Temperature-Aware Voltage Selection for Energy Optimization

M. Bao, A. Andrei, P. Eles, Z. Peng
Embedded Systems Laboratory (ESLAB)
Department of Computer and Information Science, Linköping University, Sweden
{g-minba, alean, petel, zebpe}@ida.liu.se

Abstract

This paper proposes a temperature-aware dynamic voltage selection technique for energy minimization and presents a thorough analysis of the parameters that influence the potential gains that can be expected from such a technique, compared to a voltage selection approach that ignores temperature.

1. Introduction

One of the preferred approaches for reducing the overall energy consumption of embedded system is dynamic voltage selection (DVS). This technique exploits the available slack times by reducing the voltage and frequency at which the processors operate and, thus, achieves energy efficiency.

The high power densities achieved in current SoCs do not only result in huge energy consumption but also lead to increased chip temperatures. Several approaches to thermal aware system-level design have been proposed in recent years. Of particular importance in this context is the development of adequate temperature modeling and analysis tools. The approach proposed in HotSpot [4] performs both static analysis (producing steady state temperature) and dynamic analysis (producing temperature profiles). A similar approach is proposed in [13], where dynamic adaptation of the resolution is performed, in order to speed up the analysis.

Thermal aware task allocation and scheduling have been addressed in [12]. In [11] an approach to task scheduling under peak temperature constraints is presented. Design space exploration for multiprocessor SoC architectures under area and thermal constraints is presented in [6], while in [10] thermal aware floorplanning is advocated. However, the temperature issue has been completely ignored in the proposed DVS techniques for real-time embedded systems. One exception is [7] which takes into consideration the effect of temperature on leakage at voltage scaling, in the context of a design process aimed at reducing peak temperature.

In this paper we propose a technique for temperature aware energy minimization by DVS, considering both supply voltage selection and ABB. We consider both static and dynamic temperature analysis in our optimization process. Furthermore, we perform, for the first time, a thorough analysis of the parameters that influence the potential gains that can be expected from a thermal aware DVS technique, compared to an approach that ignores temperature.

2. Preliminaries

2.1. System and Application Model

We consider architecture is realized as multiprocessor systems on chip. We assume that the processors can operate in several discrete execution modes. An execution mode is characterized by a pair of supply and body bias voltages: (V_{dd}, V_{bs}).

The functionality of the application is captured as a set of task graphs. Nodes represent computational tasks, while edges indicate data dependencies between tasks (communication). Tasks are annotated with deadlines that have to be met at runtime. We assume that the task graphs are mapped and scheduled on the target architecture, i.e., it is known where and in which order tasks and communications take place. For each task the worst case number of cycles to be executed is given.

2.2. Power Model and Voltage Selection

For dynamic power we use the following equation [3], [8]:

$$\mathrm{P}_{dyn} = C_{eff} * f * V_{dd}^{2} \qquad (1)$$

where C_{eff}, V_{dd}, and f denote the effective switched capacitance, supply voltage, and frequency, respectively.

The leakage power is expressed as follows [5], [7], [8]:

$$P_{leak} = I_{sr} * T^{2} * e^{(\frac{a*V_{dd}+\beta*V_{bs}+\gamma}{T})} * V_{dd} + |V_{bs}| * I_{Ju} \qquad (2)$$

where I_{sr} is the reference leakage current at reference temperature. T is the current temperature, V_{bs} is the body bias voltage, and I_{ju} is the junction leakage current. a, β and γ are curve fitting circuit technology dependent coefficients.

Circuit delay and operational frequency are depending on the supply and body bias voltage [8]:

$$f = \frac{1}{d} = \frac{((1+K_{1})*V_{dd}+K_{2}*V_{bs}-V_{th1})^{\alpha}}{K_{6}*Ld*V_{dd}} \qquad (3)$$

where Ld is the logic depth. K_1, K_2, K_6, and V_{th1} are technology dependent coefficients. α reflects the velocity saturation imposed by the used technology (common values $1.4 < \alpha < 2$).

In [1] we have presented an approach to combined supply voltage selection and adaptive body biasing. Given a multiprocessor architecture and a mapped and scheduled application, as presented in Section 2.1, the DVS algorithm calculates the appropriate execution modes (V_{dd} and V_{bs}) for each task, such that the total energy consumption is minimized. Another input to the algorithm is the dynamic power profile of the application, which is captured by the average switched capacitance of each task. This information will be used for calculating the dynamic energy consumed by the task in a certain execution mode, according to equ. (1). Leakage energy, during the optimization process, is calculated based on equ. (2). However, since leakage strongly depends on temperature, an obvious question is which temperature to use for leakage calculation. Ideally, it should be the temperature at which the chip will work when executing the application. This temperature, however, is not

978-3-9810801-3-1/DATE08 © 2008 EDAA

known. The algorithm in [1] requires the designer to introduce an assumed temperature which is used at energy optimization. This leads to suboptimal results, since the temperature used for energy calculation is different from the actual temperature at which the chip works. Therefore, using the calculated voltages, the chip will dissipate more energy than with voltages that would be obtained knowing the real temperature at which the chip is going to function.

In the following, based on our approach in [1], we will develop a temperature aware voltage selection technique.

3. Temperature Analysis

Temperature analysis in our proposed DVS technique is based on HotSpot [4]. When provided with the physical/thermal parameters (size and placement of cores, thermal capacitances and resistances, parameters of packaging elements) and the power profile capturing the power dissipation of each core, HotSpot produces the steady state temperature or the temperature profile of the cores. However, the temperature analysis does not support the case in which power dissipation is dependent on the temperature, which is the situation with leakage.

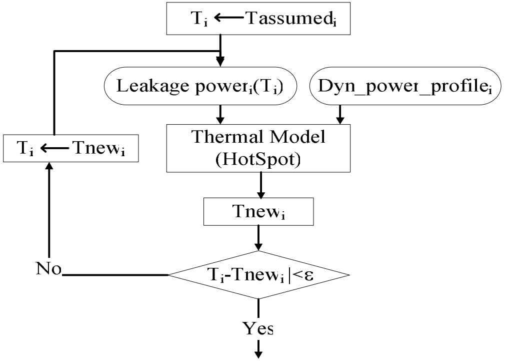


Fig. 1 Static thermal analysis with leakage (the index i indicates that the particular item is introduced/produced for each core)

The solution used by us for static temperature analysis is outlined in Fig. 1. The process is started with an "assumed" temperature and then continued iteratively until the produced temperature converges. At this steady state temperature the dissipated heat is in balance with the heat removal capacity of the package. However, it can happen that such a balance is not achieved, due to insufficient heat removal, and the temperature is increasing, potentially, to infinite. In such a case, the iterations in Fig. 1will not converge. This phenomenon, called *thermal runaway*, is detected and indicates that the design is incorrect from the thermal point of view.

In reality, however, in the context of a variable power profile, the chip will not reach a constant steady state temperature but a steady state in which temperature is varying according to a certain pattern. In order to obtain the steady state temperature profile, we need to use dynamic thermal analysis. For dynamic analysis, HotSpot is calculating temperatures at successive time steps [4]. At each step a new temperature is calculated for each block by solving the equations describing the thermal model, based on a fourth-order Runge-Kutta method. The power consumption during the time interval between two steps is extracted from the power profile for the respective block. However, leakage power is a function of the temperature and, thus, cannot be delivered as an input to the analysis.

In order to solve the above problem we have extended the thermal analysis such that the power consumption during a time step is calculated as the sum of two components: (1) the dynamic power extracted from the input power profile and (2) the leakage power calculated at the temperature level of the previous step. The process is illustrated in Fig. 2. Temperature analysis is repeated for successive periods of the application. In order to detect convergence, temperature values at corresponding time steps of these successive periods are compared.

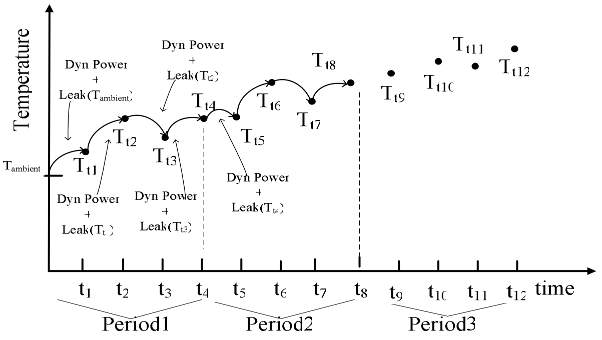
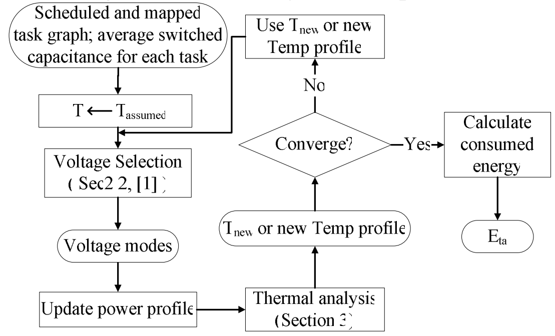


Fig. 2 Dynamic thermal analysis with leakage

For both static and dynamic analysis, convergence is reached efficiently except if thermal runaway occurs. Since dynamic thermal analysis is more time consuming than static analysis, obtaining a steady state temperature profile is slower than calculating a constant steady state temperature.



Scheduled and mapped task graph; average switched capacitance for each task
$T \leftarrow T_{assumed}$
Voltage Selection (Sec2.2, [1])
Voltage modes
Update power profile
Thermal analysis (Section 3)
T_{new} or new Temp profile
Converge?
No
Use T_{new} or new Temp profile
Yes
Calculate consumed energy
E_{ta}

Fig. 3 Temperature-aware voltage selection

4. Temperature-Aware Voltage Selection

In Fig. 3. we show the overall flow of our temperature-aware voltage selection approach. Given is a task graph mapped and scheduled on a multicore SoC, and the average switched capacitance for each task. A so called "assumed" temperature, at which each core is supposed to run, is also fixed as input. The voltage selection algorithm will determine, for each task, the voltage modes (V_{dd} and V_{bs}) such that energy consumption is minimized. Based on the determined voltage modes (and the switched capacitances known for each task) the dynamic power profiles are calculated and the thermal analysis is performed as discussed in Section 3. Depending on what the designer selects, a unique temperature or a dynamic temperature profile is determined for each core in the steady state. This new temperature/temperature-profile is now used again for voltage selection and the process is repeated until the temperature/temperature-profile converges. Once convergence has been reached, based on the determined voltage modes and temperatures, the minimized energy consumption E_{ta} (temperature aware energy) is calculated.

5. Experimental Results and Discussion

We have randomly generated applications consisting of 16 to 100 tasks. The size of each task is between 10^6 and $9{\cdot}10^6$ cycles, randomly distributed. The task graphs are mapped on SoC architectures consisting of 2 to 9 cores.

We have run experiments both considering only dynamic supply voltage selection and combined supply voltage and body bias selection. In the first case, 10 voltage levels were considered for V_{dd}, in the interval [0.6V, 1.8V]; for the second case 10 voltage modes (V_{dd}, V_{bs}) are used with V_{dd} and V_{bs} in the interval [0.6V, 1.8V] and [-0.1V, -1V], respectively.

The temperature model related coefficients for the SoC are given in Table 1. The parameters for leakage and frequency calculation (Equ. (1), (2), (3)) are the same as in [7] and [8].

Given a certain application and architecture, we run the temperature aware voltage selection algorithm illustrated in Fig. 3 and obtain the optimized energy consumption E_{ta}. For the same application and setting we run the voltage selection algorithm ignoring temperature, resulting in energy consumption E_{nta}. The temperature unaware voltage selection is realized by running one single iteration of the process in Fig. 3. By comparing E_{ta} with E_{nta} we can appreciate the efficiency of using a temperature aware voltage selection scheme.

In the case static thermal analysis was used, the total optimization time needed was less than 8 seconds. With dynamic analysis, of course, the optimization time was considerable higher, up to 350 seconds.

Table 1 Temperature model settings

Chip thickness	0.00025m
Chip size	0.001m*0.001m~0.009m*0.009m per processor
Ambient temperature	313.15K
Convection capacitance	140.4J/K
Convection resistance	0.1~0.6 K/W
Heat sink area	0.02m* 0.02m~0.03m* 0.03m
Heat sink thickness	0.005m~ 0.008m
Heat spreader area	0.01m* 0.01m~0.02m* 0.02m
Heat spreader thickness	0.001m~0.002m

5.1. Static vs. Dynamic Temperature Analysis

Voltage selection based on dynamic temperature analysis is more time consuming than the alternative using static analysis. However, dynamic analysis is more accurate and, thus, potentially can lead to huger energy savings.

We have applied both the static and the dynamic temperature analysis – based voltage selection to a set of applications and have compared the energy consumption produced by the two methods. Two categories of applications were investigated: 100 applications were such that the temperature oscillation on the cores, in steady state, was less than 5°; 250 applications produced temperature oscillations larger than 5°. Table 2 and Table 3 show the results.

Table 2 Static vs. dynamic analysis(small temperature oscillation)

Average improvement	Largest improvement	more than 1% difference	No improvement
0.01%	0.40%	0	69%

Table 3 Static vs. dynamic analysis(high temperature oscillation)

Average improvement	Largest improvement	more than 1% difference	No improvement
0.12%	5.11%	2.80%	30.1%

The obvious conclusion that can be drawn from the above experiments is that static thermal analysis is sufficiently accurate for the purposes of thermal aware voltage selection. This is the alternative we have used in the following experiments.

5.2. Temperature-Aware vs. Unaware DVS

Given a certain application, we define the *energy efficiency factor of the temperature aware DVS*, compared to the unaware one, as $G = (E_{nta}\text{-}E_{ta})/\ E_{nta}*100\%$. Obviously, E_{nta} depends on the *assumed temperature* provided by the designer. If the designer's guess is correct (equal to the temperature at which the chip functions with the selected voltages), a situation which is very unlikely, then $E_{nta} = E_{ta}$.

We have used 150 applications running at temperatures in the range 40°C to 100°C. It is assumed that the circuit cannot work above 140°C and, thus, the possible range of the designer's guess is between ambient temperature (35°C) and 140°C. The diagrams in Fig. 4 show the average value of the energy efficiency factor G as a function of how far the temperature guess is from the actual temperature at which the application runs. The experiments were run with combined V_{dd} and V_{bs} scaling, and they were performed considering two different cases for the dependency of leakage current on the temperature. For the first case we used the value $\gamma = -2223.7$ for the coefficient in equ. (2) (this is one typical value indicated in [13]). For the second case we considered $\gamma = -3223.7$, which indicates a higher degree of dependency of the leakage current on temperature (For all other experiments in the paper we considered $\gamma = -2223.7$). Fig. 5 shows the same experiments in the context of V_{dd} only scaling.

As can be seen, important energy savings can be achieved by thermal aware DVS. This is the case both for V_{dd} only and for combined V_{dd} and V_{bs} scaling. It is interesting to observe that, in the case of V_{dd} only, when temperatures are underestimated, the energy losses are smaller. The explanation is the following: When temperatures are overestimated, the temperature unaware approach assumes that leakage currents are very high (due to the high assumed temperature). Thus, the voltage selection algorithm will tend to select high supply voltages so that tasks are terminated early and slack time is used to put the circuit into low leakage modes. Since, in reality, the circuit will work at lower temperature and leakage currents will be considerably smaller (due to the exponential dependency of leakage on temperature, which at high temperature values leads to larger errors than at low temperatures), the temperature aware approach will produce smaller supply voltages, which explains the energy differences at overestimated temperature. In the case of temperature underestimations, the V_{dd} only approach will produce lower voltages (which extend the execution time in the limits of available slack) and, by this, find solutions that are close to those produced by the temperature aware approach.

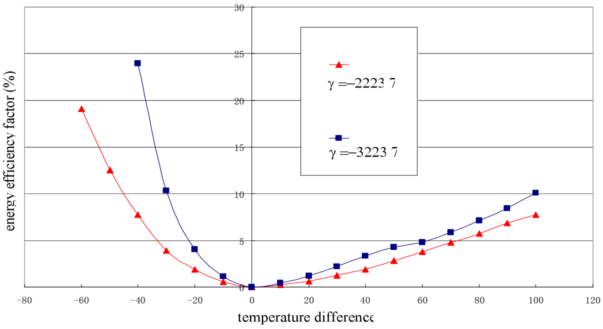


Fig. 4 Energy efficiency factor with combined V_{bs} and V_{dd} scaling

In the case of the combined approach, however, which, in addition to the V_{dd} only technique, has the opportunity to control leakage by adapting the body bias voltage V_{bs}, the temperature aware approach makes a considerable difference, both in the case of temperature under - and overestimations.

The diagrams in Fig. 4 and Fig. 5 also indicate that, as the dependency of leakage on temperature grows, the difference made by the temperature aware scaling technique becomes more significant.

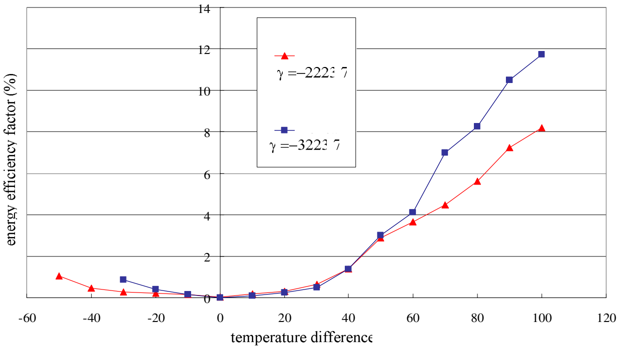


Fig. 5 Energy efficiency factor with V_{dd} only scaling

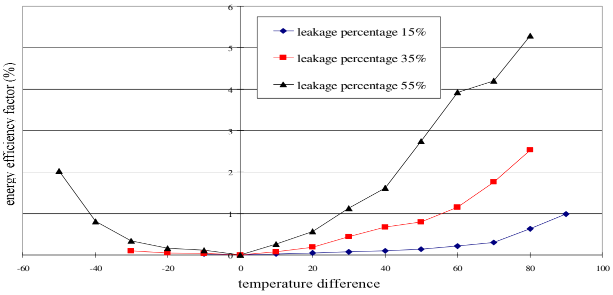


Fig. 6 Dependence on leakage percentage, V_{dd} only scaling

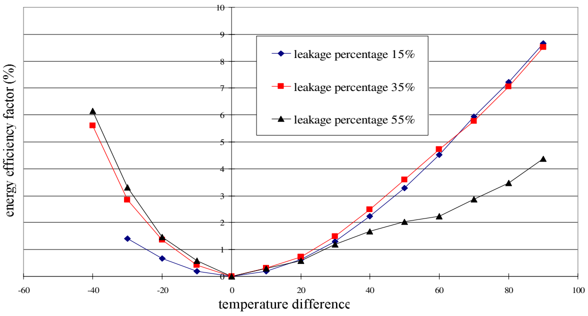


Fig. 7 Dependence on leakage percentage, V_{bs} and V_{dd} scaling

For the above experiments, the amount of leakage power (calculated at 70°C) was, on average, 50% of the total power. In a next set of experiments we have investigated the dependency of the factor *G* on the amount of leakage consumed by the circuit. We have performed our experiments with three different leakage percentage levels. The dependency is illustrated in Fig. 6 (V_{dd} only) and, Fig. 7 (combined V_{dd} and V_{bs} scaling). As expected, for a higher leakage percentage the temperature aware approach makes a larger difference. We have an interesting exception for combined V_{dd} and V_{bs} scaling in the case of temperature overestimations. The explanation is the following: In the case of high leakage percentage and if the assumed and real temperature are both high, both the temperature aware and the unaware scaling assume a very high leakage power and, thus, come close to producing that V_{dd} and V_{bs} combinations that forces down the leakage as much as possible and, by this, the produced voltage levels are becoming relatively similar. This similarity is as stronger as fewer execution modes are available on the processor.

5.3. Real Life Examples

We have investigated the efficiency of temperature aware voltage selection using two real-life examples: A GSM voice codec and a multimedia MPEG4 audio-video encoder. Details regarding the two applications can be found in [9] and [2], respectively. The GSM voice codec consists of 87 tasks considered to run on an architecture composed of 3 cores with 13 voltage modes. The MPEG4 consists of 109 tasks and is considered to run on 2 cores with 13 voltage modes.

The results are presented in Fig. 8 and they confirm the trends outlined by our previous experiments.

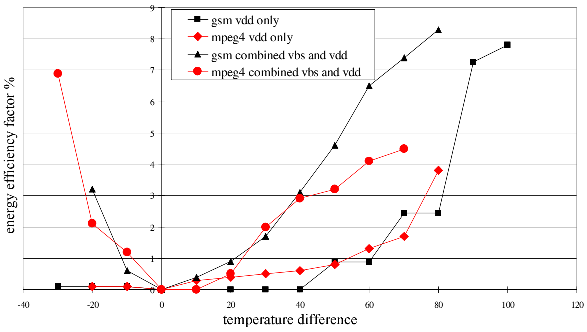


Fig. 8 Real life examples

6. Conclusions

We have presented an approach to thermal-aware voltage selection for energy minimization. We have shown that, besides having the potential to detect possible thermal runaway, a thermal aware approach can produce energy savings which can reach above 15%.

References

[1]. A. Andrei, P. Eles, Z. Peng, M. Schmitz, B. M. Al-Hashimi, Energy Optimization of Multiprocessor Systems on Chip by Voltage Selection, IEEE Transactions on Very Large Scale Integration Systems, 15(3):262-275, 2007

[2]. http://ffmpeg.mplayerhq.hu/

[3]. A. P. Chandrakasan, R. W. Brodersen, Low Power Digital CMOS Design. Norwell, MA: Kluwer, 1995.

[4]. W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, M. Stan, HotSpot: A Compact Thermal Modeling Methodology for Early-Stage VLSI Design, IEEE onVLSI Systems, 14(5):501-513, 2006.

[5]. W. P. Liao, L. He, and K. M. Lepak, "Temperature and supply voltage aware performance and power modeling at micro-architecture level," IEEE TonCAD, V24, no. 7, pp. 1042–1053, July 2005.

[6]. Y. Li, B. C. Lee, D. Brooks, Z. Hu, K. Skadron, CMP Design Space Exploration Subject to Physical Constraints, HPCA06, pp. 15-26, 2006.

[7]. Y. Liu, H. Yang, R.P. Dick, H. Wang, L. Shang, Thermal vs Energy Optimization for DVFS-enabled Processors in Embedded Systems, Int. Symp. on Quality Electronic Design (ISQED07), pp. 204 - 209, 2007.

[8]. S. Martin, K. Flautner, T. Mudge, D. Blaauw, Combined Dynamic Voltage Scaling and Adaptive Body Biasing for Lower Power Microprocessors under Dynamic Workloads, ICCAD, pp. 721-725, 2002.

[9]. M. Schmitz, B. Al-Hashimi, P. Eles, System-Level Design Techniques for Energy-Efficient Embedded Systems, Kluwer Ac. Publ., 2004.

[10]. K. Sankaranarayanan, S. Velusamy, M.R. Stan, K. Skadron, A Case for Thermal-Aware Floorplanning at the Microarchitectural Level, The Journal of Instruction-Level Parallelism, V7, Oct. 2005, pp. 1-16.

[11]. S. Wang, R Bettati, Delay Analysis in Temp.-Constrained Hard Real-Time Systems with General Task Arrivals, RTSS06, pp. 323-334.

[12]. Yuan Xie, Wei-Lun Hung, Temperature-Aware Task Allocation and Scheduling for Embedded Multiprocessor Systems-on-Chip Design, Journal of VLSI Signal Processing, 45(3), pp. 177-189, 2006.

[13]. Y. Yang, Z. Gu, C. Zhu, R. Dick, L. Shang, ISAC: Integrated Space-and-Time-Adaptive Chip-Package Thermal Analysis, IEEE TonCAD, 26(1), pp. 86-99, 2007.