iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 Issue: IV Month of publication: April 2019

DOI: https://doi.org/10.22214/ijraset.2019.4456

www.ijraset.com

Call: ☎08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

Energy Saving in Real-Time Systems with Scheduling Algorithm

Vaishali B Bambode
Department of Computer Engineering, SIPNA COET Amravati University

Abstract: A system is said to be a real-time system if it produces logically correct results within a specified time called deadline .Real-time system have to perform a multitude of complex task with strict time constraints. Thus, minimizing power consumption has become a critical aspect in designing such systems. In this work we discuss the scheduling algorithms for real-time systems and how they contribute to minimize the energy consumption. Scheduling in real time system determines the order in which real time task should be executed such that they meet the deadlines. Now a day's many scheduling algorithms have been developed to reduce energy consumption. For a real time application, comprising a set of real-time task with precedence, resource and time constraint executing on a distributed system, we think of energy efficient scheduling algorithm like weighted first come first serve (WFCFS) scheme.

Keywords: Real time system, energy reduction, scheduling, deadline, algorithm.

I. INTRODUCTION

A real time system [1]is one whose correctness depends on timing as well as functionality. When we discuss more traditional scheduling algorithms, we look at were turnaround time (throughput),fairness, mean response time. And we introduce a few new concepts like feasibility, hard and soft real time systems. It looks like real time scheduling is more critical and difficult than traditional time sharing schemes. In a simplest real time system, where task and their execution times are well known, there might not even be a scheduler. One task might simply call the next. This model makes a great deal of sense in a system where the tasks from a producer/consumer pipeline .In more complex real time system, with a larger number of tasks that do not function in a strictly pipeline fashion, it's a static scheduling. Based on the list of tasks to be run and the expected completion time for each, we can define a fixed schedule that will ensure timely execution of all tasks. For many real time systems, the work-load changes from moment to moment, based on external events. These requires dynamic scheduling. There are different scheduling algorithms to decide which process to execute first and which process to execute last to achieve maximum CPU utilization.

II. LITERATURE REVIEW

Operating system is responsible for functioning of the entire system, including task constraints and status, resource usage etc. Therefore it is one of the most effective and efficient approaches to reduce energy consumption with proper task scheduling algorithm. Most of the work on energy efficient real time scheduling concentrate on independent tasks over uniprocessor. The two most commonly used techniques that can be used for energy minimization are Dynamic Voltage Scaling (DVS)[4] and Dynamic Power Management(DPM). The application of these system-level energy management techniques can be exploited to the maximum if we can take advantage of almost all of the idle time in between processor busy time. The major challenge is to design an efficient scheduling algorithm which can use slack time to the maximum. Various energy efficient slack management schemes have been proposed for these real time distributed systems. The static scheduling algorithm uses critical path analysis and distributes the slack during the initial schedule. The dynamic scheduling algorithm provides best effort service to soft aperiodic task and reduces power consumption by varying voltage and frequencies. One of the existing work for distributed real time system is based on feedback control scheduling (FCS). This algorithm can provide real time performance efficiently, even in open environment. Within each processor the task are scheduled by the basic scheduler using Earliest Deadline First Algorithm (EDF), since EDF has been proven to be an optimal uniprocessor scheduling algorithm. It allows dynamic rescheduling and pre-emption in the queue and processing mode. Another work includes Service Rate Proportionate (SRP) slack distribution technique for energy efficiency. Both the dynamic and the rate-based scheduling schemes have been examined with this technique.

It introduces SRP a dynamic slack management technique to reduce power consumption. The SRP technique improves performance /overhead by 29 percent compared to contemporary techniques. Deadline hit ration is a major factor to be considered in a soft real time systems for better quality of service. Following algorithms increases the deadline hit ratio and thereby improving the quality of the system.

A. Weighted FCFS Scheduling Algorithm

In the First Come First Serve scheduling algorithm, as the name suggest ,the process which arrives first ,or we can say the process which requests the CPU first ,gets the CPU allocate first. It's just like FIFO queue data structure, where the data element which is added to the queue first, is the one who leaves the queue first. This is used in Batched systems. It's easy to understand and implement programmatically, using a queue data structure, where a new process enters through the tail of the queue and the scheduler selects process from the head of the queue. The perfect real life example of FCFS scheduling is buying tickets at ticket counter.

B. Dynamic Slack Management Algorithm

The slack distribution technique considers a modified Feedback Control Algorithm. This algorithm schedules dependant task effectively with precedence and resource constraints. It further minimizes the schedule length and utilizes the available slack to increase the energy efficiency. A fault tolerant mechanism uses a deferred –active-backup scheme increases the timing and provides reliability to the system.

C. Priority Scheduling Algorithm

Priority based scheduling enables us to give better service to certain processes. In our discussion of multi-queue scheduling, priority was adjusted based on whether a task was more interactive or compute intensive. It is inherently a best effort approach. If our task is competing with other high priority tasks, it may not get as much time as it requires.

D. Round Robin Scheduling Algorithm

A fixed time is allocated to each process, called quantum for execution. Once a process is executed for given time period the process is pre-empted and other process executes for given time period. Context switching is used to save states of pre-empted processes.

E. Multi-Level Queue Scheduling Algorithm

It's a scheduling algorithm that has been created for situations in which processes are easily classified into different groups. For example, Separate queues might be used for foreground and background processes. The foreground queue might be scheduled by round robin algorithm, while the background queue is scheduled by an FCFS algorithm.

F. Shortest Job First Scheduling Algorithm

Shortest Job First Scheduling works on the process with the shortest burst time or duration first. This is the best approach to minimize waiting time. This is used in batched system. It is of two types a> Non Pre-emptive b> Pre-emptive

To successfully implement it, the burst time of the processor should be known to the processor in advance, which is practically not feasible all the time. This scheduling is optimal if all the jobs/processes are available at the same time.

III.PRACTICAL REVIEW

In real time systems, the system requires the complete details of the task processing time i.e. the execution time and deadline before program execution. Each real time system application has an end –to-end deadline, by which it has to complete its execution and produce result. To understand the energy consumption and develop a system which reduces the energy consumption to its maximum many efforts has been taken. The time taken by the system to respond to an input and display of required updated information is termed as response time. In RTOS response time is very less as compared to the online processing. Real time processing is always online whereas online system need not be real time. While implementing any RTOS using scheduling algorithm factors like precedence, resource and time are considered. Task considered in real time application have precedence constraint; where one task can become eligible for the execution only after a other task has completed its execution. Some real time application have resource constraints. Resources are reusable and can be shared, but have mutual exclusion constraints. Thus only one task can be using a resource at any given time. This applies to physical resources, such as disks and network segments, as well as logical resources such as critical code section that are guarded by semaphores. Only a single instance of a resource is present in the system. A task can only request a single instance of a resource. If multiple resources are needed for a task to make progress, it must acquire all the resources through a set of consecutive resource request. In general, we assume that a task can explicitly release resources before the end of its execution .Thus, it is necessary for a task that is requesting a resource to specify the time to hold the requested resource. It's called Hold Time. The scheduler in real time system uses the Hold Time information at run time to make scheduling decisions. In real time, slack management is increasingly applied to reduce power consumption. And optimize the system with respect to performance and time overhead. This slack management technique exploits the idle time of the system through DVS in order to achieve the highest possible energy consumption. In energy efficient scheduling, the set of task will have certain deadline before which they should finish their execution and hence there is always a time gap between the actual execution time and deadline; It's called slack time. The main challenge of any real time system is to obtain and distribute the available slack in order to achieve the highest possible energy saving with minimum overhead.

An energy efficient real time system is implemented practically for the applications with strict deadlines. Real time systems are mainly used in applications with zero execution time, where a task has to be performed at a specific instant. For example, in a digital control theory taking measurement, calculating a control action and sending it to the peripheral devices occurs instantly. The RTOS can be used in applications where a task must have a fixed service amount per time unit. Applications that involves competitions for resources like network, processors and memory. Real time systems are mainly use in ATM'S, Embedded systems, Air traffic control systems, Image Processing, Cloud computing, Robotic equipment, Chemical plants etc.

IV.FUTURE WORK & CONCLUSION

To gain energy saving in real time systems using scheduling algorithms a lot of work has been done; also the precedence, resource and time constraints. In future, techniques need to be developed for better Quality of system. which increases the deadline hit ratio & improves the quality of the system. Looking at the complexity of modern applications there is a need of providing appropriate level of protection.

In This paper work, an energy efficient real time scheduling algorithms and techniques are reviewed. These algorithms are capable of handling task with precedence and resource and time constraints. Main goal of this review paper is to study about real time operating system with scheduling algorithm which is energy efficient, What all are the scheduling algorithms used in real time systems, How these scheduling algorithms helps to reduce the energy consumption in real time operating systems, practically how to implement the models to get result , real time applications of energy efficient real time system .

V. ACKNOWLEDGMENT

I wish to acknowledge Dr G S THAKARE [Associate Professor] Department Of Computer & Science engineering ,SIPNA COET for his contribution in my work of documentation and development of this review paper.

REFERANCES

[1] Rabi N. Mahapatra and Wei Zhao, (2005) "An Energy-Efficient Slack Distribution Technique for Multimode Distributed Real-Time Embedded Systems", IEEE Transactions on Parallel and Distributed Systems, vol. 16, no. 7.

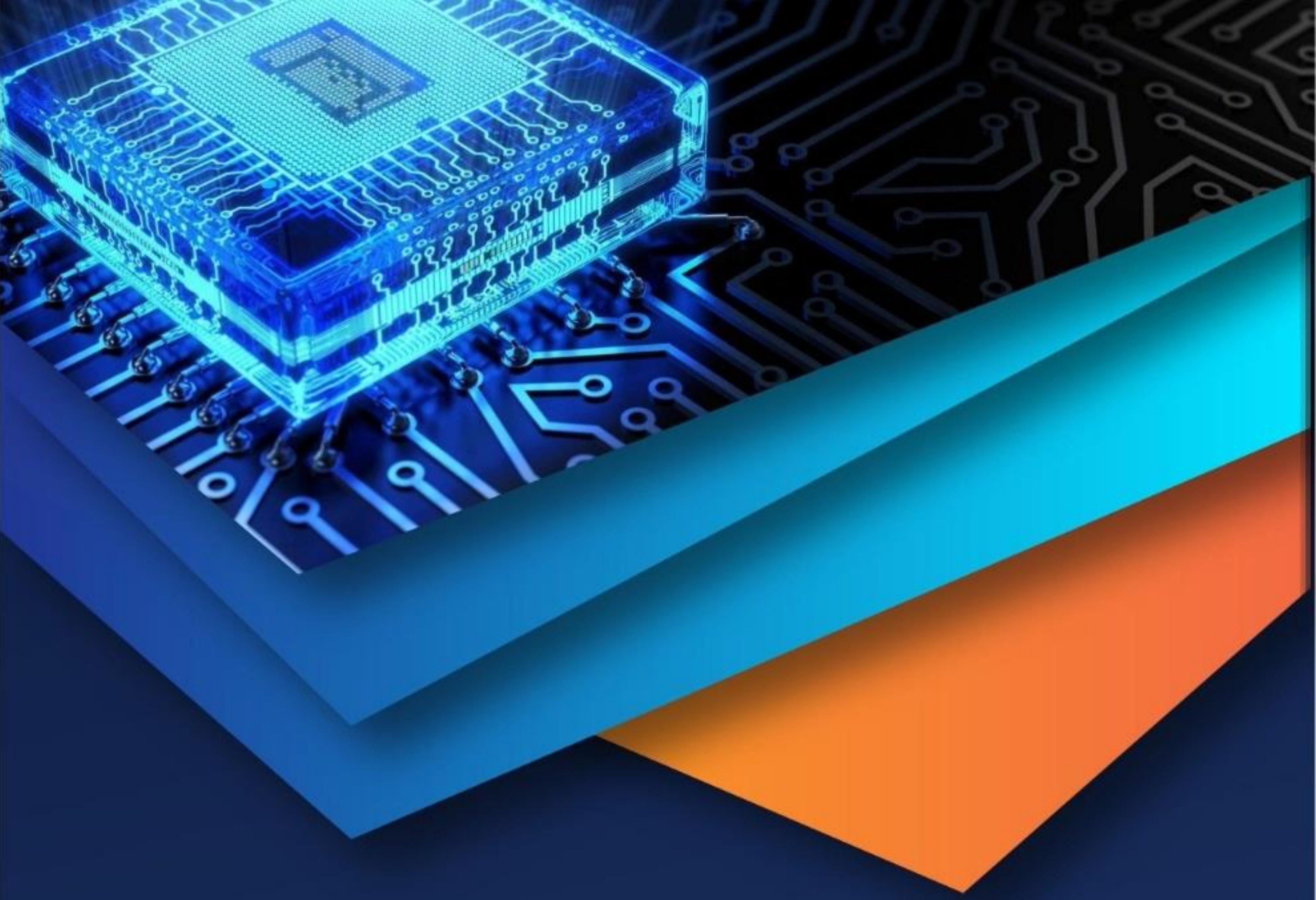
[2] Santhi Baskaran & P.Thambidurai "Energy efficient real-time scheduling in distributed systems"IJCSI International journal of computer science issues ,vol.7,Issue 3 ,No.4 May 2010.

[3] T. Ishihara and H. Yasuura, (1998) Voltage Scheduling Problem for Dynamically Variable Voltage Processors, In: International Symposium on Low Power Electronics and Design, pp. 197-202.

[4] L. Benini, A. Bogliolo, and G. De Micheli, (2000) A Survey of Design Techniques for System-Level Dynamic Power Management, In: IEEE Transactions on VLSI Systems, pp. 299-316.

[5] C.M. Krishna and Shin K. G., Real-Time Systems, (1997) Tata McGraw-Hill.

[6] Subrata Acharya and Rabi N. Mahapatra, (2008) A Dynamic Slack Management Technique for Real-Time Distributed Embedded Systems, In: IEEE Transactions on Computers, Vol. 57.



doi crossref

10.22214/IJRASET

INDEX COPERNICUS

45.98

ISRA JIF

IMPACT FACTOR: 7.129

SJIFactor

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

ISI

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)