Towards Robust and Cost-Effective Critical Real-Time Systems under Thermal-Aware Design

Javier Pérez Rodríguez

Patrick Meumeu Yomsi

CISTER-TR-190511

2019/07/09

Towards Robust and Cost-Effective Critical Real-Time Systems under Thermal-Aware Design

Javier Pérez Rodríguez, Patrick Meumeu Yomsi

CISTER Research Centre

Polytechnic Institute of Porto (ISEP P.Porto)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8321159

E-mail: perez@isep.ipp.pt, pamyo@isep.ipp.pt

https://www.cister-labs.pt

Abstract

The advent of multi-core platforms in critical realtime domains such as the avionics, automotive and railways to achieve higher and higher computing performances has turned the view on thermal concerns of the underlying chip die while it is still mandatory to meet all the temporal constraints. As a matter of fact, high chip temperature may not only degrade system performance and reliability, but it may also damage the chip permanently. We propose a methodology to address this problem, based on fixed task-to-core mapping and per-core analysis to derive a sound system model without feedback loops. To this end, it is important to have a better and deeper understanding of the existing thermal models in the literature. This is the main contribution of this research.

Towards Robust and Cost-Effective Critical Real-Time Systems under Thermal-Aware Design

Javier Pérez Rodríguez and Patrick Meumeu Yomsi
CISTER Research Centre, ISEP, Polytechnic Institute of Porto, Portugal
Email: {perez; pmy}@isep.ipp.pt

***Abstract*—The advent of multi-core platforms in critical real-time domains such as the avionics, automotive and railways to achieve higher and higher computing performance has turned the view on thermal concerns of the underlying chip die while it is still mandatory to meet all the temporal constraints. As a matter of fact, high chip temperature may not only degrade system performance and reliability, but it may also damage the chip permanently. In this paper, we propose a methodology to address this problem, based on fixed task-to-core mapping and per-core analysis to derive a sound system model without feedback loop. To this end, it is important to have a better and deeper understanding of the existing thermal models in the literature. This is the main contribution of this research.**

1. Introduction

For several decades now, critical real-time systems have consistently and continuously been under the spotlights of experts from both industry and academia. This is because they exposed stringent functional and non-functional requirements that have to be met, otherwise catastrophic consequences may occur. In general, these systems are modeled by using a finite set of recurrent tasks to be executed on a targeted hardware platform (e.g., the Intel Core2 from Intel, the 4-core Arm V7 Raspberry Pi 3 B and B+ from Arm; the TMS-320-C6678 from TI, the Tile-Gx3000 from Tilera, and the MPPA-256 architecture from Kalray) and each task commonly consists of a potentially infinite number of instances (jobs). Each job is characterized by four parameters: (1) a release time, which defines the instant time at which the job becomes available and ready for execution; (2) a worst-case execution time, which defines an upper-bound on the actual execution time of the job on the targeted platform; (3) a minimum inter-arrival time, which defines how frequent is the release of a new job[1]; and finally (4) a deadline, which defines a time window, from the release, wherein the job has to complete its execution. While each task's functional correctness is important for these systems, the time at which the result is produced is also central. To this end purpose, several factors have to be considered at the system design time. Examples include the task interactions, concurrency, and interference at the software level; and the mechanisms governing the execution of the tasks (preferably with a great level of details) at the hardware level. To date, an entire body of knowledge, techniques and methodologies have been proposed in the literature on the topic, some of which are now mature, especially for single core platforms. However, new challenges arise almost on a daily basis. This is due to the ever growing complexity and computational demand of the applications at the software level and/or the non-disclosure of valuable and detailed information on the targeted platform by the hardware vendors. Despites these noticeable limitations and the constant necessity for miniaturization of the emerging hardware components, we have been witnessing the integration of more and more processing elements in smaller silicon areas in order to achieve better performance. As a matter of fact, the integration scale has been doubling every three years [1], [2]. From a software viewpoint, this has resulted in forcing the processor to execute workloads at high frequencies most of the time. Hence, (i) the necessity for hardware miniaturization on one side; and (ii) the ever increasing computational demand of the applications on the other side, put together, have highlighted a serious problem: the soaring power dissipation of the integrated circuits, which in turn translates in temperature dissipation. Obviously, high temperatures create a number of problems, because transistors may fail to switch properly and therefore can lead to transient and/or permanent errors for the entire system. Specifically, an increment in the temperature until an uncontrolled value can affect drastically the runtime behavior of the tasks, and also the platform. This phenomenon holds true irrespective of hosting the execution of the tasks on a platform with a single or several cores. According to Borkar [1], the price for cooling down a watt of temperature in a processor is about $\$1-\$3$ or more. Consequently, this opens a broad avenue for research for the design of cost-effective and more robust critical real-time systems in critical real-time domains such as the avionics, automotive and railways. To the best of our knowledge, the thermal problem for critical real-time systems has been addressed in the literature by either switching off some core(s) [3], [4] or by re-scaling the cores speed [5], [6], [7], [8]. Roughly speaking, this means that the thermal management of the platform is handled

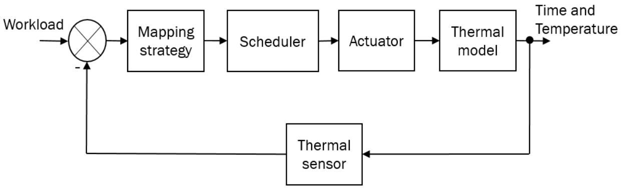

Figure 1. Classical control block diagram for thermal management.

[1] If the frequency is constant, then the task is said to be periodic.

by using a feedback control block diagram as illustrated in Figure 1. Here, action is taken only when the reported temperature by the thermal sensor rises above a predefined threshold. Below the threshold, no specific optimization and/or workload distribution strategy is used to maintain both the temporal and thermal behavior of the system. As a consequence, the time spent in cooling down the system may cause temporal changes in the original tasks schedule and then jeopardize the schedulability. Furthermore, not all platforms can support speed re-scalability, unfortunately. In this work, we argue and believe that the problem must be addressed from a different angle. In our opinion, it is possible to create a new "correct-by-construction" framework, preferably unique, wherein we model under the same umbrella both the temporal and thermal "on-core" and "un-core" activities for each processing element. For a given mapping, the on-core model will capture the activity (temporal and thermal) of the core under analysis, whereas the un-core model will capture the interference (temporal and thermal) imposed by other processing elements and share resources. As a result, it will become easier to derive an analysis that predicts the run-time behavior of the entire system without any need of a feedback loop (see Figure 2).

2. Problem statement

Nowadays, multi-core platforms are pervasive in numerous critical real-time systems due to the enormous computing capabilities they offer. While meeting domain-specific standards' requirements (e.g., the ARINC-653 and DO-178C in the avionics; and the ISO-26262 in the automotive) in terms of temporal requirements, our main objective is to address the following question. As the adoption of a multi-core platform exposes the underlying chip die to several heating sources and the temperature of each core can interfere with the thermal dissipation of the neighboring cores, how to adapt and/or design a robust and cost-effective thermal model of the platform that can easily be coupled with the adopted temporal model of the application so as to make it possible for the system designer to capture in an accurate manner both the chip-wide and the localized thermal behaviors of the system at run-time? The derived thermal model, associated with the temporal model, will allow for a sound thermal-aware schedulability analysis for the entire system.

3. Overview of existing thermal models

Before going into details, it is worth mentioning that power models as described in the literature have failed to manage temperature, despite the well-known duality between heat transfer and electrical phenomena. Consequently, to pave the way towards a convincing solution to the aforementioned problem, an educated strategy commands us to proceed by exploring all the thermal models that have been proposed in the literature in first place. In this regard, only two thermal models have been developed to the best of our knowledge: (1) a *coarse-grained* model referred to as TEMPEST [9], which uses a Resistance-Capacitance (RC) *parallel circuit* representation; and (2) a *fine-grained* model referred to as HotSpot [10], [11], which uses a RC *serial circuit* representation. Below, we briefly discuss their advantages and disadvantages.

- ▷ **TEMPEST.** This model has been proposed by Dhodapkar et al. [9]. Here, temperature is tracked only at a macro-architectural level, i.e., at the chip level. Consequently, this model is not flexible and allows only for chip-wide thermal-aware techniques such as Dynamic Voltage and Frequency Scaling (DVFS) [12] and Fetch Toggling [13] for reducing the processor peak temperatures. On the positive side, TEMPEST is easily portable to new hardware architectures and simple to implement because it makes it possible to safely bound the temperature of the underlying platform irrespective of the localization of eventual hotspots and it is agnostic to the hardware run-time mechanisms. However, it has been proven that localized heating occurs much faster than chip-wide. In this case, chip-wide treatments are too conservative, unfortunately.
- ▷ **HotSpot.** This model has been proposed by Skadron et al. [10]. In contrast to TEMPEST, temperature is tracked at the granularity of individual micro-architectural units and the equivalent RC circuits have at least one node for each unit. As such, this model allows for the detection of hotspots and to promptly activate a thermal response. The system designer can operate at block-level on the underlying platform or even below, and so, he can capture and handle the effects of hotspots more accurately. However, the model is way less portable and much more complex to implement as it requires a detailed understanding of the mechanisms governing the run-time behavior of most hardware components (e.g., branch predictor, load-store queue, D-cache etc.). In addition, the sampling rate at which the detection of new hotspots is performed have to be closely scrutinized as it plays a central role here, unfortunately.

4. Envisioned approach

From the discussion conducted in Section 3, it follows that the HotSpot model exposes better features than TEMPEST for the design of an accurate thermal-aware management technique upon a multi-core platform. However, the right level of abstraction that would make it unnecessary to model all the micro-architectural units and still achieve a sound analysis is missing. To fill this gap, we plan to proceed in three phases as follows.

First, we plan to revisit the task-to-core mapping strategies available in the literature in order to take into account the thermal profile of each task in our mapping procedure. During this phase, we will promote mapping strategies for which the increase of the overall platform temperature is as minimum as possible. This will be achieved by using a stochastic-based approach for example. Second, for the resulting mapping, we will adopt a per-core analysis and build a unique "correct-by-construction" framework wherein we model both the temporal and thermal "on-core" and "un-core" activities for each processing element. Our combined system model

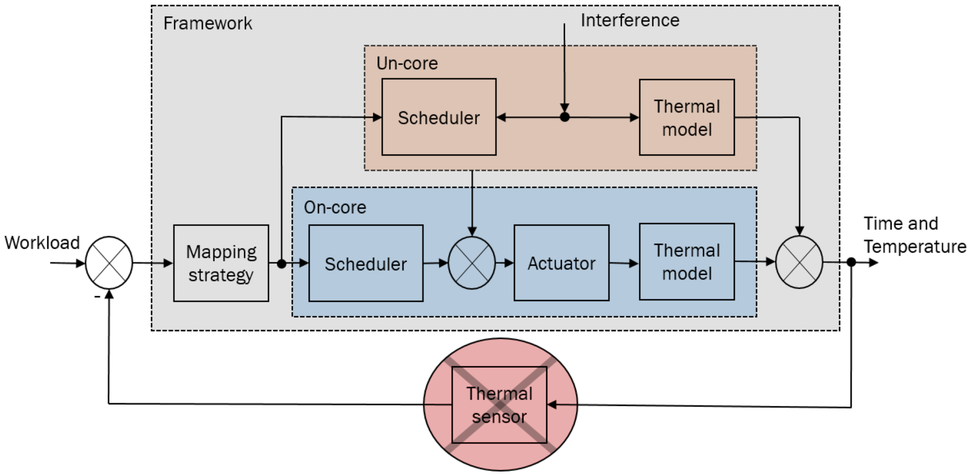

Figure 2. Envisioned control block diagram for thermal management.

will allow us not only to guarantee soundness, but also to optimize for thermal efficiency and thus costs. The on-core model will capture the activity (temporal and thermal) of the core under analysis, whereas the un-core model will capture the interference (temporal and thermal) imposed by other processing elements and share resources. Finally, we will derive an analysis that predicts the run-time behavior of the entire system from a temporal and thermal viewpoint without any need of a feedback loop (see Figure 2). Note that in presence of such a feedback loop, the imprecision of the thermal sensor may lead to an optimistic, if not wrong, analysis. Consequently, our proposed approach will rely on strong mathematical foundations based on an open control loop with perturbations for each core.

5. Conclusion

In this work, we detailed our research roadmap for the design of a robust and cost-effective critical real-time system under thermal-aware design. We revisited the thermal models available in the literature and briefly discussed their advantages and disadvantages. We reached the conclusion that the HotSpot thermal model exposes the most promising features to help us meet our objectives both from a temporal and thermal viewpoint, but it requires some adjustments. The main challenge is to find the correct level of abstraction that would make it unnecessary to model the thermal behavior of all micro-architectural units. Finally, we elaborated on the directions that we are planning to explore to derive our thermal-aware schedulability analysis for multi-core platforms.

Acknowledgment

This work was partially supported by National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology), within the CISTER Research Unit (UID/CEC/04234); by the European Union through the Clean Sky 2 Joint Undertaking, under the H2020 Framework Programme (H2020-CS2-CFP08-2018-01), grant agreement nr. 832011 (THERMAC).

References

[1] S. Borkar, “Design challenges of technology scaling,” *IEEE Micro*, vol. 19, no. 4, pp. 23–29, July 1999.

[2] R. Mahajan, “Thermal management of CPUs: A perspective on trends, needs and opportunities.” in *Proceedings on 8th International Workshop on THERMal INvestigations of ICs and Systems (THERMINIC)*, 10 2002.

[3] P. Kumar and L. Thiele, “Thermally optimal stop-go scheduling of task graphs with real-time constraints,” in *16th Asia and South Pacific Design Automation Conference (ASP-DAC 2011)*, Jan 2011, pp. 123–128.

[4] Y. Chandarli, N. Fisher, and D. Masson, “Response time analysis for thermal-aware real-time systems under fixed-priority scheduling,” in *IEEE 18th International Symposium on Real-Time Distributed Computing*, April 2015, pp. 84–93.

[5] N. Bansal, T. Kimbrel, and K. Pruhs, “Dynamic speed scaling to manage energy and temperature,” in *45th Annual IEEE Symposium on Foundations of Computer Science*, Oct 2004, pp. 520–529.

[6] N. Bansal and K. Pruhs, “Speed scaling to manage temperature,” in *STACS*, V. Diekert and B. Durand, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 460–471.

[7] S. Wang, Y. Ahn, and R. Bettati, “Schedulability analysis in hard real-time systems under thermal constraints,” *Real-Time Systems*, vol. 46, pp. 160–188, 10 2010.

[8] Y. Fu, N. Kottenstette, C. Lu, and X. D. Koutsoukos, “Feedback thermal control of real-time systems on multicore processors,” in *Proceedings of the Tenth ACM International Conference on Embedded Software*, ser. EMSOFT. New York, NY, USA: ACM, 2012, pp. 113–122.

[9] A. Dhodapkar, C. How Lim, G. Cai, and R. Daasch, “TEM2P2EST: A Thermal Enabled Multi-model Power/Performance ESTimator,” in *Power-Aware Computer Systems*, 06 2001, pp. 112–125.

[10] K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, “Temperature-aware microarchitecture: Extended discussion and results,” in *In Proceedings of the 30th Annual International Symposium on Computer Architecture*, 07 2003, pp. 2–13.

[11] W. Huang, M. R. Stan, K. Skadron, K. Sankaranarayanan, S. Ghosh, and S. Velusamy, “Compact thermal modeling for temperature-aware design,” in *41st Design Automation Conference.*, July 2004, pp. 878–883.

[12] D. R. Sulaiman, M. Ibrahim, and I. Hamarash, “Dynamic voltage frequency scaling (DVFS) for microprocessors power and energy reduction,” *4th International Conference on Electrical and Electronics Engineering*, 12 2005.

[13] K. Skadron, T. Abdelzaher, and M. R. Stan, “Control-theoretic techniques and thermal-RC modeling for accurate and localized dynamic thermal management,” in *Proceedings Eighth International Symposium on High Performance Computer Architecture*, Feb 2002, pp. 17–28.