance of multi-core processor two major factors are taken into consideration one is to improve the execution of array methodology by using co processor and other is yo design an array based co processor to improve the hit ratio of the co processor**

*Keywords–**Co-Processor, DSP Processor,* **Array Processor.**

I. INTRODUCTION

A new methodology is been proposed known as array based co processor design in which a FPGA processor which has a better efficiency in processing Network and Document Benchmark Programs will be connected to the Processsor as Co Processor. For efficient processing of data all the Graphical Benchmark Programs will be processed by the Multi core processor and it will have a array Methodology coding which will analyze the data and if the data is Network or Document Benchmark program it will make the co processor to process it. By implementing this methodology the load on processor will be reduced and the efficiency of processing will be increased

II . CO PROCESSOR DESIGN USING ARRAY METHODOLOGY

The working of a processor can be explained with the block diagram as shown in Figure 1.which gives an idea about how Process interacts with memory in bidirectional manner. The array Co Processor is embedded with array methodology coding the coprocessor is used by the processor for memory allocation and to interact with the RAM in array manner. By this methodology the processor will interact with the RAM device in a array passion by which the RAM will be divided into many arrays and each array will be allotted for a default program to be utilized thus the processor can access the data and codes easily by searching in the specified memory location. When an need arises where a program need more memory space than the allotted memory at this case by using artificial intelligence we an combine the memory and utilize to executed the program.

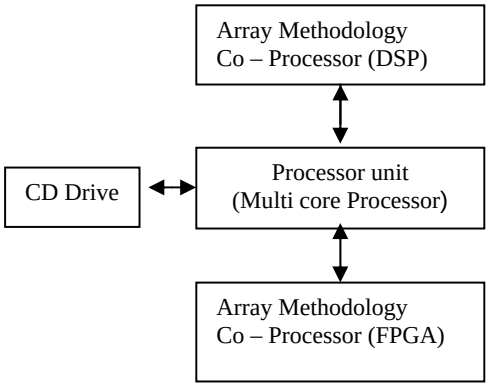

Figure 1. Block Diagram of Array methodology Based Co- Processor Design

This method is highly applicable when we want to run a small program and the system RAM is of high capacity. During this condition to make the work of processor simple by allowing it to search in allotted array methodology can be implemented. The efficiency can be known by the HIT ratio of the Cache hit ratio refers to the ratio between no of times the cache memory is accessed to the no of times data is accessed by the processor.

To enhance this operation of accessing Cache in array manner and implement array based processing a specially designed coprocessor is designed to implement the application. To derive an equation to find the configuration of co processor many Processor has been interfaced to the multi core processor and the performance is noted down .A PIC processor is been interfaced with the multi core processor and it is made to be dedicated for Array based processing and its performance was measured.

A multi core processor with 2 GHz speed is selected and it is made to interface with a FPGA processor the FPGA is dedicated for processing only the array cache and array methodology coding. Many FPGA processors with different

processor speed is taken for testing as a first step a FPGA processor with 2 GHz speed is taken and Time taken for array .To measure the performance of different multi processor interfaced with co processor. Multi core of different clock speed is chosen and its performance with different Co Processor configuration is tabulated. First the time consumed to execute a program without using array methodology is tabulated

By spacing the memory in array manner the cache port can be accessed in an efficient manner. This can be known from the percentage of hit ratio tabulated for different programs

Table 1. Performance of Processor using Graphical and Networking Benchmark Programs

Program Specification	Physical capacity of program	Time consumption in seconds
Graphical Bench	1.2GB	0.58
Graphical Bench	1.14 GB	0.52
Graphical Bench	1.10 GB	0.52
Graphical Bench	1.04 GB	0.51
Graphical Bench	1. GB	0.5
Graphical Bench	850 MB	0.62
Graphical Bench	780MB	0.65
Networking Benchmark	790 MB	0.65
Networking Benchmark	1.05 GB	0.51
Networking Benchmark	1.08 GB	0.52
Networking Benchmark	875 MB	0.62
Networking Benchmark	805 MB	0.62

These are the performance of a multi core processor to improve the time factor in executing the graphical bench mark program a co processor is designed using a DSP processor .The design of DSP co processor is to be that the sum of all the clock speed of co processor should be equal to the clock speed of the processor to enhance synchronization. The no of co processor depends on the speed of the processor. In other words if P represents the speed of the processor which has 4 co processor and the co Processor speed is by Cp1 , Cp 2 , Cp 3 ,Cp4 .The seed of co processor is derived from the equation

$$P = Cp1 + Cp2 + Cp3 + Cpn$$

The same programs were taken and executed in a multi core processor. First the processor in which the performance is measured is taken. The clock speed of the processor is noted down and accordingly the speed of co processor is designed the mother processor which is taken is 2 GHz clock speed processor .First Two Co processor is implemented with a clock speed of 1 GHz each

Table 2. Performance Comparison of Processor using Document Benchmark Programs Executed by Processor and Co Processor

Program Specification	Physical capacity of program	Time consumption in seconds	Time in seconds When co processor is used
Document Bench	300 MB	0.64	0.63
Document Bench	180MB	0.95	0.96
Document Bench	150 MB	0.95	0.96
Document Bench	144 MB	1.11	1.12
Document Bench	135 MB	1.1	1.1
Document Bench	80 MB	1.31	1.30
Document Bench	75 MB	1.31	1.32

Figure 2. Shows the Program Size with respect to time consumption in seconds to execute a Document Benchmark program by multi core processor and DSP Based Co Processor

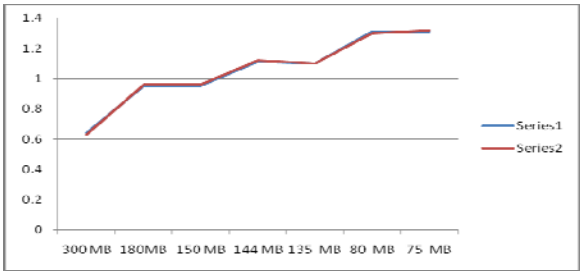

Figure 2. Document Benchmark by multi core processor and DSP Based Co Processor

Table 3. Performance Comparison of Processor using Graphical Benchmark Programs

Program Specification	Physical capacity of program	Time consumption in seconds	Time consumption in seconds by using co processor
Graphical Bench	1.2GB	0.58	0.50
Graphical Bench	1.14 GB	0.52	0.45
Graphical Bench	1.10 GB	0.52	0.45
Graphical Bench	1.04 GB	0.51	0.51
Graphical Bench	1. GB	0.5	0.45
Graphical Bench	850 MB	0.62	0.58
Graphical Bench	780MB	0.65	0.58

The performance with respect to Program Size and time consumption in seconds is to execute a Graphical Benchmark program by multi core processor and DSP Based Co Processor is shown in figure 3.

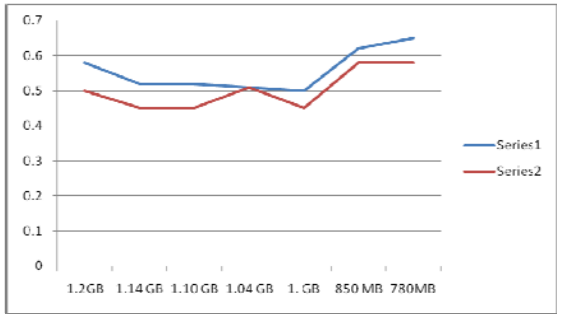

Figure.3. Graphical Benchmark program by multi core processor and DSP Based Co Processor

Networking Benchmark is also processed in the same manner and tabulated as below and Figure 4. shows the Performance Comparison of Processor using Graphical Benchmark and networking Benchmark Programs Executed by Processor and Co Processor

Table 4. Performance Comparison of Processor using Networking Benchmark Programs

Program Specification	Physical capacity of program	Time consumption in seconds	Time consumption in seconds using Co processor
Graphical Bench	1.2GB	0.58	0.57
Graphical Bench	1.14 GB	0.52	0.52
Graphical Bench	1.10 GB	0.52	0.52
Graphical Bench	1.04 GB	0.51	0.50
Graphical Bench	1. GB	0.5	0.5
Graphical Bench	850 MB	0.62	0.62
Graphical Bench	780MB	0.65	0.65
Networking Benchmark	790 MB	0.65	0.67
Networking Benchmark	1.05 GB	0.51	0.51
Networking Benchmark	1.08 GB	0.52	0.54
Networking Benchmark	875 MB	0.62	0.62
Networking Benchmark	805 MB	0.62	0.62

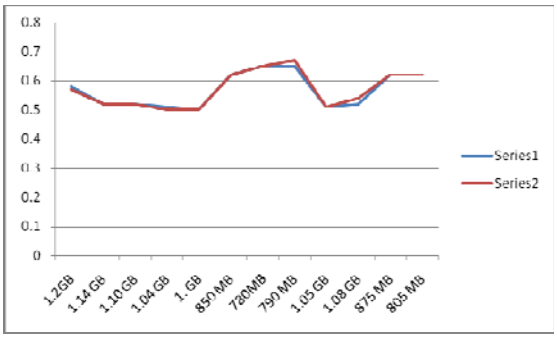

Figure 4. Graphical Benchmark and Network Benchmark programs by multi core processor and DSP Based Co Processor

Keeping this performance into consideration the processor is coded with a comparison for file extension using the benchmark By using this methodology the processor will utilize only for Graphical Benchmark Programs by this methodology graphical programs are executed with much clarity and in a time efficient manner . This coding also reduces the work load of the processor. The Processor is coded with this methodology and the readings are

Table 5. Performance Comparison of Processor using Graphical Benchmark and networking Benchmark Programs

Program Specification	Physical capacity of program	Time consumption in seconds	Time consumption in seconds using Co processor
Graphical Bench	1.2GB	0.58	0.57
Graphical Bench	1.14 GB	0.52	0.52
Graphical Bench	1.10 GB	0.52	0.52
Graphical Bench	1.04 GB	0.51	0.50
Graphical Bench	1. GB	0.5	0.5
Graphical Bench	850 MB	0.62	0.62
Graphical Bench	780MB	0.65	0.65

From the above reading it is been enumerated that the co processor designed by DSP works much effective for Graphical Benchmark programs.

A Multi core and FPGA Processor Performance

To enumerate the performance of different processor with graphical benchmark programs the program is made to execute in a FPGA processor first and the time to execute the program is noted down programs with different benchmark programs. This will give a clear Idea about which programs are to be routed to which co processor while the processor is busy.

1)Processor Performance with Graphical Benchmark Programs

The performance of Different Benchmark programs are analyzed using multi core processor which acts as main processor and the results are compared with the results obtained when same programs are executed with FPGA Processor.

Table 6. Performance of Graphical Benchmark Programs in Multi core Processor and FPGA Processor

Program Specification	Program Size	Time consumption in seconds with Multicore processor	Time consumption in seconds with FPGA processor
Graphical Bench	1.2GB	0.54	0.59
Graphical Bench	1.14 GB	0.42	0.52
Graphical Bench	1.10 GB	0.42	0.53
Graphical Bench	1.04 GB	0.42	0.51
Graphical Bench	1. GB	0.39	0.45
Graphical Bench	850 MB	0.38	0.45
Graphical Bench	780MB	0.38	0.43

Using the above values a graph is been plotted between program size and the time consumed to process it by multi core processor and in the same manner graph is plotted for the program size and processing speed by using FPGA is shown in Figure 5. and Figure 6. respectively.

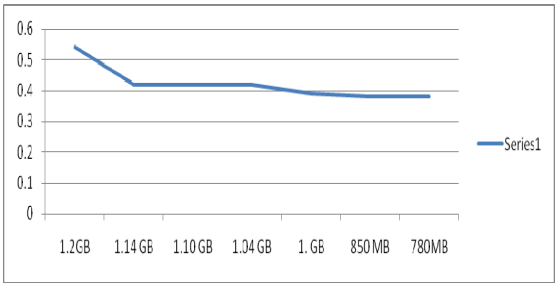

Figure 5. Graphical Benchmark program by using multi core processor

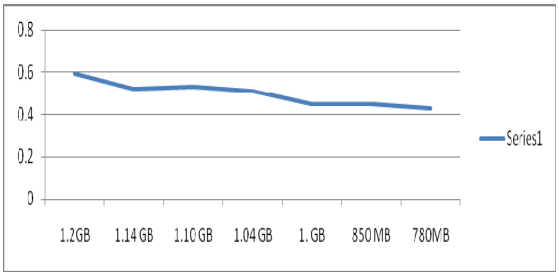

Figure 6. Graphical Benchmark program by using FPGA processor

Using both the graph it can be concluded that the multicore processor processes data in an average time of 0.422 and the Field proogrammable gate array has consumed an average time of 0.49 . Now both the time line

graph for multi core processor and FPGA is compared to have a clear view of the processor and is shown in Figure 7.

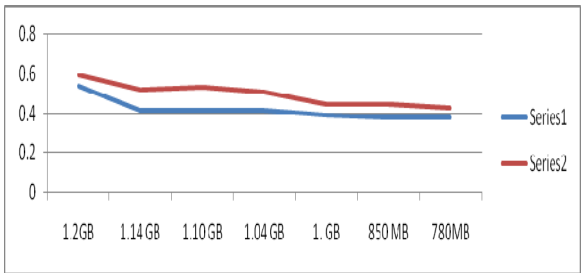

Figure 7. Graphical Benchmark programs Performance by using multi core processor and FPGA processor

The comparison graph gives a conclusion that the multicore processor has a better efficiency than the FPGA processor for graphical benchmark programs .

2) Processor Performance with Networking Benchmark Programs

The same process of testing can be carried out with networking benchmark programs to analyze the performance of FPGA processor with networking Benchmark programs.

Table 7. Performance of Network Benchmark Programs in Multi core Processor and FPGA Processor

Program Specification	**Program Size**	**Time consumption in seconds with Multi-core processor**	**Time consumption in seconds with FPGA processor**
Network Benchmark	790 MB	0.64	0.62
Network Benchmark	1.05 GB	0.52	0.47
Network Benchmark	1.08 GB	0.53	0.50
Network Benchmark	875 MB	0.60	0.57
Network Benchmark	805 MB	0.62	0.60

To analyze the performance of FPGA with network Benchmark programs different programs are made to run in multi-core processor and FPGA processor and the time consumed to process is tabulated and its pictorial view is shown in Figure 8.

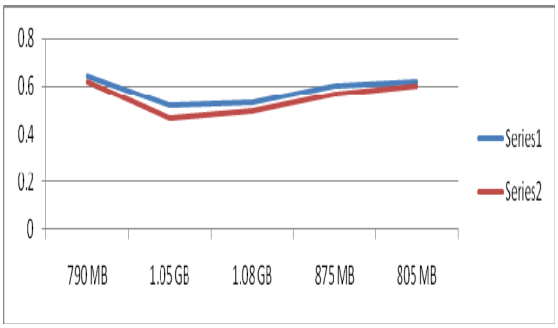

Figure 8. Network Benchmark programs Performance by using multi core processor and FPGA processor

Using the same procedure both the processor is been tested for document benchmark programs and the results are tabulated and also a graph is drawn to analyze the perfomance

Table 8. Performance of Document Benchmark Programs in Multi core Processor and FPGA Processor

Program Specification	**Program Size**	**Time consumption in seconds with Multi-core processor**	**Time consumption in seconds with FPGA processor**
Document Bench	300 MB	0.99	0.84
Document Bench	180MB	0.95	0.81
Document Bench	150 MB	0.95	0.8
Document Bench	100 MB	0.86	0.74
Document Bench	80MB	0.85	0.74

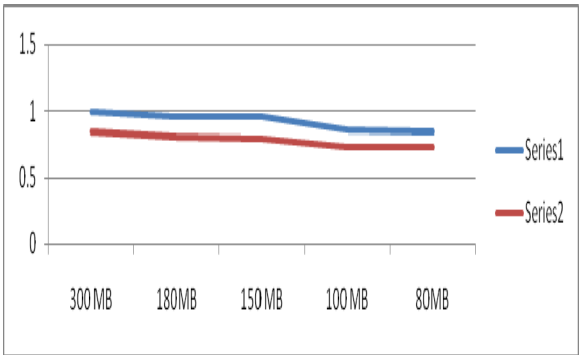

Figure 9. Document Benchmark programs performance by using multi core processor and FPGA processor

Analysing the perfomance for both the processor in graphical and desktop benchmark programs the FPGA processor performs better than multi core processor in terms of time cosumption.

REFERENCES

[1] E. Rohou and M. Smith, Dynamically managing processor temperature and power,1999, *Proc. FDDO-2*.
[2] S. I. Souri, K. Banerjee, A. Mehrotra, and K. C. Saraswat, Multiple Si layer ICs: Motivation, performance analysis, and design implications,*Proc Design Automation Conf*., 2000, pp. 873- 880.
[3] S. Das, A. Chadrakasan, Three-dimensional integrated circuits: performance, design methodology, and CAD tools, *Proc. IEEE Annual Symp. on VLSI*, 2003, pp. 13-18.
[4] M.B. Kleiner, S. A. Kuhn, P. Ramm and W. Weber,Thermal analysis of vertically integrated circuits, *Tech. Dig. Int'l Electron Device Meeting*, 1995, pp.487-490.
[5] A. Rahman, R. Reif, Thermal analysis of three dimensional (3-D) integrated circuits (ICs), *Proc.Int'l Interconnect Technology Conf.* 2001, pp. 157-159.
[6] F. Fallah and M. Pedram, Standby and active leakage current control and minimization in CMOS VLSI circuits. *IEICE Trans. On Electronics*, Special Section on Low-Power LSI and Low-Power IP, Vol.E88-C, No. 4, 2005, pp. 509-519.
[7] B. Chatterjee, M. Sachdev, S. Hsu, R Krishnamurthy, and S. Borkar, Effectiveness and scaling trends of leakage control techniques for sub-100nm CMO technologies, *Proc. Int'l Symp. Low- power Electronics*, 2003, pp. 122-127.
[8] M.L. Mui, K. Banerjee, A. Mehrota, Power supply optimization in sub-130nm leakage dominant technologies, *Proc. Int'l Symp. Quality Electronic Design*, 2004, pp. 409-414.
[9] Int'l Technology Roadmap for Semiconductors (ITRS), 2004.
[10] K. Banerjee, A. Mehrotra, A. Sangiovanni-Vincentelli, and C. Hu ,On Thermal Effects in Deep Sub-Micron VLSl Interconnects, *Proc. Design Automation Conf.* 1999, pp. 885-891.