Design of an Efficient Interconnection Network of Temperature Sensors

Pablo Ituero Marisa López-Vallejo Miguel Ángel Sánchez Marcos Carlos Gómez Osuna
Dpto. de Ingeniería Electrónica, ETSI Telecomunicación, Universidad Politécnica de Madrid
Ciudad Universitaria s/n, 28040 Madrid, Spain
{pituero,marisa,masanchez,cgomez}@die.upm.es

***Abstract*—Temperature has become a first class design constraint because high temperatures adversely affect circuit reliability, static power and degrade the performance. In this scenario, thermal characterization of ICs and on-chip temperature monitoring represent fundamental tasks in electronic design. In this work, we analyze the features that an interconnection network of temperature sensors must fulfill. Departing from the network topology, we continue with the proposal of a very light-weight network architecture based on digitalization resource sharing. Our proposal supposes a 16% improvement in area and power consumption compared to traditional approaches.**

I. Introduction

Current nanometer technologies have allowed extraordinary integration densities in digital circuits. However, as technology scales down to 90 nm and below power densities and operating temperatures of the circuits continue to rise at an alarming rate. Dynamic power and thermal management (DPM and DTM) appeared as solutions to avoid spacial and time distributed hotspots to sustain current performance improvement trends deep into the nanometer regime. Moreover, faults due to failure mechanisms like negative-bias temperature instability (NBTI) and time-dependent dielectric breakdown (TDDB) present an exponential dependence on temperature. This translates into shorter circuits lifetimes, what should be considered by DPM/DTM policies to maintain the levels of reliability and life-expectancy that consumers have come to expect.

In this context, proactive approaches become a must. Designers should develop architectures adaptable to variations of all kinds, which rely heavily on information gathered from on-chip monitoring circuits. Information from in situ monitors is trustable, much better than indirect measures [1]. Thus, thermal sensors that can monitor spacial and temporal hotspots as the circuit ages become critical [2]. Specially in complex circuits like multi-core processors, SoC or NoCs, where there is room to adapt the system performance or workload to the particular thermal circumstances that occur. However, allocating an arbitrarily large number of such monitors will not only create a significant area overhead, but routing the data from the sensor registers to a central processing unit will also pose a challenge [3].

Our work tries to cover this section of the new adaptive techniques based on real-time monitoring that has not been fully approached by the scientific literature yet. Specifically, we present a new temperature monitoring network architecture based on the concept of digitalization resource sharing. The network displays the following features:

- Ultra light weight. The impact on area and power of the network should be negligible. Moreover, no performance penalty is allowed.
- Multi-purpose. The network can be used both for calibration and monitoring.
- Simplicity and scalability. The hierarchical structure of the network allows efficient handling of information by the system.
- A comfortable interface to higher abstraction layers is also provided. Actually, a common/shared point to calibration and readings for all monitors is a design goal.

The structure of the paper is the following. First, the network technical requirements are analyzed. After that, section IV describes the network structure and the proposed topology and digitalization resource sharing technique. Finally, experimental results are presented and some conclusions are drawn.

II. Temperature Monitoring Technical Requirements

Designers must face nanometer challenges by providing thermal-efficient systems that balance or equally distribute, in time and space, possible on-chip hot spots. In this scenario, DTM techniques arise as a promising solution. DTM relies on accurately sensing and managing on-chip temperature, both in space and time, by optimally allocating smart temperature sensors in the silicon.

The range of temperatures that these sensors must cover will depend on the technology and the applications that the IC is targeting. In the case of commercial ICs exposed to standard conditions, the monitor should provide a thermal feedback covering a range from room temperature to the maximum acceptable junction temperature (MAJT), which is a technology dependent parameter. E.g. the MAJT for the Intel i5 is 111^oC, according to its datasheet, therefore the range of temperatures for a thermal monitor in this processor should be $25 - 111^oC$.

Accuracy is a very important issue in thermal monitoring. Underestimation will result in the circuit operating at a harmful temperature, whereas overestimation will impact performance

This work was funded by the CICYT project DELTA TEC2009-08589 of the Spanish Ministry of Science and Innovation.

negatively since the DTM policies, such as dynamic voltage and frequency scaling (DVFS), would be unnecessarily triggered [4]. Current sensor implementations are capable of providing accuracy levels around 1^oC [5], which translates into 2 W of CPU power in desktop computers.

Temperature resolves on the order of milliseconds and has a spatial constant around $1mm$ [6], these figures give an idea of the range of sampling frequencies and the number of sensors that an accurate DTM system should have. E.g. in the extreme case that one sensor was placed for every square mm, and supposing the die size of an Intel i7, the number of sensors needed by the system would be in the order of 250.

III. Network Overview

From the requirements that thermal sensors demand, now we identify the characteristics that a network of on-chip thermal monitors must fulfill. Specifically we target the on-chip network that connects a set of thermal monitors —or nodes— not necessarily evenly distributed [3], to a control system that processes their information. It is interesting to consider that the network should be prepared to deal with different kinds of nodes, since the system might need several types of monitors. E.g. consider the case when a DTM policy needs very localized thermal information at certain points of the chip but also average thermal information over a bigger area; probably two different sensors would be needed.

Given that the network is the only way to access the monitors, it has to accomplish two main functions: monitoring and calibration. Considering the former, the network needs to provide sufficient bandwidth so as to send all the data that the sensors yield at the rate that the control policies require. As far as the calibration is concerned the network has to provide support for all the actions that are required during the calibration process.

We have identified several features that are fundamental for a network of this kind. First, and most obvious, from the electronic viewpoint, the network must suppose a small overhead in terms of area, reliability, power and self-heating of the system. Routing and processing the information produced by a vast number of sensors in most cases will be the limiting factor that sets the upper bound to sensors data rate [3]. A solution that fulfills the systems requirements, will necessarily go through a trade-off between area —especially the routing of each sensor to a processing unit— and power —mainly dependent on the data frequency of the interconnection lines.

The flexibility to host as many different types of sensors as possible is another key characteristic that will allow the network to be implemented in a variety of systems. A consequence of this need for adaptability is the requirement for standard interfaces that not only go towards the sensor-ends but also cover the network-OS and the network-PCB interfaces.

Another feature is the ability of the network to deal with hierarchy levels. Specifically, in [3] a distinction between global and local monitoring is made. The purpose of global monitoring is to track the critical variations on the core to help the control intervene at general-emergencies, such as surpassing of the safe limit in the junction temperature. Local monitoring aims to establish a detailed map of the information of each monitor that, as a whole can be interpreted by the control mechanisms.

Yet another feature is the dynamic sensor-selection to avoid collecting data from those sensors that will not provide useful information, as proposed in [4]. Ideally, the network should prevent the sensor from working whenever its information would not be used.

Concerning previous works, apart from many approaches that just employ point-to-point connections to reach their temperature sensors, the first innovative approach that we found in the literature is that by Székely et al [7]. This pioneer work established all the basis of thermal-aware electronic design from thermal simulation to thermal monitoring. They proposed to insert the thermal test circuitry into the boundary-scan architecture and compare all the temperatures to a maximum rating. This idea of connecting all the monitors through a one-wire chain emulating a global shift-register imposes a lower bound in the routing of the network and has been employed frequently in the literature. Recent standards used in state-of-the-art processors, such as the Platform Environment Control Interface (PECI) by Intel, also make use of single-wire interfaces with the monitors.

A recent work in thermal monitoring, [8], has proposed a starred network topology that connects each of the sensing nodes to a central node. The transmission is performed in parallel and to diminish the elevated number of interconnection lines that are required, the measurements undergo a compression stage —specifically they execute a reduction from eight to four bits. Still the architecture supposes a big amount of connections and furthermore there is a significant loss in precision that could not be acceptable for the DTM policies. This implies that any comparison with this work of a scheme that does not employ any compression would lead to unfair erroneous results.

IV. Proposals

In this work we approach the topic of monitor networks on-chip from two different perspectives. Two contributions aiming at different aspects of the field:

- The network topology
- The digitalization resource sharing

The next subsections describe each of them.

A. Network Topology

Even when there have been many contributions in the field of the Networks-on-Chip (NoC), the scientific literature has paid so far little attention to the topology of on-chip networks specifically designed for monitoring. Among the several papers dealing with DTM policies that require a certain number of sensors, very few put forward any kind of network interface, and those that do it use a simple point-to-point connection between the sensors and the central control.

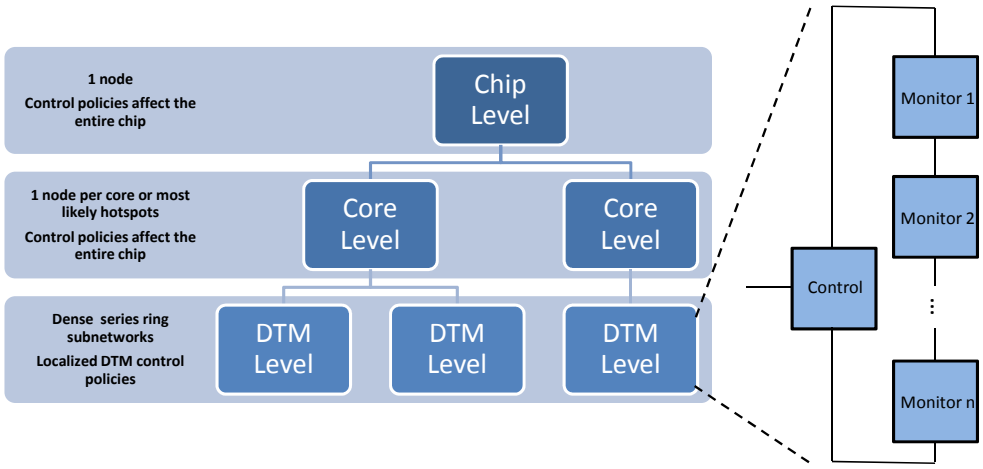


Fig. 1. Proposal for the hierarchy model of the sensor network.

Our vision of this problem embraces a hierarchy model that divides the network into three levels of complexity. The proposal is shown in figure 1. Depending on the thermal management requirements, the designer would decide to implement all three hierarchy levels or just one or two. For example, for a circuit with limited thermal issues, just the upper level controlling if the peak temperature surpasses a security bound temperature would be enough. In contrast, for a system with complex DTM policies that require the early detection of hot spots, all three levels would be necessary. Next, we describe each of the three hierarchy levels.

The top of the hierarchy of our model is a single sensing node that sends simple monitoring information to a central control with access to the OS and PCB interfaces. The kind of information produced at this level entails current peak on-chip temperatures. This configuration was used by early processors, such as the Intel® Centrino [9]. This processor uses a fixed thermal sensor, an analog diode, tuned to the max specified junction temperature. In case of abnormal conditions, such as cooling system malfunction, the circuit asserts a signal that activates a programmable self management power saving action that protects the CPU from operating out of its specified thermal range [10].

Concerning the second level of the hierarchy, we propose a small set of sensing nodes placed near most likely hot spots linked by point-to-point series connections to the central control of the first level. In this case the control is ready to provide more data, such as the average temperature which helps detect spatial thermal gradients. At this level, still, the control asserts data that will affect the behavior of the whole core. A similar configuration is found in the Intel® Core^{TM} Duo architecture [11], where in addition to the analog thermal diode, multiple sensing devices are distributed throughout the die in the possible hot spots [10].

Continuing with our proposal, as shown in figure 1, departing and ending from each second-level node, in the third level there is a series ring sub-network connecting several third-level sensing nodes. This level is thought for dense sensor networks with complex DTM policies that include localized actions with just the information of sensors from a single sub-network.

In a system with several levels of hierarchy, the upper levels would extract the data they need from the information that is sent from the lower levels. E.g. the maximum temperature of a core that is needed in the second level corresponds to the maximum of the measurements realized by the set of sensors in the lower level. Supposing a centralized control, all the information collected by lower levels, which in the case of dense networks may mean an important bandwidth, must be delivered to the top level.

Let us focus on the third and lowest level of the hierarchy, which is the one that has to deal with more network nodes, and thus it is where the network gets more complicated or equivalently where there are more opportunities of improvement. So far, the best solution for this kind of networks found in the scientific literature and the one adopted by the industry is a boundary-scan like network similar to the one depicted in figure 2. Now, we are going to consider the most simple implementation of these boundary-scan like proposals in which at each round all the bits from all the monitors are transmitted sequentially over the same connection, as depicted in figure 3. In this case, the maximum number of monitors, n, connected to the same line is limited by

$$n < \frac{f_{clock}}{f_s q} \tag{1}$$

where f_{clock} is the clock frequency of the system, f_s is the sampling frequency, and q the number of bits in each measurement.

In the next section we describe a new network architecture based on the concept of digitalization resource sharing that supposes an important improvement in area and power when compared to the boundary-scan like scheme.

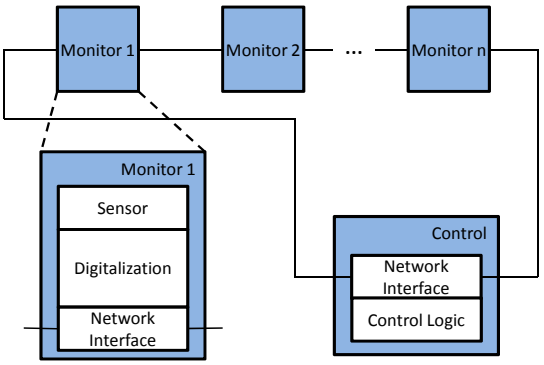


Fig. 2. Network architecture for the boundary-scan style.

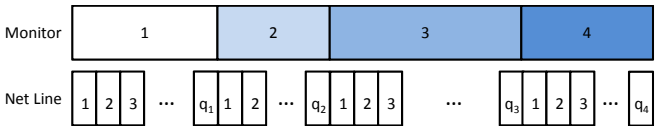


Fig. 3. Functioning of a basic boundary-scan like network.

B. Digitalization Resource Sharing

For each monitor of the network there is an interface that turns an analog signal into a digital one, normally by means of an ADC or a time-to-digital converter. Let us, thus, divide each monitor into a sensing block and a digitalization block. Interestingly, the part that normally occupies more area and consumes more power is the digitalization block [5].

Our proposal in this area is to make several sensors share the digitalization resources. In some cases, the nature of the analog signal will prevent this solution —e.g. an ADC that takes a voltage as input could have serious sensitivity issues if it is placed far from the sensing block. However, with certain types of monitors, this solution is completely feasible and actually saves area and power consumption.

Specifically, we focus on sensors whose analog varying signal is a pulse width or a ring-oscillator frequency, i.e. the digitalization part is a time-to-digital converter. This kind of signals is very easy to deliver from different points of the chip to a certain spot where the digitalization is performed. There is a certain dependency of the delay of the transmission lines on some of the factors that are to be measured, such as the temperature, the aging, etc. However, this variability is small enough to not affect the sensibility of the conversion, and it will be covered by our noise budget.

A whole generation of temperature sensors based upon time-to-digital converters have appeared in the last few years imposing a new paradigm because of their reduced power consumption and area. The common characteristics of this kind of sensors are a sensing part that produces a pulse with a varying duration as a function of the temperature and a digitalization part that normally includes a counter that measures and quantizes the pulse duration. For example, [12] employs a delay line with several gates to generate a temperature-dependent pulse. Also remarkable, the sensors in [5] make use of the leakage current thermal dependencies to produce the pulse and are characterized by a very small power consumption. Additionally, the use of thermal dependencies of interconnections to generate varying pulses was proposed in [13].

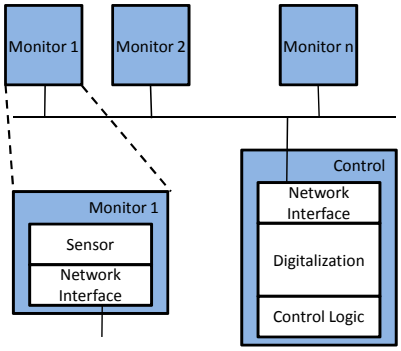


Fig. 4. Network architecture for the digitalization resource sharing.

Note, that although some works provide a signal with a varying frequency at the output of the sensor —such as a ring oscillator— this signal can easily be converted into a varying pulse by means of a counter, fed by this signal, that counts up to a fixed number.

Aiming at this kind of monitors that produce a pulse with varying durations, we propose two different schemes for the sharing of the digitalization resources. The first scheme, shown in figure 4, proposes that the digitalization part —i.e. the counter— is shared by a number of sensors and is placed at a certain control block separate from the sensors. The sensors are connected to the control block through a network and, with a time-multiplexing mechanism, send the pulse each at a time, so that when the time slot of one sensor is finished, the counter starts a new count. In this case, the counter might use different clock frequencies for each sensor depending on the type of sensor and their sensibility parameters, stored during the calibration phase. Synchronization is very important since the sensors must start to feed their pulse when their time slot starts.

The number of sensors that can share a counter is limited by the sampling frequency of the monitors and the duration of their pulses. The relationship of these parameters is expressed by:

$$f_s = \frac{1}{\sum_{i=1}^{n} \Delta T_i} \tag{2}$$

where f_s is the sampling frequency, n is the number of sensors connected to the counter and $\Delta T_i = max\{T_i\} - min\{T_i\}$ is the difference between the maximum and the minimum pulse produced by the sensing part of monitor i. An example of the functioning of this scheme is shown in figure 5.

In the second proposal, we impose a new restriction which is that all the sensors that connect to a counter must employ the same quantizing frequency for their time-to-digital conversion.

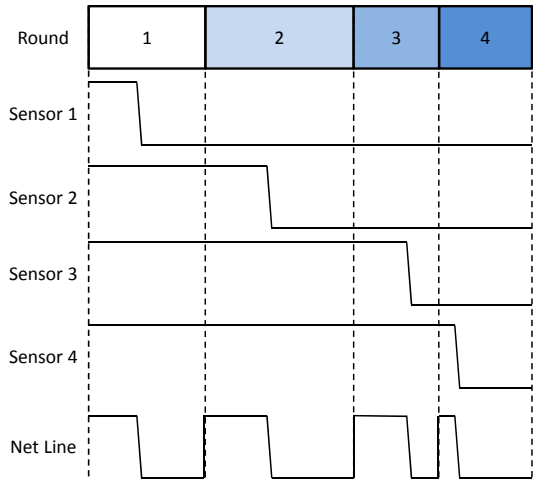


Fig. 5. Functioning example of the first proposal of digitalization resource sharing.

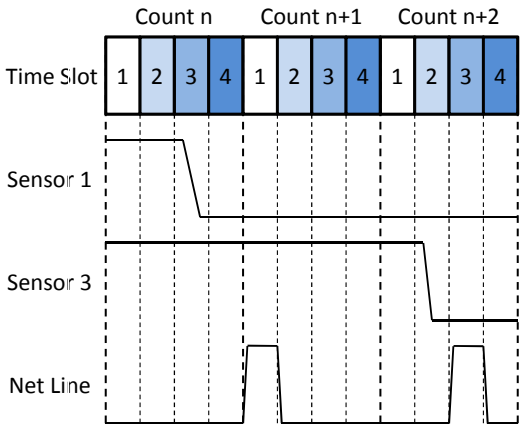


Fig. 6. Functioning example of the second proposal of digitalization resource sharing.

This is not a strong limitation since normally all the sensors are of the same kind, have the same layout, and thus suffer from the same sensibility issues. With this restriction, it is no longer necessary that the counter performs different counts for each sensor, instead, a single count can serve for all. More precisely, all the sensors connected to a counter start their pulse at the same time and the count starts at that moment; whenever a pulse from a sensor finishes, the current count is registered and associated with that particular sensor. In this way, an important power saving is produced compared with the other versions in which n counts were executed.

The scheme requires that all the monitors are able to communicate their pulse-end at any counting cycle, and more importantly that the control discriminates which one sent the signal. This is achieved by the division of each counting cycle into n slots, being n the number of monitors connected to a single counter. Each monitor is assigned a slot and whenever their sensing part asserts a pulse-end, the monitor-to-network interface sends a pulse to the network in their next slot. Note that this, at most, produces an error of a counting period which compares to the one produced by the standard counter-based time-to-digital conversion. This scheme is depicted in figure 6.

The maximum number of monitors that can be connected to a single monitor is no longer limited by the sampling frequency, in this case it is the clock frequency of the system, f_{clock} what restricts the minimum time slot size, and therefore the number of words that can be sent. Particularly, the number of monitors employing the same counter is bounded by

$$n < \frac{\Delta T f_{clock}}{2^q} \tag{3}$$

where q is the number of bits of the monitor signal; and $\Delta T = max\{T\} - min\{T\}$ is the difference between the minimum and the maximum pulse produced by the sensing part.

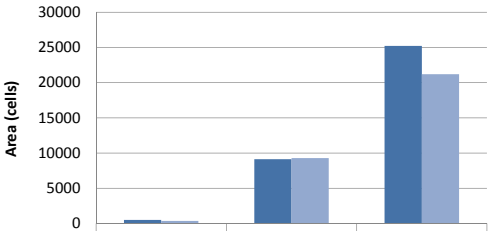
V. Comparison with previous works and discussion

To demonstrate the advantages of our network architecture we have implemented a 32-monitor network of our second proposal and another 32-monitor network of the traditional boundary-scan like approach. The implementations have the minimum circuitry that fulfills the protocols involved in each network. We synthesized the designs targeting a 90-nm standard cell library from UMC and the numeric results come from the synthesis simulation under typical conditions.

The details of the implementation are as follows. The sensing part of the monitors simulates a temperature sensor and provides a PWM signal that needs an 8-bit quantization. All the monitors need the same quantizing frequency and, in the case of the boundary-scan like network, an extra control line distributes this signal. This decision is questionable, since we could have implemented a module in each monitor that produced this frequency from the clock tree, however it would just increase their area and power consumption. The control module simply stores the data from the 32 monitors and selects the biggest, since we consider these as the indispensable functions for it. We fixed the working frequency at 10 MHz and used a sampling period of 1ms.

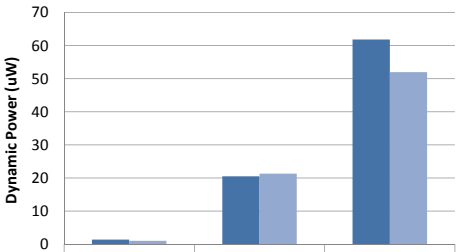
Figures 7 and 8 summarize the synthesis results of both architectures. As shown, we achieve an area improvement of 25% in each monitor due to the lack of digitalization modules, in the complete network, we get an improvement of 16%. Concerning the power consumption, we achieve a significant reduction of 16% in the whole network. As the network adds more monitors, the improvement approaches a maximum of a 25% since the monitors represent a higher and higher portion of the network.

Note that apart from the energy saving due to the union of all digitalization processes into a single one, there is also an important dynamic power reduction caused by the smaller number of loads and unloads in the shared net line. From the analysis of the boundary-scan-like implementation, we get that the number of transitions in each of the interconnection wires



	Single Monitor	Control	Net
Boundary-Scan Like	504	9140	25227
Digital Resource Sharing	374	9294	21207

Fig. 7. Area results and comparison.



	Single Monitor	Control	Net
Boundary-Scan Like	1,39	20,50	61,85
Digital Resource Sharing	1,02	21,33	52,00

Fig. 8. Power results and comparison.

of an n-monitor network is upbounded by $\sum_{i=1}^{n} q_i$, being q_i the number of bits of the ith monitor. If we now turn to the case of our network architecture, the number of transitions in the interconnection line is upbounded by $2n$, twice the number of monitors because each monitor just transmits a pulse. The savings in dynamic power consumption on the interconnection line are significant; in fact, for very distributed networks with long wires, this signaling model could be extended to any kind of smart sensor, including those that do not realize a time-to-digital conversion. The idea is to turn any digital word into a width-varying pulse by means of a counter. The power overhead caused by the counter compares to that of the boundary-scan like protocol and is compensated by the energy savings in the transmission line.

Another important feature of our architecture is that the calibration process is performed employing a single digitalization block, and that the write-back information of each sensor is kept at the control block rather than at each of the monitors. This simplifies the calibration stage and the linearization process since all the required logic is instantiated just once and it is easier to control.

VI. Conclusions

Thermal issues in the late CMOS era suppose serious challenges for electronic engineers. Hotspots, NBTI and TDDB appear as first class design issues that require strict control policies. In this context, DTM —which acts at run-time to optimize the IC temperature— has proven as the best solution. DTM policies require a thermal map of the chip that is provided by a set of temperature sensors. During the last few years the research community has made a big effort to provide a number of temperature sensors targeting the requirements imposed by DTM policies. However, little attention has been paid to the network that must connect all these sensors and deliver their data to a central control.

In this work we started analyzing the requirements and ideal features of this kind of networks. Then, we proposed a three-level hierarchy model for them that matches the historic development of thermal control needs. Finally, focusing on third level of the hierarchy, which is the densest, we introduced a new network architecture, based on the concept of digitalization resource sharing. The time-to-digital conversion of all the monitors is realized at the same control module and at the same time. When compared to the traditional boundary-scan like network, our architecture achieves a 16% saving in area and power consumption. Furthermore it reduces importantly the activity on the network, it is easily scalable and simplifies the calibration process.

References

[1] S. Feng, S. Gupta, A. Ansari, and S. Mahlke, "Maestro: Orchestrating Lifetime Reliability in Chip Multiprocessors," in *High Performance Embedded Architectures and Compilers: 5th International Conference, Hipeac 2010, Pisa, Italy, January 25-27, 2010, Proceedings*. Springer-Verlag New York Inc, 2010, pp. 186–200.

[2] D. Sylvester, D. Blaauw, and E. Karl, "Elastic: An adaptive self-healing architecture for unpredictable silicon," *Design Test of Computers, IEEE*, vol. 23, no. 6, pp. 484 –490, june 2006.

[3] S. Memik, R. Mukherjee, M. Ni, and J. Long, "Optimizing Thermal Sensor Allocation for Microprocessors," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 3, pp. 516–527, 2008.

[4] J. Long, S. Memik, G. Memik, and R. Mukherjee, "Thermal monitoring mechanisms for chip multiprocessors," *ACM Transactions on Architecture and Code Optimization*, vol. 5, no. 2, 2008.

[5] P. Ituero, J. Ayala, and M. Lopez-Vallejo, "A Nanowatt Smart Temperature Sensor for Dynamic Thermal Management," *IEEE Sensors Journal*, vol. 8, no. 12, pp. 2036–2043, 2008.

[6] A. Drake, *Adaptive Techniques for Dynamic Processor Optimization*. Springer, 2008, ch. 7 Sensors for Critical Path Monitoring, pp. 145–174.

[7] V. Szekely, M. Rencz, and B. Courtois, "Tracing the thermal behavior of ics," *IEEE Design & Test of Computers*, vol. 15, no. 2, pp. 14–21, 1998.

[8] K. Vaddina, E. Nigussie, P. Liljeberg, and J. Plosila, "Self-timed thermal sensing and monitoring of multicore systems," in *Proceedings of the 2009 12th International Symposium on Design and Diagnostics of Electronic Circuits&Systems-Volume 00*. IEEE Computer Society, 2009, pp. 246–251.

[9] *Intel Pentium M Processor with 2-MB L2 Cache and 400 MHz Front Side Bus*, http://download.intel.com/design/mobile/datashts/30218908.pdf, May 2009.

[10] E. Rotem, J. Hermerding, A. Cohen, and H. Cain, "Temperature measurement in the Intel® CoreTM Duo Processor," 2006.

[11] A. Naveh, E. Rotem, A. Mendelson, S. Gochman, R. Chabukswar, K. Krishnan, and A. Kumar, "Power and Thermal Management in the Intel® Core Duo," *Intel Technology Journal Vol. 10# 2, 2006*.

[12] P. Chen, C.-C. Chen, C.-C. Tsa, and W.-F. Lu, "A Time-to-Digital-Converter-Based CMOS Smart Temperature Sensor," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 8, pp. 1642–1648, August 2005.

[13] B. Datta and W. Burleson, "Low power on-chip thermal sensors based on wires," in *Very Large Scale Integration, 2007. VLSI-SoC 2007. IFIP International Conference on*, 2007, pp. 258–263.