We are IntechOpen,
the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Real-Time Systems

Ming Fan

Additional information is available at the end of the chapter

Abstract

Since 2004, most of chip vendors have begun to shift their major focus from single-core to multi-core architecture (W. Wolf. Signal Processing Magazine, IEEE, 26(6):50–54, 2009). One major reason of this shift is that it reaches a physical limit by scaling transistor size and increasing the clock frequency to improve the computing performance on a single-core architecture (Agarwal et al. Proceedings of the 27th International Symposium on, pages 248–259, June 2000), that is, the overall chip cannot be reached within a single clock cycle. Multi-core architecture, however, brings innovative and promising opportunities to further improve the computing performance. By providing multiple processing cores on a single chip, multi-core systems can dramatically increase the computing performance and mitigate the power and thermal issues with the same performance achievement as single-core systems. As multi-core architecture has been more and more dominant in the industrial market, there is an urgent demand for effective and efficient techniques for the design of multi-core systems.

In this chapter, we first analyze the thermal behavior on multi-core real-time systems by taking the heat transfer among different cores into consideration. Then we analyze the energy consumption for a given speed scheduling on multi-core systems.

Keywords: Real-Time, Multi-Core, Thermal, Power, Energy, Periodic Scheduling

1. Introduction

Today, multi-core architecture has been widely supported by most of major chip vendors, including Intel, AMD, IBM, Nvidia, ARM, Sum microsystems, Qualcomm, Broadcom, and so on. Some of the chip manufacturers have already launched 16-core chips into the market, that is, AMD OpteronTM 6300 Series [1]. It is not surprising that in the coming future, hundreds or even thousands of cores will be integrated into a single chip [22]. When moving toward

multi-core architecture, it comes with new and critical challenges in design of multi-core systems, particularly multi-core real-time systems.

As architecture becomes more and more complicated, besides the timing constraint, many other design constraints are taken into consideration in real-time system design and development. Traditional approaches focus exclusively on timing constraints [3,4,6,7]. Today, many other design constraints (e.g. power/energy, thermal, and reliability) also need to be considered seriously in real-time system design [8,9,11,12,14].
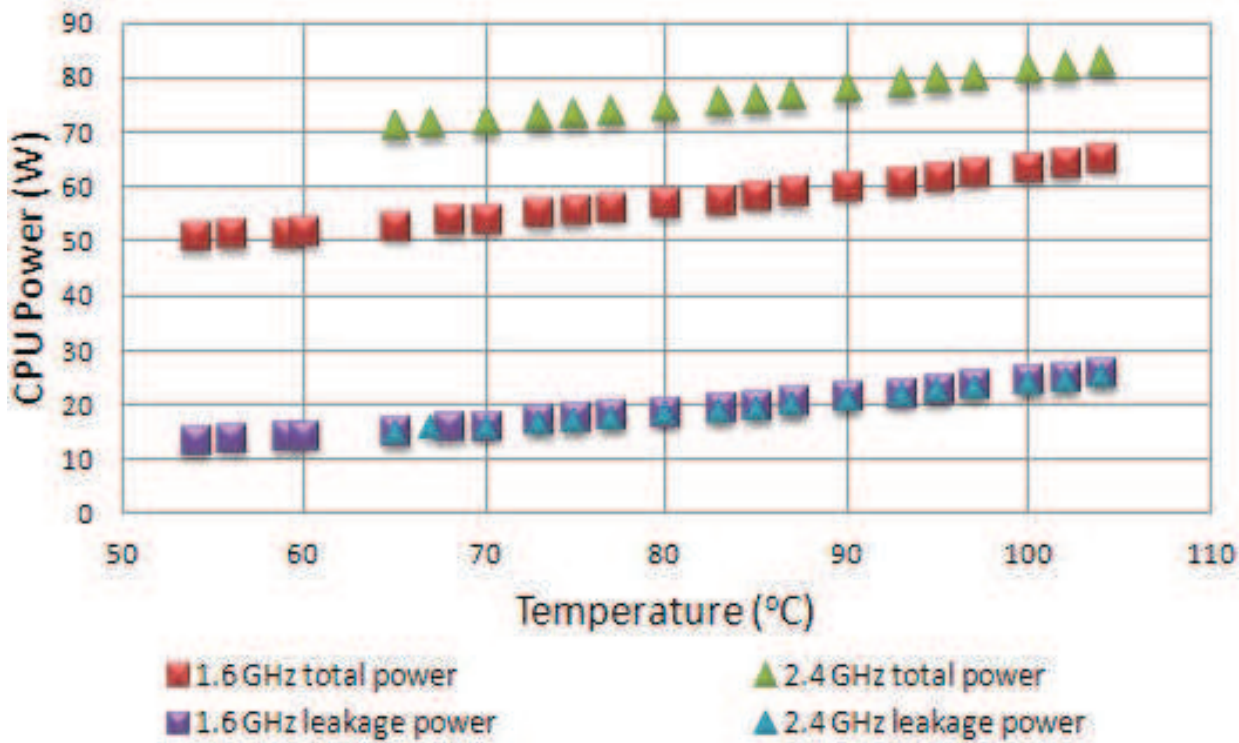

Figure 1. Power vs. Temperature [9]: Intel Core i5-2500K (32 nm Sandy Bridge), voltage 1.26 V, frequency at 1.6 GHz and 2.4 GHz, respectively.

1.1. Power/energy analysis in multi-core real-time systems

Catalyzed by continuous transistor scaling, hundreds of billions of transistors have been integrated on a single chip [13]. One of the immediate consequence caused by the tremendous increase of transistor density is the soaring power consumption [5], which further results in severe challenges in energy and temperature[11,17]. Today, power has become a critical and challenging design objective in front of system designers.

1.2. Thermal analysis in multi-core real-time systems

The continuously increased power consumption has resulted in a soaring chip temperature. Moreover, as design paradigm shifts to deep submicron domain, high chip temperature leads to a substantial increase in leakage power consumption [13], which in turn further deteriorates

the power situation due to the interdependency between temperature and leakage power. For instance, with Intel core i5-2500K (32 nm Sandy Bridge), the leakage power roughly grows up to 2× from 55°C (13 W) to 105°C (26 W), see **Figure 1**. Furthermore, the soaring chip temperature adversely impacts the performance, reliability, and packaging/cooling costs [17]. As a result, power and thermal issues have become critical and significant for advanced multi-core system design. In next section, we introduce some necessary backgrounds of multi-core scheduling with power and thermal awareness, respectively.

The aggressive semiconductor technology scaling has pushed the chip power density doubled every two to three years [16,20], which immediately results in an exponential increasing in heat density. High temperature can degrade the performance of systems in various ways. Therefore, there is a great need of advanced techniques for thermal/temperature aware design of multi-core systems.

2. Preliminary

2.1. Multi-core platform and task model

The multi-core platform consists of M processing cores, $M \geq 2$, denoted as **P, P** = $\{P_1, P_2, \ldots, P_M\}$. Each core P_i has N running modes, each of which is characterized by a 2-tuple set (v_k, f_k), where v_k represents the supply voltage and f_k represents the frequency under mode k, $1 \leq k \leq N$.

Let S represent a static and periodic speed schedule, which indicates how to vary the supply voltage and working frequency for each core at different time instants. A speed schedule S is constituted by several state intervals, which is described as below:

Definition 1 [25]: Given a speed schedule S for a multi-core system, an interval $[t_{q-1}, t_q]$ is called a state interval if each core runs only at one mode during that interval.

Recall that speed schedule S is a periodic schedule, let L denote the length of one scheduling period. According to Definition 1, a speed schedule S essentially consists of a number of non-overlapped state intervals. Let Q represent the number of non-overlapped state intervals within one scheduling period of S, then we have that [25]

1. $\cup_{q=1}^{Q} [t_{q-1}, t_q] = [0, L]$

2. $[t_{q-1}, t_q] \cap [t_{p-1}, t_p] = \varnothing$, if q ≠ p

For a single state interval $[t_{q-1}, t_q]$, let $\mathbf{K}_q$ represent the interval mode, $\mathbf{K}_q = \{k_1, k_2, \ldots, k_M\}$, where ki denotes the running mode of core P_i in interval $[t_{q-1}, t_q]$.

2.2. Power model

The overall power consumption (in Watt) of each core is composed of two parts: dynamic power P_{dyn} and leakage power P_{leak}. We assume that: (1) P_{dyn} is varied with respect of supply voltage but independent of temperature, (2) P_{leak} is sensitive to both temperature and supply voltage. Specifically, for the dynamic power, we know that it is proportional to the square of

supply voltage and linearity of working frequency. We assume that the working frequency is linearly proportional to supply voltage, thus the power consumption of the i-th core (P_i) under running mode k_i can be formulated as below [18].

$$P_{dyn,i} = \gamma_{ki} * v_{ki}^3 \tag{1}$$

where γ_{ki} is a constant value determined by the platform and running mode, and v_{ki} is the supply voltage of core P_i determined by the running mode.

For leakage power, although there is a very complicated relationship between leakage power and temperature from circuit level perspective, Liu et al. [23] found that a linear approximation of the leakage temperature dependency is fairly accurate. Work [12] further formulated the leakage power of core P_i as below:

$$P_{leak,i} = (\alpha_{ki} + \beta_{ki} * T_i(t)) * v_{ki} \tag{2}$$

where α_{ki} and β_{ki} are constants depending on the core running mode, that is, mode ki.

Consequently, the total power consumption of core P_i at time t, denoted as $P_i(t)$, can be formulated as:

$$P_i(t) = (\alpha_{ki} + \beta_{ki} * T_i(t)) * v_{ki} + \gamma_{ki} * v_{ki}^3 \tag{3}$$

For convenience in our presentation, we rewrite the above formula by separating the elements into temperature independent/dependent parts such that

$$P_i(t) = \psi_i + \phi_i * T_i(t) \tag{4}$$

where $\psi_i = \alpha_{ki} * v_{ki} + \gamma_{ki} * v_{ki}^3$ and $\phi_i = \beta_{ki} * v_{ki}$.

$$
\begin{bmatrix} P_1(t) \\ \vdots \\ P_M(t) \end{bmatrix} = \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_M \end{bmatrix} + \begin{bmatrix} \phi_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \phi_M \end{bmatrix} \begin{bmatrix} T_1(t) \\ \vdots \\ T_M(t) \end{bmatrix} \tag{5}
$$

Or

$$\mathbf{P}(t) = \mathbf{\Psi} + \mathbf{\Phi}\mathbf{T}(t) \tag{6}$$

Note that we use the bold text for a vector/matrix and the unbolded text for a value, for example, **T** represents a temperature vector while T represents a temperature value.
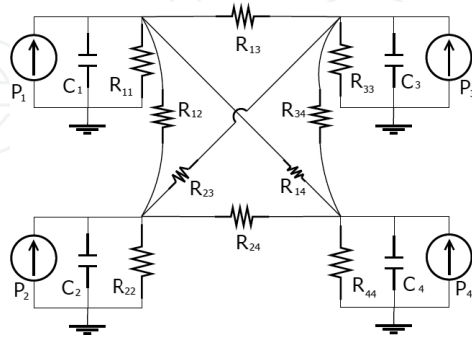
2.3. Thermal model

Figure 2. Illustration for thermal phenomena on multi-core system [26].

Figure 2 illustrates the thermal circuit model for a multi-core platform consisting of four processing cores. C_i represents the thermal capacitance (in Watt/°C) of core P_i, and R_{ij} represents the thermal resistance (in J/°C) between core P_i and P_j. Note that the thermal model adopted here is similar to the one used in related work [19,21]. Let T_{amb} represent the ambient temperature, then the thermal phenomena of core P_i in **Figure 2** can be formulated as

$$C_i \frac{dT_i(t)}{dt} + \frac{T_i(t) - T_{amb}}{R_{ii}} + \sum_{j \neq i} \frac{T_i(t) - T_j(t)}{R_{ij}} = P_i(t) \tag{7}$$

Let

$$\begin{cases} \delta i = \dfrac{T_{amb}}{R_{ii}} \\[2em] g_{ij} = \begin{cases} \sum_{k=1}^{M} \dfrac{1}{R_{ik}}, & \text{if j=i} \\[1.5em] \dfrac{-1}{R_{ij}}, & \text{otherwise} \end{cases} \end{cases} \tag{8}$$

Then the thermal model in equation (7) can be rewritten as

$$C_i \frac{dT_i(t)}{dt} + \sum_{j=1}^{M} g_{ij} T_j(t) = P_i(t) + \delta i \tag{9}$$

Accordingly, for the entire system, the thermal model can be represented as

$$\mathbf{C}\frac{d\mathbf{T}(t)}{dt} + \mathbf{g}\mathbf{T}(t) = \mathbf{P}(t) + \boldsymbol{\delta} \tag{10}$$

where $\mathbf{C}$ and $\mathbf{g}$ are MxM matrices

$$\mathbf{C} = \begin{bmatrix} C_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & C_M \end{bmatrix}, \mathbf{g} = \begin{bmatrix} g_{11} & \cdots & g_{1M} \\ \vdots & \ddots & \vdots \\ g_{M1} & \cdots & g_{MM} \end{bmatrix} \tag{11}$$

and $\boldsymbol{\delta}$ is an $Mx1$ vector

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_M \end{bmatrix} \tag{12}$$

Note that $\mathbf{C}, \mathbf{g},$ and $\boldsymbol{\delta}$ are all constants that are determined by the multi-core platform only. Moreover, $\mathbf{C}$ is the thermal capacitance matrix with none zero values only on the diagonal, and $\mathbf{g}$ is a thermal conductance matrix. The thermal model adopted here is a generic model which takes the heat transfer among different cores into consideration. Thus, it can be directly applied on thermal analysis for both temperature transient state and temperature stable state.

3. Temperature analysis in multi-core real-time systems

There is an interactive effect between power and temperature, that is, high power leads to high temperature, which in turn further aggravates the power consumption. In order to calculate the energy consumption accurately and efficiently, it is necessary to develop an efficient solution to calculate the temperature first.

In this section, we first present a temperature formulation within thermal transient state for a constant speed schedule interval. Then we present an analytical solution to calculate the temperature within thermal steady state for a periodic speed schedule.

3.1. Temperature analysis in system thermal transient state

In this subsection, we will formulate the temperature variation [26] within one state interval $[t_{q-1}, t_q]$ in the system thermal transient state.

First, by applying power model given by equation (6) into thermal model given by equation (10), we can derive that

$$\mathbf{C}\frac{d\mathbf{T}(t)}{dt} + \mathbf{g}\mathbf{T}(t) = \mathbf{\Psi} + \mathbf{\Phi}\mathbf{T}(t) + \mathbf{\delta} \qquad (13)$$

Then we simplify the above equation by letting $\mathbf{G} = \mathbf{g} - \mathbf{\Phi}$. Thus the above equation can be rewritten as

$$\mathbf{C}\frac{d\mathbf{T}(t)}{dt} + \mathbf{G}\mathbf{T}(t) = \mathbf{\Psi} + \mathbf{\delta} \qquad (14)$$

Since $\mathbf{C}$ represents the capacitance matrix, according to the circuit nature, we know that matrix $\mathbf{C}$ contains no zero values only on its diagonal. Thus, we can see matrix $\mathbf{C}$ is non-singular. Therefore, the inverse of $\mathbf{C}$, that is, $\mathbf{C}^{-1}$ exists. Then we can further simplify equation (14) into below

$$\frac{d\mathbf{T}(t)}{dt} = \mathbf{A}\mathbf{T}(t) + \mathbf{B} \qquad (15)$$

where $\mathbf{A} = -\mathbf{C}^{-1}\mathbf{G}$ and $\mathbf{B} = \mathbf{C}^{-1}(\mathbf{\Psi} + \mathbf{\delta})$. The system thermal model formulated by equation (15) has a form of first order Ordinary Differential Equations (ODE). Particularly, if all coefficients are constant, then there exists a solution which can be formulated as

$$\mathbf{T}(t) = e^{t\mathbf{A}}\mathbf{T}_0 + \mathbf{A}^{-1}(e^{t\mathbf{A}} - \mathbf{I})\mathbf{B} \qquad (16)$$

For a state interval $[t_{q-1}, t_q]$, it is important to point out that all coefficients in the above are constant. Specifically, let K_q be the corresponding interval mode, and let $\mathbf{T}(t_{q-1})$ be the temperature at the starting point of that interval. Then according to equation (16), the ending temperature of that interval, that is, $\mathbf{T}(t_{q-1})$, can be directly formulated as

$$\mathbf{T}(t_q) = e^{\Delta t_q \mathbf{A}_{Kq}}\mathbf{T}(t_{q-1}) + \mathbf{A}_{Kq}^{-1}(e^{\Delta t_q \mathbf{A}_{Kq}} - \mathbf{I})\mathbf{B}_{Kq} \qquad (17)$$

where $\Delta t_q = t_q - t_{q-1}$.

3.2. Temperature analysis in system thermal steady state

In this subsection, we will formulate the temperature variation [26] within one state interval $[t_{q-1}, t_q]$ in the system thermal steady state.

Consider a periodic speed schedule S, and let $\mathbf{T}(0)$ be the initial temperature at time instant 0. For an arbitrary state interval $[t_{q-1}, t_q]$ within speed schedule S, to obtain its steady-state temperature, one intuitive way is to trace the entire schedule S by consecutively calculating the temperature from the first scheduling period until system reaches its thermal steady state.

Although this approach works from theoretical point of view, when considering the practical computational time cost (which would be extremely expensive), it turns out that this intuitive approach would be inefficient or even impractical. Thus, it would be desirable and useful to develop an efficient solution to rapidly calculate steady-state temperatures for a periodic speed schedule.
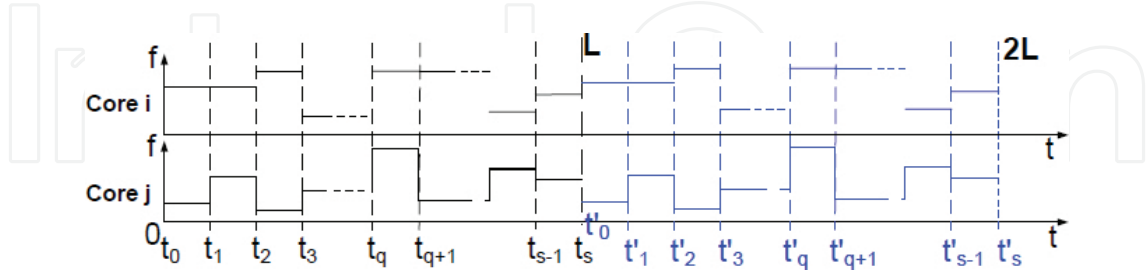

Figure 3. A speed schedule within two scheduling periods.

Let us first consider the temperature variation at the end of each scheduling period, that is, $t = nL$, where $n \geq 1$. Let the scheduling points of $S(t)$ in the first period be $t_0, t_1, \ldots, t_s$, respectively. After repeating $S(t)$, let the corresponding points in the second scheduling period be $t_0', t_1', \ldots, t_s'$, respectively (see **Figure 3**). Note that

$t_0 = 0$, $t_0' = t_s = L$, and $t_s' = 2L$. According to equation (17), at time t_1 and t_1', we have

$$\mathbf{T}(t_1) = e^{\Delta t_1 \mathbf{A}_{K1}} \mathbf{T}(t_0) + \mathbf{A}_{K1}^{-1}(e^{\Delta t_1 \mathbf{A}_{K1}} - \mathbf{I})\mathbf{B}_{K1} \tag{18}$$

$$\mathbf{T}(t_1') = e^{\Delta t_1' \mathbf{A}_{K1}} \mathbf{T}(t_0') + \mathbf{A}_{K1}^{-1}(e^{\Delta t_1' \mathbf{A}_{K1}} - \mathbf{I})\mathbf{B}_{K1} \tag{19}$$

Subtract equation (18) from (19) on both sides, and simplify the result by applying $\Delta t_1 = \Delta t_1'$, $t_0 = 0$ and $t_0' = L$, we get

$$\mathbf{T}(t_1') - \mathbf{T}(t_1) = e^{\Delta t_1 \mathbf{A}_{K1}}(\mathbf{T}(L) - \mathbf{T}(0)) \tag{20}$$

Follow the same trace of the above derivation, we have that

$$\mathbf{T}(t_2') - \mathbf{T}(t_2) = e^{\Delta t_2 \mathbf{A}_{K2}} e^{\Delta t_1 \mathbf{A}_{K1}}(\mathbf{T}(L) - \mathbf{T}(0))$$

$$\cdots \tag{21}$$

$$\mathbf{T}(t_s') - \mathbf{T}(t_s) = e^{\Delta t_s \mathbf{A}_{Ks}} \ldots e^{\Delta t_2 \mathbf{A}_{K2}} e^{\Delta t_1 \mathbf{A}_{K1}}(\mathbf{T}(L) - \mathbf{T}(0))$$

Since $t_s = L$, $t_s' = 2L$, and let $e^{\Delta t_s \mathbf{A}_{Ks}} \ldots e^{\Delta t_2 \mathbf{A}_{K2}} e^{\Delta t_1 \mathbf{A}_{K1}} = \mathbf{K}$, the last equation in (21) can be rewritten as

$$\mathbf{T}(2L) - \mathbf{T}(L) = \mathbf{K}(\mathbf{T}(L) - \mathbf{T}(0)) \tag{22}$$

In the same way, we can construct that

$$\mathbf{T}(xL) - \mathbf{T}((x-1)L) = \mathbf{K}^{x-1}(\mathbf{T}(L) - \mathbf{T}(0)) \tag{23}$$

where x = 1,2, …,n. Sum up the above n equations, we get

$$\mathbf{T}(nL) = \mathbf{T}(0) + \sum_{x=1}^{n} \mathbf{K}^{x-1}(\mathbf{T}(L) - \mathbf{T}(0)) \tag{24}$$

In the above, $\{\mathbf{K}^{x-1} | x = 1,2,\ldots,n\}$ forms a matrix geometric sequence. If $(\mathbf{I} - \mathbf{K})$ is invertible, then we have

$$\mathbf{T}(nL) = \mathbf{T}(0) + (\mathbf{I} - \mathbf{K})^{-1}(\mathbf{I} - \mathbf{K}^n)(\mathbf{T}(L) - \mathbf{T}(0)) \tag{25}$$

Next, we consider the temperature variation for an arbitrary time instant when repeating a periodic speed schedule S(t). Given a periodic speed schedule S(t), let t_q ($t_q \in [0,L]$) be an arbitrary time instant within schedule S(t). Moreover, let's repeat S(t) for n times, where $n \geq 1$. Let $\mathbf{T}(nL + t_q)$ denote the temperature of $\mathbf{T}(t_q)$ in the n-th scheduling period, by following the similar way of the above derivation, we can get that

$$\mathbf{T}(nL + t_q) = \mathbf{T}(t_q) + \mathbf{K}_q(\mathbf{I} - \mathbf{K})^{-1}(\mathbf{I} - \mathbf{K}^n)(\mathbf{T}(L) - \mathbf{T}(0)) \tag{26}$$

where $\mathbf{K}_q = e^{\Delta t_q \mathbf{A}_{Kq}} \ldots e^{\Delta t_2 \mathbf{A}_{K2}} e^{\Delta t_1 \mathbf{A}_{K1}}$.

Until now, we have been able to formulate the temperature variation of an arbitrary time instant in the n-th scheduling period. Next, we will further formulate the temperature variation of an arbitrary time instant in the system steady state. Consider an arbitrary time instant, that is, t_q, $0 \leq t_q \leq L$, within the first scheduling period. The brief idea of calculating the steady-state temperature corresponding to t_q is to let n go to infinity in equation (27). We formally describe our method in Theorem 1.

Theorem 1 [25]: Given a periodic speed schedule S(t), let $\mathbf{T}(L)$ and $\mathbf{T}(t_q)$ be the temperatures at time instant L and t_q, $t_q \in [0,L]$, respectively. If for each eigenvalue λ_i of $\mathbf{K}$, we have $|\lambda_i| < 1$, then the steady-state temperature corresponding to t_q can be formulated as

$$\mathbf{T}_{ss}(t_q) = \mathbf{T}(t_q) + \mathbf{K}_q(\mathbf{I} - \mathbf{K})^{-1}(\mathbf{T}(L) - \mathbf{T}(0)) \tag{27}$$

Proof:

First, based on equation (26), by letting n→+∞, the steady-state temperature of the q-th scheduling point in S(t) can be represented as

$$\mathbf{T}_{ss}(t_q) = \mathbf{T}(t_q) + \mathbf{K}_q(\mathbf{I} - \mathbf{K})^{-1}(\mathbf{I} - \lim_{n \to \infty} \mathbf{K}^n)(\mathbf{T}(L) - \mathbf{T}(0)) \tag{28}$$

When n→+∞, the matrix sequence $\mathbf{K}^n$ converges if and only if $|\lambda_i| < 1$, for each eigenvalue λ_i of $\mathbf{K}$ [14]. Under this condition, we have $\lim_{n \to \infty} \mathbf{K}^n = 0$. Moreover, if $|\lambda_i| < 1$ holds, then $(\mathbf{I}\text{-}\mathbf{K})$ is invertible. Thus, the steady-state temperature of the q-th scheduling point in S(t) can be further formulated as

$$\mathbf{T}_{ss}(t_q) = \mathbf{T}(t_q) + \mathbf{K}_q(\mathbf{I} - \mathbf{K})^{-1}(\mathbf{T}(L) - \mathbf{T}(0)) \tag{29}$$

Note that as n→+∞, unless the temperature runs away and causes the system to break down, we know that the system could eventually achieve its thermal steady state. That means for each eigenvalue λi of $\mathbf{K}$, the condition of $|\lambda i| < 1$ should always hold. Therefore, it is reasonable and practical to make such assumption shown in Theorem 1.

4. Energy analysis in multi-core real-time systems

Besides temperature, energy consumption is another important and challenging issue in the design of multi-core real-time systems. We have now been able to formulate the temperature variation in a multi-core system in the previous section. In this section, we will discuss how to formulate the energy consumption on multi-core systems with consideration of the inter-dependence between leakage power and temperature.

In the rest of this section, we first present an analytical solution to calculate energy consumption of an arbitrary state interval. Then we present a solution to calculate the total energy consumption of the entire speed schedule.

4.1. Energy analysis for one scheduling state interval

Consider a state interval, that is, [t_{q-1}, t_q] with initial temperature of $\mathbf{T}(t_{q-1})$. The energy consumption of all cores within that interval can be simply formulated as

$$\mathbf{E}(t_{q-1}, t_q) = \int_{t_{q-1}}^{t_q} \mathbf{P}(t)dt \tag{30}$$

Based on our system power model, given by equation (6), we have

$$\mathbf{E}(t_{q-1},\ t_q) = \Delta t_q \mathbf{\Psi} + \mathbf{\Phi} \int_{t_{q-1}}^{t_q} \mathbf{T}(t)dt \tag{31}$$

Theorem 2 given below is targeted to solve the above energy calculation problem.

Theorem 2 [25]: Given a state interval $[t_{q-1},\ t_q]$, let $\mathbf{T}_{q-1}$ be the temperature at time t_{q-1}. Then the overall system energy consumption within interval $[t_{q-1},\ t_q]$ can be formulated as

$$\mathbf{E}(t_{q-1},\ t_q) = \Delta t_q \mathbf{\Psi} + \mathbf{\Phi} \mathbf{G}^{-1}\mathbf{H} \tag{32}$$

where $\Delta t_q = t_q - t_{q-1}$, and $\mathbf{H} = \Delta t_q(\mathbf{\Psi} + \mathbf{\delta}) - \mathbf{C}(\mathbf{T}(t_q) - \mathbf{T}(t_{q-1}))$.

Proof:

In equation (31), let $\mathbf{X} = \int_{t_{q-1}}^{t_q} \mathbf{T}(t)dt$, then the energy formula can be rewritten as

$$\mathbf{E}(t_{q-1},\ t_q) = \Delta t_q \mathbf{\Psi} + \mathbf{\Phi} \mathbf{X} \tag{33}$$

On the other hand, according to the system thermal model given by equation (10), we have

$$\mathbf{C}\frac{d\mathbf{T}(t)}{dt} + \mathbf{G}\mathbf{T}(t) = \mathbf{\Psi} + \mathbf{\delta} \tag{34}$$

where $\mathbf{G} = \mathbf{g} - \mathbf{\Phi}$. By integrating on both sides of the above equation with respect to time t, where $t \in [t_{q-1},\ t_q]$, we can get

$$\mathbf{C}(\mathbf{T}(t_q) - \mathbf{T}(t_{q-1})) + \mathbf{G}\int_{t_{q-1}}^{t_q} \mathbf{T}(t) = \Delta t_q(\mathbf{\Psi} + \mathbf{\delta}) \tag{35}$$

Note that $\mathbf{X} = \int_{t_{q-1}}^{t_q} \mathbf{T}(t)dt$; thus from the above, we can derive that

$$\mathbf{X} = \mathbf{G}^{-1}\mathbf{H} \tag{36}$$

where $\mathbf{H} = \Delta t_q(\mathbf{\Psi} + \mathbf{\delta}) - \mathbf{C}(\mathbf{T}(t_q) - \mathbf{T}(t_{q-1}))$

Applying equation (36) into (33), we can see that

$$\mathbf{E}(t_{q-1},\ t_q) = \Delta t_q \mathbf{\Psi} + \mathbf{\Phi} \mathbf{G}^{-1}\mathbf{H} \tag{37}$$

From Theorem 2, we can see that for an arbitrary state interval $[t_{q-1},\ t_q]$, once the beginning temperature $\mathbf{T}(t_{q-1})$ is known, the total energy consumption within $[t_{q-1},\ t_q]$ can be easily and directly calculated.

Subsequently, given a periodic speed schedule S and an initial temperature $\mathbf{T}_0$, we are able to calculate the energy consumption within an arbitrary state interval in any scheduling period.

Corollary 1 [25]: Given a periodic speed schedule S(t) consisting of Q state intervals, let $\mathbf{T}_0$ be the initial temperature. Then the energy consumption within the q-th state interval in the n-th scheduling period, denoted as $\mathbf{E}(t_{q-1} + nL, t_q + nL)$, can be calculated as

$$\mathbf{E}(t_{q-1} + nL, t_q + nL) = \Delta t_q \mathbf{\Psi}_{Kq} + \mathbf{\Phi}_{Kq} \mathbf{G}_{Kq}^{-1} \mathbf{H}_{Kq} \tag{38}$$

where $\Delta t_q = t_q - t_{q-1}$, and $\mathbf{H}_{Kq} = \Delta t_q (\mathbf{\Psi}_{Kq} + \boldsymbol{\delta}) - \mathbf{C}(\mathbf{T}(t_q + nL) - \mathbf{T}(t_{q-1} + nL))$.

Corollary 1 can be easily derived from Theorem 2. With the help of Corollary 1, given any periodic speed schedule on a multi-core platform, when repeating that schedule, we can quickly calculate the energy consumption within any state interval.

Accordingly, given a periodic speed schedule, when analyzing the system thermal steady state, we can directly calculate the energy consumption of any state interval within the system steady state.

Corollary 2 [25]: Given a periodic speed schedule S(t) consisting of Q state intervals, let $\mathbf{T}_0$ be the initial temperature. Then the energy consumption within the q-th state interval in the system steady state, denoted as $\mathbf{E}_{ss}(t_{q-1}, t_q)$, can be calculated as

$$\mathbf{E}_{ss}(t_{q-1}, t_q) = \Delta t_q \mathbf{\Psi}_{Kq} + \mathbf{\Phi}_{Kq} \mathbf{G}_{Kq}^{-1} \mathbf{H}_{ss\,Kq} \tag{39}$$

where $\Delta t_q = t_q - t_{q-1}$, and $\mathbf{H}_{ssKq} = \Delta t_q (\mathbf{\Psi}_{Kq} + \boldsymbol{\delta}) - \mathbf{C}(\mathbf{T}_{ss}(t_q) - \mathbf{T}_{ss}(t_{q-1}))$.

Corollary 2 is directly derived from Corollary 1 by replacing the transient temperatures with steady-state temperatures.

Author details

Ming Fan[*]

Address all correspondence to: mf.mingfan@gmail.com

Broadcom Ltd, 3151 Zanker Road, San Jose, CA, United States of America

References

[1] AMD. Amd server processor series. pages http://www.amd.com/US/PRODUCTS/SERVER/PROCESSORS/6000–SERIES–PLATFORM/6300/Pages/6300–series–processors.aspx, 2013.

[2] J.H. Anderson, V. Bud, and U.C. Devi. An EDF-Based Scheduling Algorithm for Multiprocessor Soft Real-Time Systems. In Proceedings of Euromicro Conference on Real-Time Systems (ECRTS), July 2005.

[3] B. Andersson, S. Baruah, and J. Jonsson. Static-Priority Scheduling on Multiprocessors. In Proceedings of IEEE Real-Time Systems Symposium (RTSS), December 2001.

[4] M. Bao, A. Andrei, P. Eles, and Z. Peng. On-line thermal aware dynamic voltage scaling for energy optimization with frequency/temperature dependency consideration. In Design Automation Conference, 2009. DAC '09. 46th ACM/IEEE, pages 490–495, July 2009.

[5] V.N. Darera and L. Jenkins. Utilization bounds for rm scheduling on uniform multiprocessors. In Embedded and Real-Time Computing Systems and Applications, 2006. Proceedings of 12th IEEE International Conference on, pages 315–321, 0-0.

[6] M. Fan and G. Quan. Harmonic semi-partitioned scheduling for fixed-priority real-time tasks on multi-core platform. In Design, Automation Test in Europe Conference Exhibition (DATE), pages 503–508, March 2012.

[7] Q. Han, M. Fan, and G. Quan. Energy minimization for fault tolerant real-time applications on multiprocessor platforms using checkpointing. In Low Power Electronics and Design (ISLPED), 2013 IEEE International Symposium on, pages 76–81, September 2013.

[8] V. Hanumaiah and S. Vrudhula. Energy-efficient operation of multi-core processors by dvfs, task migration and active cooling. Computers (TC), IEEE Transactions on, pages 1–14, 2012.

[9] Henry. A comparison of intels 32nm and 22nm core i5 cpus: Power, voltage, temperature, and frequency. Intel Cor., pages http://blog.stuffedcow.net/2012/10/intel32nm–22nm–core–i5–comparison/, 2012.

[10] H. Huang and G. Quan. Leakage aware energy minimization for real-time systems under the maximum temperature constraint. In DATE, pages 1–6, 2011.

[11] H. Huang, G. Quan, J. Fan, and M. Qiu. Throughput maximization for periodic real-time systems under the maximal temperature constraint. In Design Automation Conference (DAC), 2011 48th ACM/EDAC/IEEE, pages 363–368, June 2011.

[12] ITRS. International Technology Roadmap for Semiconductors (2011 Edition). International SEMATECH, Austin, TX, http://public.itrs.net/.

[13] R. Jayaseelan and T. Mitra. Temperature aware task sequencing and voltage scaling. ICCAD, pages 618–623, 2008.

[14] S. Bell and J. Rosenblatt. Mathematical analysis for modeling, December 1998.

[15] G. Moore. Cramming more components onto integrated circuits. Electronics Magazine, 38(8):114–117, May 1965.

[16] G. Quan and V. Chaturvedi. Feasibility analysis for temperature-constraint hard real-time periodic tasks. Industrial Informatics, IEEE Transactions on, 6(3):329–339, 2010.

[17] J. Rabaey, A. Chandrakasan, and B. Nikolic. Digital Integrated Circuits: A Design Perspective. Englewood Cliffs, NJ: Prentice-Hall, 2003.

[18] S. Sharifi, R. Ayoub, and T.S. Rosing. Tempomp: Integrated prediction and management of temperature in heterogeneous mpsocs. In Design, Automation Test in Europe Conference Exhibition (DATE), 2012, pages 593–598, March 2012.

[19] K. Skadron, M.R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan. Temperature-aware microarchitecture. ICSA, pages 2–13, 2003.

[20] I. Ukhov, M. Bao, P. Eles, and Z. Peng. Steady-state dynamic temperature analysis and reliability optimization for embedded multiprocessor systems. In Design Automation Conference, June 2012.

[21] D. Yeh, L.S. Peh, S. Borkar, J. Darringer, A. Agarwal, and W. Hwu. Thousand-core chips [roundtable]. Design Test of Computers, IEEE, 25(3):272–278, 2008.

[22] Y. Liu and Y. Huazhong. Temperature-aware leakage estimation using piecewise linear power models. IEICE transactions on electronics, 93(12):1679–1691, 2010.

[23] W. Wolf. Multiprocessor system-on-chip technology. Signal Processing Magazine, IEEE, 26(6):50–54, 2009.

[24] V. Agarwal, M.S. Hrishikesh, S. Keckler, and D. Burger. Clock rate versus ipc: the end of the road for conventional microarchitectures. In Computer Architecture, 2000. Proceedings of the 27th International Symposium on, pages 248–259, June 2000.

[25] M. Fan, R. Rong, S. Liu, and G. Quan. Energy calculation for periodic multi-core scheduling in system thermal steady state with consideration of leakage and temperature dependency. The Journal of Supercomputing, 71(7):2565–2584, 2015.

[26] M. Fan, V. Chaturvedi, S. Sha, and G. Quan. An analytical solution for multi-core energy calculation with consideration of leakage and temperature dependency, Low Power Electronics and Design (ISLPED), 2013 IEEE International Symposium on, Beijing, pages 353–358, 2013.