Model And Simulation Of A System With Power Optimization

Madhuri Jadhav[1], Sakha Gabhane[2]

[1]Student, EXTC Department , Rajiv Gandhi Institute of Technology, University of Mumbai, India
[2]Associate Professor, EXTC Department , Rajiv Gandhi Institute of Technology, University of Mumbai, India

---

Abstract - ***Modern developments in embedded systems furnishes the application of distributed control systems over a wireless network. However, some problems emerge during the design and operation of such systems, such as multi-process scheduling of the Electronic Control Unit (ECU), memory management and also most importantly power management. Software tools help to get through the effective analysis and solution of such problems. The concept of Model-Based Design and related software tools fits these requirements and becomes commonly accepted by the communities. Hence a model is generated which includes a controller and an optimizer. The optimizer includes an application of on-line gradient estimation technique (i.e. Infinitesimal Perturbation Analysis or IPA) which is used for the power control of ECUs. A Dynamic Voltage Scaling (DVS) controller is created and accessed using SimEvents, a discrete event simulator in the MATLAB. By monitoring the current workload of the system and performing on-line gradient estimations, this DVS controller dynamically updates the input voltage of the ECU of the device, so that the power consumed by the ECU is optimized and at the same time the quality of service measured by the average system time of a job is guaranteed. Also the performance does not deteriorate at the worst temperature conditions.***

Key Words: SimEvents, ECU, DVS, IPA, power

1.INTRODUCTION

When designing a DES, the optimal parameter configuration gets more complicated. As DES is usually subjected to stochastic inputs and random noise, significantly greater struggle (in terms of simulation time and computation complexity) is needed to obtain a precise computation of the system. The widely applicable ''finite difference" method is time-consuming as well as less accurate. Thus Infinitesimal Perturbation Analysis (IPA) technique is chosen which is a great tool to resolve the optimal parameter configuration issue. By using IPA, one can obtain sensitivity information of the parameterized system from a single simulation. With such sensitivity information, we gain the current performance of the system as well as the first order approximation of the performance measures as a function of the configurable parameters.

1.1 Dynamic Voltage Scaling

Common methods apply on-off control to control the duty-cycle of the ECU. According to this method, when the workload of the ECU is low or inactive, the processor's circuit will be entirely shut down. When a new job enters, the computer is alerted and the job is processed at desired speed. Also, the DVS techniques manage power consumption by altering the frequency (i.e. speed of processing) of the processor.

The main intention of DVS power control is that when the workload of the ECU is low, the DVS controller lowers the input voltage of the processor so that it works at a moderately slow pace, thus less energy is required. Whereas when the workload is high, DVS controls the processor working at an upper frequency, so that the QoS can be ensured. However, due to the unpredicted nature of job coming processes and the uncertainty of the workload of incoming jobs, the DVS controller needs to be carefully designed so that it can lower power usage without hindering the overall performance of the technique.

1.2 Queueing Model And Dvs Controller

The Control Unit is modeled as a single server queuing system. Incoming jobs are stored in a buffer that can be modeled as a FIFO queue with immense capacity. The arrival process is random and unpredictable. The inter-arrival times are random variables that are independent and identically distributed (i.i.d.). Also it is assumed that no information about the arrival process, arrival rate and the distribution is known. Also, the size of each independent job is random. Here we have assumed the workload as exponentially distributed with the average load of 1 million operations.

It is known that the processing speed is a function of the input voltage. This function is given by,

$$[{}^{dS}\!/_{d\theta_k}]_{IPA}$$

$$V = \frac{V_t}{1-c_1 f} \quad \text{or} \quad f = \frac{1}{c_1}(1-\frac{V_t}{V}) \tag{1}$$

Here, f is the processing speed (in MHz), V is the input voltage (in Volts). V_t, is the threshold voltage of the device and c_1 is a device constant.

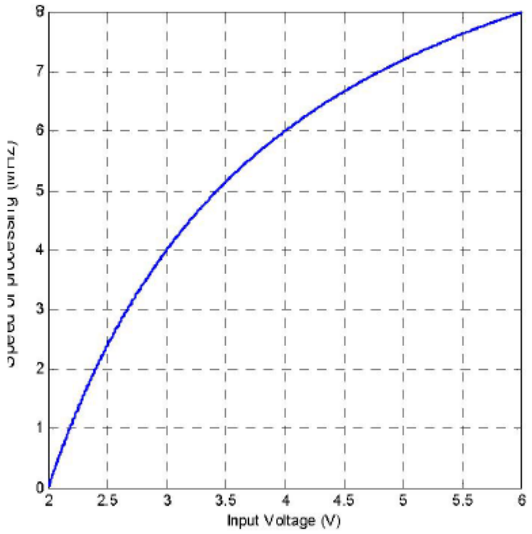


Figure 1: Processing speed of an ECU at different voltage levels

Fig. 1 illustrates the functional relationship between V and f

The energy utilized by the processor to process a job can be given by,

$$P = c_2 N V^2 \quad (2)$$

where c_2 =0.4167 x 10^{-3} is a device dependent constant, N is the number of operations required to process the job, V is the input voltage and P is the energy consumed (in Joules). For a job with IM operations, Fig. 2 presents its energy consumption at various voltages.

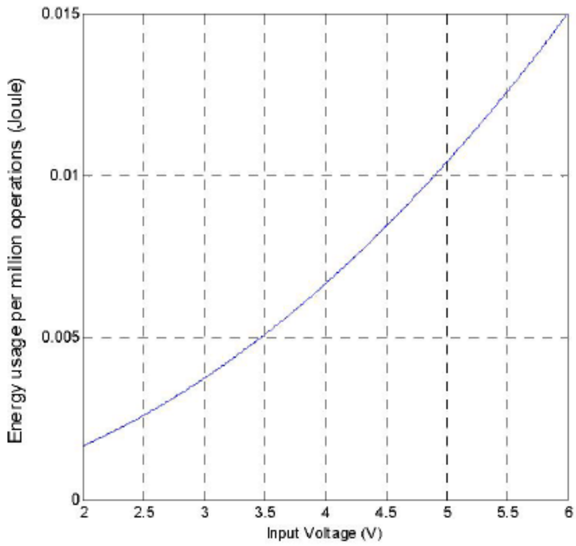


Figure. 2 Energy usage of an ECU at different voltage level

The processing speed model along with the energy model (i.e., eq. (1) and (2)) allows DVS power control. For some job with 10^6 operations, when input voltage varies from 4V to 6V, the energy usage changes from 6.67mJ to 15mJ.

2. LITERATURE SURVEY:

In [1] Xian et al. present an algorithm for scheduling in multiprocessor systems to save energy. For scheduling periodic real-time tasks, their method uses EDF scheduling to make sure the meeting of deadlines of all tasks and hence minimizing energy consumption. A polynomial time heuristic method is shown as the problem is NP-hard. This problem is solved assuming that unbounded and continuous range of frequencies are available. Later, the work is changed so that maximum frequency and the bounded discrete frequencies are available.

Niu et al. in [2] propose an approach to economize leakage and dynamic energy in embedded systems by affiliating DVFS and PPM. In the case when processor is active, the algorithm chooses a processor speed such that the dynamic and leakage power consumption are balanced.

In [3] an extended list-scheduling algorithm is applied to reschedule the scheduled tasks. At each step, the scheduler determines energy saving of a task when it is scheduled at the present step and the next step. The energy difference of these two steps are represented as the task's energy saving. Then a task with a higher energy saving and lower slack time gets higher priority to be scheduled.

In [4] J. Luo and N. K. Jha have presented an intra-task voltage scheduling algorithm based on a static timing analysis. In this technique, a given task is divided into several subtasks, and then suitable supply voltage (resulting from static timing analysis of previous segments and based on worse-case execution time of the task) is given to each segment. This approach has a high energy reduction ratio by consuming the idle time, and selecting suitable voltage/frequency to occupy these idle times to the fullest.

3. SIMEVENTS MODEL OF THE DVS CONTROLLER:

Figure 3. and Figure 4. is a SimEvents/Simulink model of DVS for the single server queuing system.

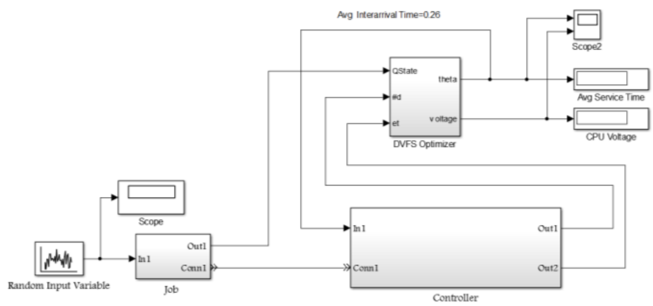


Figure 3. Model based design of DVS system using SimEvents and MATLAB

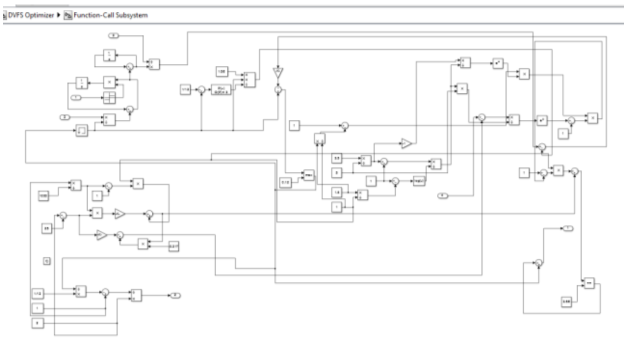


Figure 4. Model-based design of DVS optimizer using SimEvents and MATLAB/Simulink

The SimEvents is a "queues and servers" style discrete event simulation which allows entities to be passed from block to block to constitute the movement of jobs through the microprocessor. This model generates entities using an intergeneration. The created entities are stored in the FIFO Queue block before being delayed by the Single Server block and then being sent to the Entity Sink. The time for the entity to pass through the FIFO Queue block and Single Server block is marked by the Start Timer and End Timer blocks. This time is used in conjunction with the counts of entities passing through other blocks by the DVS optimizer to calculate performance measurements below (see eqs. (3) through (9)). The perturbation is given by the Random Service Time subsystem block that varies the service time used by the Single Server block.

2.1 DVS Controller based on IPA

The DVS controller monitors the current workload and dynamically adjusts the input voltage. In order to provide energy savings and ensure QoS simultaneously, an optimizer is generated to calculate the optimal input voltage. This performance metric is given by

$$J(V) = wP(V) + S(V) \qquad (3)$$

$J(V)$ Comes from two parts: (i) $P(V)$ -the average energy consumption of a job, and (ii) $S(V)$ -the average service time of a job. In (3), w is a weighting constraint which can be thought of as the reason of energy relative to delay, By using (3), energy can be optimized without impairing the QoS offered by the processor. By combining (1), (2) and (3), the performance metric can be reconstructed as,

$$J(\theta) = wc_2\left(\frac{V_t}{1-c_1/\theta}\right)^2 + S(\theta) \qquad (4)$$

Now in (4), V is replaced by θ --the average service time of a job ($\theta = 1/f$). According to the essential condition for optimality, when minimizing (4), the optimal solution must satisfy $f^*(MHz)$ (5)

During imitation, $\frac{ds}{d\theta}$ can be calculated and $\frac{dJ}{d\theta}$ can be generated, a gradient method then can be applied to find a θ that satisfies (5). In the DVS controller, we adopt a gradient method with constant step size. That is, in the k^{th} iteration of the optimization, starting from θ_k, a new average service time is generated by,

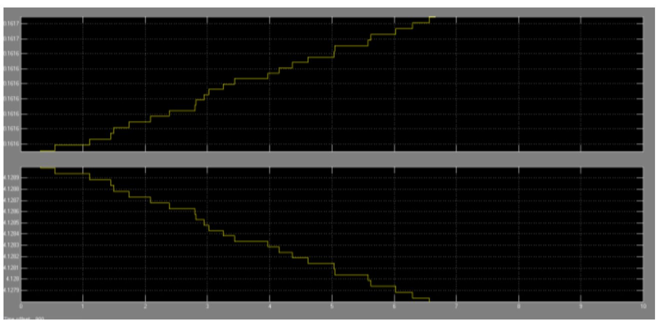
$$\theta_{k+1} = \theta_k - \Delta\left[\frac{dJ}{d\theta_k}\right] \qquad (6)$$

Where Δ is a constant step size. (6) is the well-known "steepest decent method" and by applying it, it is guaranteed that $J(\theta_{k+1}) \le J(\theta_k)$. In (6), the $dJ/d\theta_k$ is projected using,

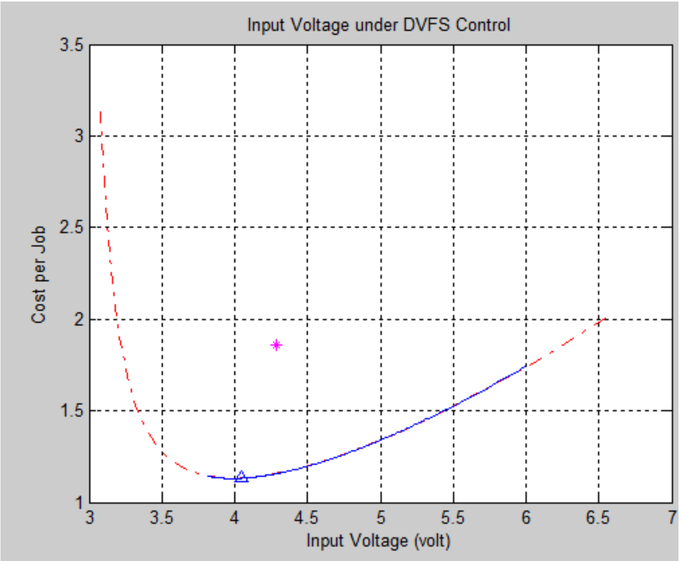
$$\left[\frac{dJ}{d\theta_k}\right] = -\frac{2wc_1c_2V_1^2\theta_k}{(\theta_k - c_1)^3} + \left[\frac{ds}{d\theta_k}\right]_{IPA} \qquad (7)$$

with $[dS/d\theta_k]_{IPA}$ being the gradient estimate generated by an IPA estimator [1]. By combining (6) and (7), the iterative process can be summarized to find an optimal θ (and thus V) to reduce the average cost of a job.

3. RESULTS:



This result shows that the proposed model ensures QoS as well as the minimization of the energy consumption of the processor. Also when the voltage increases from 4V to 6V, the average cost per job with the power gets optimized at one point. This following result shows the comparison of the power with the application of DVS and without the application of DVS and it can be clearly seen that the performance metrics and the goals are achieved as the power consumption is scaled down after applying DVS algorithm.



REFERENCES

[1] Xian, Changjiu, Yung-Hsiang Lu, and Zhiyuan Li. "Energy-aware scheduling for real-time multiprocessor systems with uncertain task execution time."*Proceedings of the 44th annual Design Automation Conference*. ACM, 2007.

[2] Niu, Linwei, and Gang Quan. "Reducing both dynamic and leakage energy consumption for hard real-time systems." *Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems*. ACM, 2004..

[3] Gruian, Flavius, and Krzysztof Kuchcinski. "LEneS: task scheduling for low-energy systems using variable supply voltage processors." *Proceedings of the 2001 Asia and South Pacific Design Automation Conference*. ACM, 2001.

[4] R. Springer, D. K. Lowenthal, B. Rountree, and V. W. Freeh, "Minimizing execution time in MPI programs on an energy-constrained, power-scalable cluster," presented at the Proceedings of the eleventh ACM SIGPLAN symposium on Principles and practice of parallel programming, New York, New York, USA, 2006

[5] Balasubramanyam, Vidya, et al. "Modeling and Simulation of an Energy Harvesting System."