IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-6, Issue-4 E-ISSN: 2347-2693

Performance Estimation of Scheduling Algorithms on Microblaze Softcore Processor

Manisha Das[1*], S.V.Admane[2], Jitendra B. Zalke[3]

[1*]Electronics Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur, India
[2] Electronics Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur, India
[3] Electronics Design Technology, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

[*]*Corresponding Author: dasmg1@rknec.edu, Tel.: 9975981668*
Available online at: www.ijcseonline.org

Received: 16/Mar/2018, Revised: 24/Mar/2018, Accepted: 12/Apr/2018, Published: 30/Apr/2018

Abstract— Multitasking or multiprocessing is the need of today's modern embedded system. Multitasking can reach the desired increase in performance as needed by today's embedded systems. Multitasking can be achieved by scheduling the competing tasks and allocate them to the shared resources one by one at very high speed (so fast and frequently) so as to create an illusion that all tasks are running in parallel. There are various algorithms to implement a scheduling scheme depending upon the need and constraints of the application. The proposed research work aims at evaluating and comparing the performance of Round Robin and Static Priority Scheduling algorithm by implementing them the on single Microblaze soft core processor and porting xilkernel on FPGA platform.

Keywords—Microblaze, Xilkerenel, Scheduling, FPGA

I. INTRODUCTION

An embedded system is an electronic/electro-mechanical system designed to perform a specific function. It's a Combination of Hardware and Software dedicated to an applications. Today's world demand for embedded devices having, high speed, high performance, low power, etc. It includes an advanced applications like image processing, audio/video encoding/decoding, network application and high reliability. To achieve these goals, there is a need to design and developed an efficient software algorithms which are efficiently executed on hardware. Since process of changing the hardware to achieve desired performance goal as per the applications is complex and expensive further it is a time consuming process which can increase the time to market of the product.

In some applications as needed, the process of upgrading / changing the software is flexible, fast and financially beneficial as compared the process of upgrading / changing hardware.. Multi-programming is one of the most important aspects of operating systems. The key to Multi-Programming is scheduling of the task. Efficient scheduling can be achieved by effectively allocating the resources that are shared by the competing tasks as and when required. Multi-programming is attained by the efficient task scheduling of the CPU. Selecting and optimizing an appropriate combination of hardware and software platforms can meet the escalation in performance for an embedded system. The proposed research work is targeted to explore and analyze the performance of round robin and static priority task scheduling algorithm on FPGA based programmable platforms. A MicroBlaze soft-core processor based system is designed and implemented on FPGA platform. The xilkernal is ported on soft-core processor embedded system. The multiple tasks are created and scheduled using round robin and static priority scheduling algorithm and tested for its performance. The performance of these two algorithms is evaluated on the basis of predefined performance parameters.

II. RELATED WORK

Mohammad T. Kawser et al.[1] analysed the performance of promotionally fair and round-robin scheduling algorithm for transmission mode in long-term evaluation to observe the impact of the scheduling algorithm on channel throughput and fairness. It was concluded that for different modes of transmission proportionally fair algorithm outperforms round robin scheduling algorithm. Rukhsar Khan et al. [2] analysed the performance of shortest job first and first come first serve scheduling algorithm by implementing them on windows operating system and Turbo C. It was concluded that SJF outperforms FCFS algorithm on the basis of their timing performance. Alban Allkoci et al. [3] in their work compared the performance of round robin and priority scheduling algorithm. C language was chosen to develop the codes and the hardware platform used as a personal computer. It was

concluded that priority algorithm outperforms the round-robin algorithm in terms of timing performance. Lalit Kishor et. al. [5] compared frequently used scheduling algorithms and also proposed one new scheduling algorithm by combining shortest job first and round-robin algorithm which yielded better timing performance. The proposed work aims at first designing a processor environment on FPGA hardware platform with the help of soft-core processor environment, porting kernel followed by designing multitasking applications and running them using round robin and static priority scheduling algorithm. The quintessence of the work done lies in analysing the performance of multitasking applications environment by estimating the performance of scheduling algorithms on a flexible hardware platform.

A. Multitasking

Multitasking refers to the execution of multiple processes / multitasks simultaneously. It is based on the ability of OS to suspend and resume the tasks executing concurrently. An OS must allocate resources amongst the competing process. The application is a user instantiated program and the different layers of an application program are as shown in Figure.1.

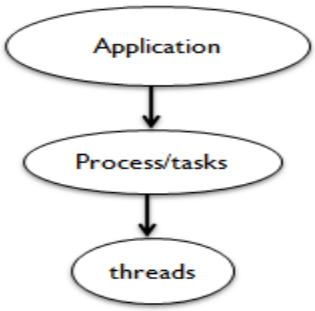

Figure.1 Application Program Layers

B. Scheduling

Scheduler is Part of the OS responsible for determining which task will wait and which will progress. It assigns processes to be executed by the processor over time. The allocation of the CPU resources is done by an algorithm known as the Scheduling algorithm. There are various algorithms by which scheduling can be achieved such as First in First out, Round-Robin, Fixed priority scheduling, Earliest deadline first scheduling, Rate-monotonic scheduling etc. In each of this scheduling algorithm, the scheduler assigns resources to different competing process decided by the inherent algorithm. In Round-robin scheduling every task is assigned a fixed time by the scheduler and each process run for the specified amount of time and then the next process runs followed by next process and the cycle goes on. In fixed priority scheduling algorithm, the OS assigns a fixed priority rank to every process. The scheduler then arranges them in the ready queue according to their priorities. Whenever a higher priority process arrives it interrupts the ongoing lower priority process and resources are allocated to it. The performance of any scheduling algorithm is evaluated on the basis of following factors;

a) Throughput: Number of processes executed per time unit.

b) Turnaround time: Total time between submission of a process and its completion.

c) Response time: Amount of time it takes to produce first response, from the instant when a request was submitted.

d) Waiting Time: The time the process remains in the ready queue.

e) Fairness: Equal CPU time to each process (or more generally appropriate times according to each process priority and workload).

f) CPU Overhead: Inter-task communication.

g) CPU utilization: The maximum use of CPU when it is busy.

C. Softcore Processor

A soft-core processor is a microprocessor fully described in software, usually in an HDL, which can be synthesized for programmable hardware, such as FPGAs [7].

Advantages of Soft Core Processors:

- Higher level of abstraction – easier to understand
- More flexible – designers can change the core by editing source code or selecting parameters.
- Platform independent – can be synthesized for any IC technology, including FPGAs, ASICs, etc.

The MicroBlaze embedded soft-core as shown in Figure. 2 is a reduced instruction set computer (RISC). It is optimized for Xilinx FPGA implementations. It supports a variety of RTOS implementation. It can perform high-Frequency operation [9].

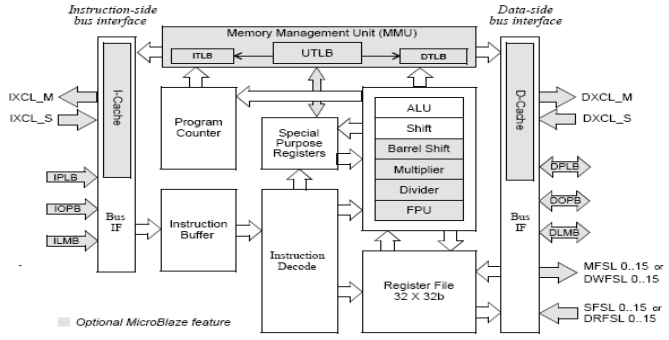

Figure.2 MicroBlaze Core Block Diagram [11]

D. *Xilkernel*

Xilkernel is a small, robust embedded kernel that provides scheduling multiple execution contexts and a set of higher level services [10].

1) Processes
2) Threads
3) Context Switching
4) Scheduling
5) Synchronization
6) Interprocess Communication.

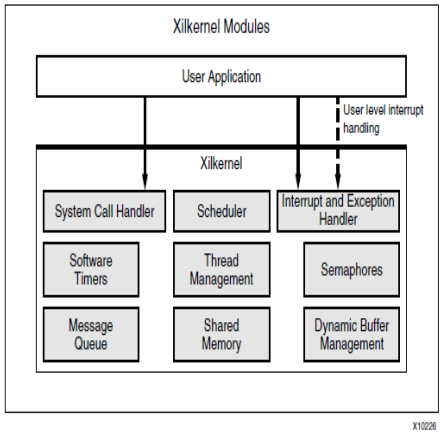

Figure.3 Xilkernel Modules [12]

III. METHODOLOGY

Soft-core processor environment is implemented on FPGA (Field Programmable Gate Array) based development platform. FPGA used for hardware implementation is Virtex 5 – XUPV5-LX110T (Xilinx micro-processor Virtex 5 LXT: High-performance logic with advanced serial connectivity). The XUPV505-LX110T is a feature-rich general purpose evaluation and development platform with onboard memory and industry-standard connectivity interfaces. It features the Virtex-5 XC5VLX110T device. Peripherals available on board like LED, LCD display, serial port etc can be used depending on the need of the application. Also, general purpose input output (GPIO) pins are available for external hardware connection. Initially, single MicroBlaze processor system is developed for serial communication through UART (Universal Asynchronous Receiver Transmitter). The processor was Built along with block RAM, timer0, bus control PLB (processor local bus), LMB and Serial port access Using Xilinx ISE 13.1 based EDK tool; XPS (Xilinx platform studio). Xilkernel OS is made to run on MicroBlaze soft-core processor which is implemented in FPGA. Xilkernel OS is also ported on the soft-core processor based system. The system timer clock frequency is set to be 50MHz. The system timer interrupt interval is 200ms. Using this, we can handle scheduling of different tasks. Multi-tasking can be done depending on task scheduling policies like Round robin and Priority based. Single soft-core based processor environment using Micro-blaze was developed and Xilkernel OS was ported on it to run applications which can further be divided into smaller task/threads.

Thus dividing time consuming applications into small threads and computing them in parallel will improve the speed of operation and reduce the overhead. Four threads were created using xilkernel functions.

Thread1: counts from 0 to 5 and sleeps for 9000ms
Thread2: counts from 0 to 10 and sleeps for 8000ms
Thread3: counts from 0 to 15 and sleeps for 7000ms
Thread4: counts from 0 to 20 and sleeps for 1000ms.

Thread1 is assigned highest priority followed by thread2 followed by thread3 and finally thread4 who is assigned lowest priority. These four threads were scheduled using round robin and fixed priority algorithm. The threads were run on the hardware platform

IV. RESULTS AND DISCUSSION

The response time and average turnaround time for all the tasks were calculated separately for both the scheduling algorithms. The results are as shown below in tabular form:

Table 2. Results for response time

Thread	Response(timer ticks)	
	Round robin algorithm	***Priority algorithm***
T1	1	1
T2	2	9
T3	3	22
T4	4	41
Average response time	2.5	18.25

Table 3. Results for turnaround time

Thread	Turnaround time(timer ticks)	
	Round robin algorithm	***Priority Algorithm***
T1	26	8
T2	49	21
T3	64	40
T4	74	114
Average turnaround time	53.25	45.75

Table 4. Results for throughput

Throughput	(tasks/sec)
Round robin algorithm	0.375
Priority algorithm	0.437

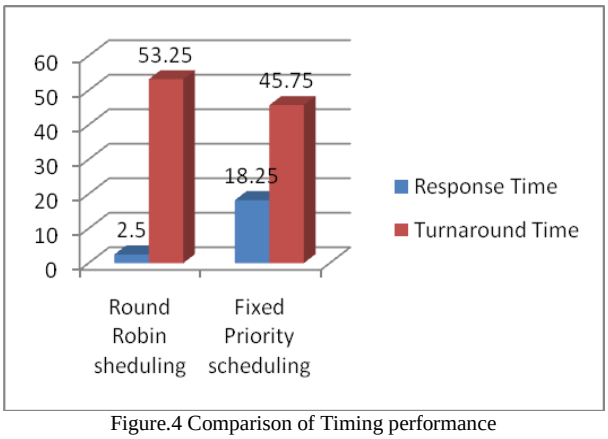

Figure.4 Comparison of Timing performance

V. CONCLUSION AND FUTURE SCOPE

By performing the timing analysis it is verified that the response time for each task is less when they are scheduled using round robin algorithm as compared to a priority algorithm. Thus it can be concluded that round-robin algorithm is more proportionally fair than priority algorithm. The average turnaround time form round robin algorithm is 16.39% more than priority algorithm hence it can be concluded that latency of round-robin algorithm is more than priority algorithm. The throughput of Priority algorithm is more than that of the round robin algorithm. One can schedule computationally efficient tasks with the help of priority algorithm so as to maximize performance and throughput. The scope of the above research work can be extended to highly computational tasks like video encoding, image processing, and signal processing. These applications can be designed using a various scheduling algorithms and their performance can be evaluated.

REFERENCES

[1] Mohammad T. Kawser, Hasib M. A. B. Farid, Abduhu R. Hasin, Adil M. J. Sadik, and Ibrahim K. Razu "Performance Comparison between Round Robin and Proportional Fair Scheduling Methods for LTE" , International Journal of Information and Electronics Engineering, Vol. 2, No. 5, September 2012

[2] Ms. Rukhsar Khan, Mr. Gaurav Kakhani "Analysis of Priority Scheduling Algorithm on the Basis of FCFS & SJF for Similar Priority Jobs" International Journal of Computer Science and Mobile Computing, IJCSMC, Vol. 4, Issue. 9, September 2015, pg.324 – 331

[3] Alban Allkoci, Elona Dhima, Igli Tafa "Comparing Priority and Round Robin Scheduling Algorithms" IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 3, No 1, May 2014

[4] E.O. Oyetunji and A. E. Oluleye, "Performance Assessment of Some CPU Scheduling Algorithms", Res. J. Inform. Technol., 1(1): 22-26, 2009, ISSN: 2041-3114, Aug 2009.

[5] Lalit Kishor ,Dinesh Goyal , "Time Quantum Based Improved Scheduling Algorithm " International Journal of Advanced Research in Computer Science and Software Engineering

[6] Abdulrazaq Abdulrahim ,Saleh E Abdullahi ,Junaidu B. Sahalu " A New Improved Round Robin (NIRR) CPU Scheduling Algorithm International Journal of Computer Applications (0975 – 8887) Volume 90 – No 4, March 2014

[7] Sharmik V Admane, Jitendra B Zalke, Manisha Das," Implementation of Speed Efficient Image Processing algorithm on Multi-Processor System on Chip (MPSoC)", International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2763 Issue 01, Volume 4 (January 2017)

[8] Tong, J.G. Anderson, I.D.L., Khalid, "Soft-Core Processors for Embedded Systems" M.A.S., Microelectronics, 2006. ICM '06. International Conference (IEEE)

[9] Dirk Koch, Frank Hannig, Daniel Ziener (eds.) "FPGAs for Software Programmers" Springer International Publishing (2016)

[10] David E. Simon-"An Embedded Software Primer" Addison Wesley (1999), pearson education D. David.Simon

[11] MicroBlaze Processor Reference Guide Embedded Development Kit.

[12] xilkernel Reference Guide Embedded Development Kit.

Authors Profile

Ms.Manisha Das pursed Bachelor of Engineering in Electronics Engg. (2011) and Master of Technology in VLSI Design (2015) from R.T.M. Nagpur university,India.She is currently working as Assistant Professor in Department of Electronics Engineering, Shri Ramdeobaba College of Engineering and Management,Nagpur,India. Her main research work focuses on Embedded system design, Biomedical Image Processing and Artifitial Intelligence.

Mr. Sharmik Admane is currently working as Assistant Professor in Department of Electronics Engineering of Shri Ramdeobaba College of Engineering and Management, Nagpur. His area of Interest is Low cost Embedded Systems and Embedded operating system. He has complteted his Bachelor of Engineering in Electronics Engineering from R.T.M. Nagpur University, India and Master of Technology in Digital Systems from Pune University, India.

Mr. Jitendra Zalke is currently working as Assistant Professor in Department of Electronics Design Technology of Shri Ramdeobaba College of Engineering and Management, Nagpur. His area of Interest is Embedded Systems and VLSI Design. He has 8years of teaching experience.

