International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org Volume 2 Issue 1 ǁ January. 2013 ǁ PP.23-26

Adaptive system level scheduling under fluid traffic flow conditions in Multiprocessor systems

[1]Kedilaya Shreeganesh.B, [2]Dr.G.M.Patil

[1]Professor, Srinivas School of Engg, Mukka, Mangalore, India.
[2]Principal, Basava Kalyan Engg college, Bidar, India.

Abstract: Fluid traffic flow conditions faced by system level due to its ever increasing complexity in bus architecture means architects can face disruptive scheduling, may need a re- look with respect to traffic pattern analysis at system level and scheduling accordingly in a virtual component framework supporting reactive scheduling. Here, we enumerate the why's and how's of disruptions in scheduling and present a traffic centric scheduling mechanism which will account for shift to traffic centric system level design paradigm [1]. Despite the plethora of studies and commercial solutions proposed, scheduling is still considered as one of the scientific areas where substantial improvements can be gained by the development and application of new research approaches [2]. Transferring this knowledge into practice is really difficult because of poor evaluation of benefits that cannot be got by adopting methods in real time processes and understanding which solution can give better results according to a predefined set of tasks.

Keywords: VLSI, Traffic, Multiprocessor, Robustness.

I. Introduction

In the emerging VLSI technology paradigm, the consumer plays important role. The tasks initiated by the consumer drive the system traffic. Sophisticated scheduling algorithms are used by real time computer systems to process the tasks within dead line. The performance of scheduling algorithm is measured by its ability to generate a feasible schedule for set of real time tasks. Task assignment schemes for homogenous multiprocessor systems are considered where each processor executes a RM scheduling algorithm. This problem has been addressed in a number of studies[2,3,4,5]. Typically the task assignment schemes apply variants of well-known heuristic bin-packing algorithms where the set of processors is regarded as a set of bins. The bin packing problem is concerned with packing different sized items into fixed sized bins using the least number of bins. In [6] two heuristic assignment schemes are proposed refer to as Rate Monotonic Next Fit (RMNF) and Rate-Monotonic First Fit (RMFF). In both schemes tasks are sorted in decreasing order of their periods before the assignment is started. Tasks are assigned to a current processor until schedule ability condition is isolated in which case current processor is marked full and new processor is selected. This is used in offline scheme.

Some work is done to analyze resource allocation schemes for tasks with availability constraints [7]. Qi et.aldeveloped three heuristic algorithms to tackle the problem of scheduling jobs while maintaining machines [8].

In this work, the concepts of predictability, flexibility and stability of a scheduling solution have been tried.

Evaluation of scheduling method consists of

(i) Effectiveness: This indicated the effectiveness of a scheduling method on a multiprocessor system as a result of scheduling policy. How the specified scheduler performs in a steady traffic environment. This is related to evaluating the performance of schedules under the assumptions that no event can disrupt the scheduling.
(ii) Robustness: Ability of the scheduler to maintain its performance in case of disruptions. Robustness is maintained by schedule overlap.
(iii) Flexibility: It is the ability to respond effectively to changing circumstances upon increase in number of processors.

In this study effectiveness of present schedulers its robustness need for disruptive scheduling in homogenous system is analyzed based on emerging and new types of traffics. Scheduling strategy has to ensure that tasks are executed within dead line.

II. Model description and problem formulation

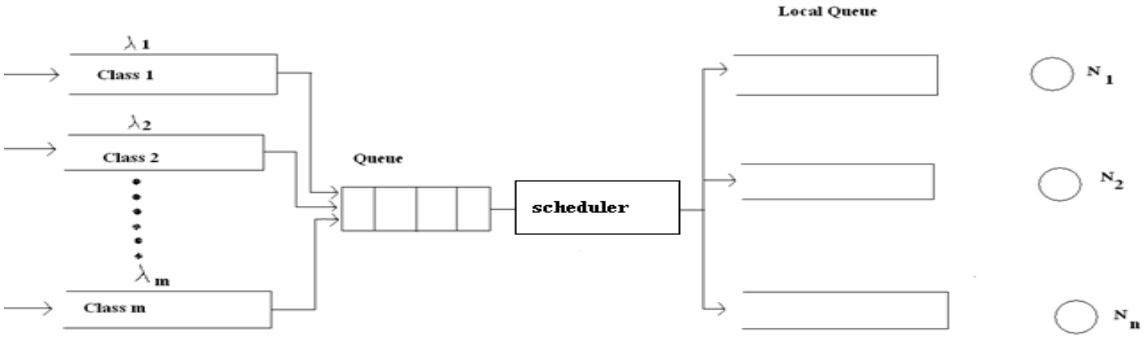


Fig.1.System Model of Multiprocessors.

We consider a queuing architecture of a homogenous system in which 'n' processors are connected to 'm' classes of task through a scheduler 'm' and 'n' are integers greater than or equal to 1. All processors are identical .The system architectural model consists of task scheduler, queue scheduling strategy for multiple classes of tasks and 'n' local task queues. A schedule queue stores incoming tasks and maintains an ideal performance in response time. The scheduler processes the tasks in FCFS manner and dispatchesit to one of the processors. We formulate the scheduling problem as a problem between number of incoming tasks and number of processors required to execute this task when scheduling policy is RMFF and RMNF.

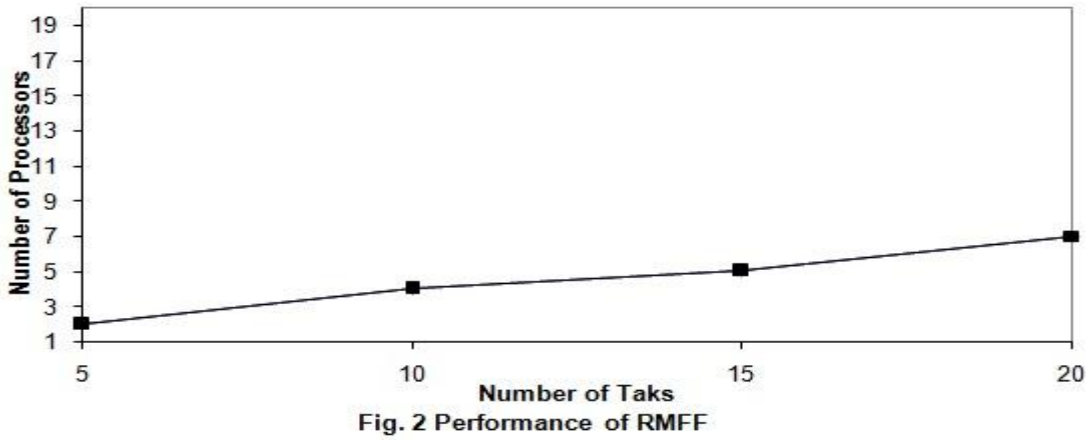


Fig. 2 Performance of RMFF

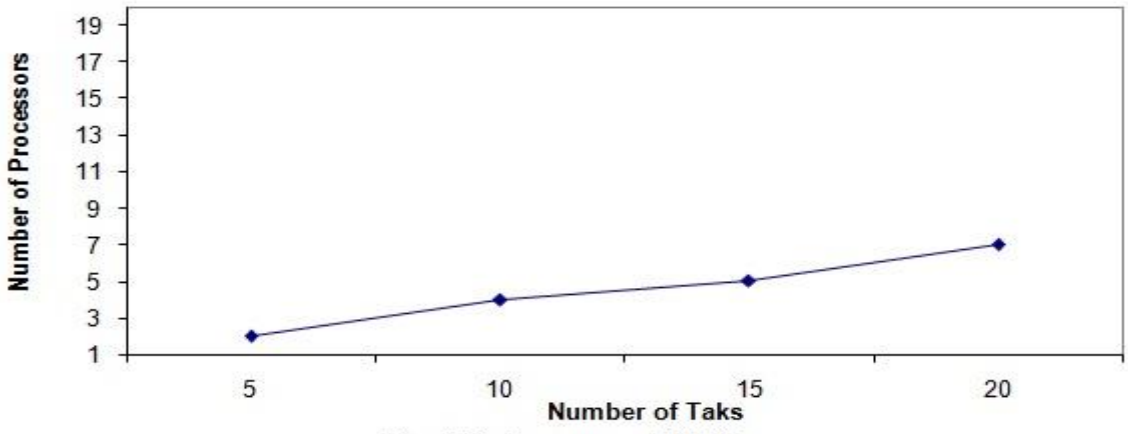


Fig. 3 Performance of RMNF

Number of processors required increases as the number of tasks increases. Since the number of processor is limited, disruptive scheduling is essential.

III. Approach and Contributions

We propose a scheme to tackle the problems of allocation and assignment. Algorithms in class have been shown to rapidly produce high quality solutions for co synthesisproblem [9]. No limitations are imposed on quantities and types of systems resources. However resource use is minimized as a consequence of minimizing energy consumption and price. The algorithm supports reconfigurable FPGAs. FPGA is added to processor block .FPGA based implementations improve energy/time efficiency by an order of magnitude in comparison with processors. Runtime reconfiguration will make it more practical for FPGAs to support numerous tasks. Commercially available reconfigurable devices include vertex from Xilinx. Xilinx uses 1-D configuration model as shown in fig 4.

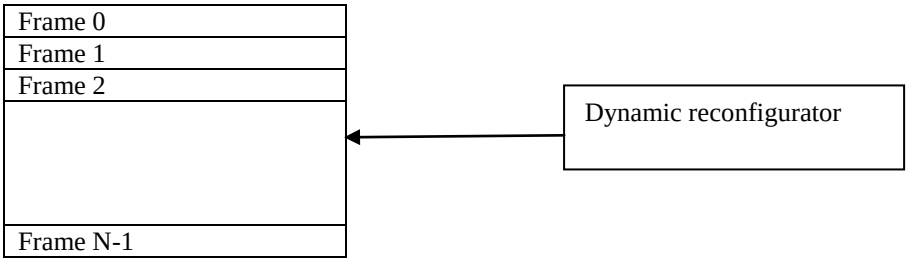


Fig4.Dynamically reconfigurable FPGA model.

FPGA's are reconfigured at framelevel. Only one frame may be refigured at anytime. Each ready tasks need to be loaded on to continuous frames in the FPGA before execution for each frame. The task has a specific configuration pattern.The power consumption of FPGA may be divided into two categories, Execution power and reconfiguration power. For framelevel dynamically reconfigurable FPGA's [10] reconfiguration power is proportional to configuration frequency.

An arrangement to analyze the scheduling approach using FPGA is shown in Fig.5.

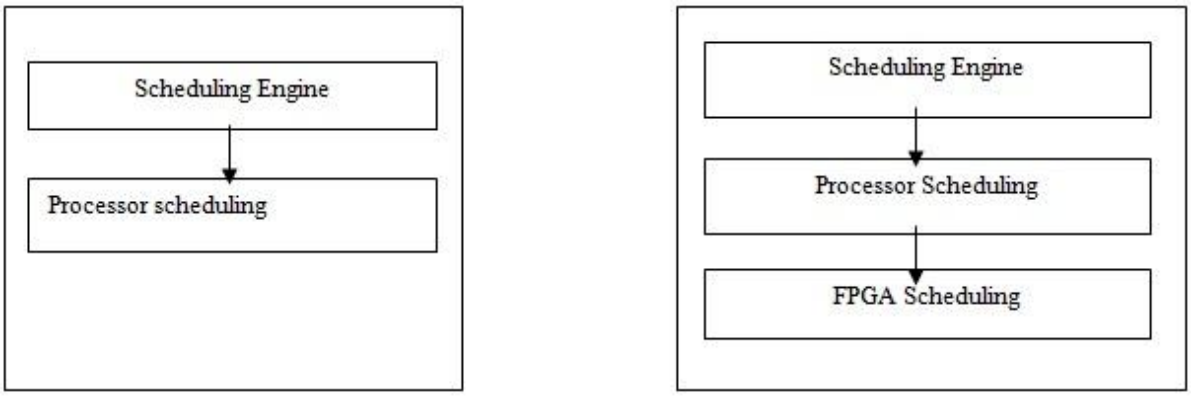


Fig 5a) Multiprocessor scheduling Fig.5b) Processor- FPGA scheduling

IV. Results and analysis

In this section we present simulation results for processor - FPGA scheduling. Here we compare the scheduling results for multi processor scheduling and processor – FPGA scheduling. Simulation results are presented for scheduling with(i) 3 Processor& (ii) Two Processor & one FPGA.The writing reconfiguration time for task i, i.e, r is estimated based on the number of configuration frames (N-frames) used by each task, the frame size(M-bits), the width of the configuration interface (k-bits) which is 8-bits and the configuration frequency f as follows.

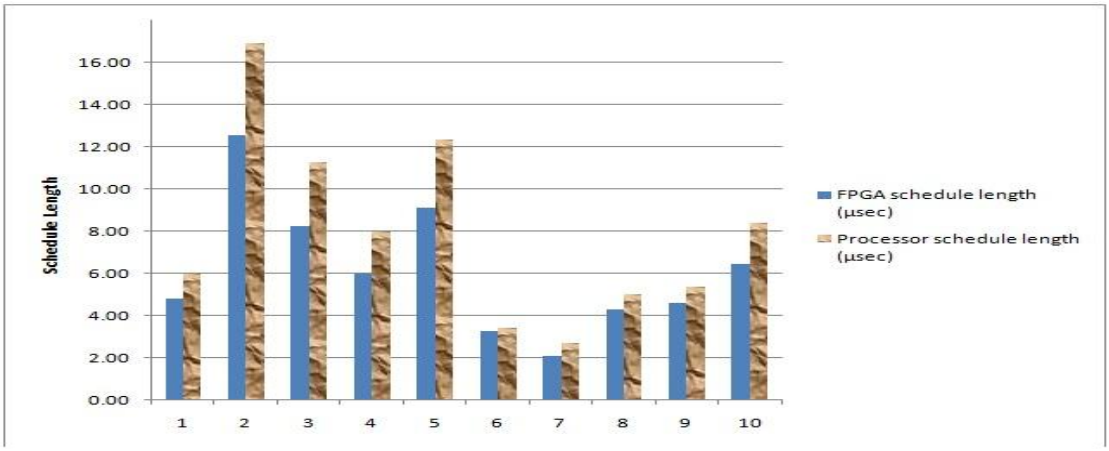
r =N-frames x M-bits/(k-bits x f)+C

Where C is a constant overhead due to initial synchronization and sitting ofthe address & other configuration registers. We use the example & results from [11] to calculate the schedule length & average reconfiguration power. Based on average reconfiguration power, Processor schedule length is estimated. Results are shown in Fig.6.

Table 1

SCHEDULING RESULTS

Example	FPGA Schedule length(µs)	Processor Schedule length (µs)	Average reconfiguration power(mw)
1	4815	5970.6	47.67
2	2530	16915.5	67.11
3	8253	11276.5	72.76
4	5992	7969.3	67.78
5	9139	12337.6	76.03
6	3282	3446.1	14.69
7	2066	2727.1	62.41
8	4270	4995	43.00
9	4600	5382	67.74
10	6444	8377.2	58.42



Example

Fig.6.Scheduling Results

V. Summary and Future work

This scheme reduces the schedule length. An increasing number of applications with traffic constraints are running on homogeneous computing platforms. Existing scheduling systems ignore the disruptive traffic imposed by multiclass applications. Present schedulers focus on robust conditions. A scheduler to work with disruptive conditions is formulated wherein reactive scheduling and reconfigurability are looked in. As part of future directions; a traffic centric scheduling method is to be formulated.

REFERENCES

[1] Kai Nagel " Traffic Networks" ETH Zurich CH-8092 Zurich Switzerland 2002.
[2] J.A.Stankovic and K Ramamritham (editors) Hard real Time Systems IEEE Computer Society press, 1988.
[3] J.X.T Leung and J. Whitehead. On the complexity of Fixed priority scheduling of Periodic Real Time tasks Performance evaluation 2:237-250,1982
[4] S. Davari and S. K Dhall.An online algorithm for Real Time Allocation. In 19 th Annual Hawaii international conference on system sciences pages 194-200;1986
[5] S. Davari and S. K Dhall.An online algorithm for Real Time Allocation. In IEEE Real Time Systems symposium pages 194-200; 1986
[6] S K Dhull and C L Liu on a real time scheduling problem Operation Research 26(1):127-140 Jan/Feb 1978
[7] E Sanlaville and g Schmidel "Machine Scheduling Problem with Availability Constrants;" Proc Ninth Int'/workshop project management and scheduling (PMS 04) 2004.
[8] A Dogon and F Ozgiiner" Kenable Matching and Scheduling of Precedence Constrained Tasks in Hiterogenous Distributed Computing" Proc 29thInt'\conf Parallel Processing(ICPP'00) PP 307-314,2000.
[9] J Goodman and A P Chandrkasan"An energy efficient reconfigurable public-key-cryptography processor architecture" in Proc Int. workshop cryptographic hardware and Embedded system Aug 2000 pp 175-190
[10] S Choi R Scrofano, U K Prasanna and J W Jang " Energy efficient signal processing using FPGA's" in Proc Int. Symp Field Programmable Gate Arrays Feb 2003 pp 225-234.
[11] L.Shang,R.P.Dick.andN.K.Jha "Hardware-Software Co synthesis of Low power Real time Distributed embedded Systems With Dynamically Reconfigurable FPGAs"IEEETrans.on Computer aided Design of ICs and systems.vol26 ,no 3