Proceedings of the 10th WSEAS International Conference on COMPUTERS, Vouliagmeni, Athens, Greece, July 13-15, 2006 (pp1220-1223)

Microprocessor Thermal Benchmark

MARIUS MARCU
Department of Computer Science, "Politehnica" University of Timisoara, Faculty of Automation and Computer Science and Engineering, 2 V. Parvan, Timisoara, Romania, Phone: (40) 256-403263, Fax: (40) 256-403214, E-

MIRCEA VLADUTIU
Department of Computer Science, "Politehnica" University of Timisoara, Faculty of Automation and Computer Science and Engineering, 2 V. Parvan, Timisoara, Romania, Phone: (40) 256-403261, Fax: (40) 256-403214, E-

HORATIU MOLDOVAN
Department of Computer Science, "Politehnica" University of Timisoara, Faculty of Automation and Computer Science and Engineering, 2 V. Parvan, Timisoara, Romania, Phone: (40) 256-201279, Fax: (40) 256-201279, E-

Abstract: - Power consumption and heat dissipation become key elements in the field of high-end integrated circuits, especially those used in mobile and high-speed applications, due to their increase of transistor count and clock frequencies. Dynamic thermal management strategies have been proposed and implemented in order to mitigate heat dissipation. However, there is a lack of a tool that can be used to evaluate DTM strategies and thermal response of real life systems. Therefore, in this paper we introduce and define for the first time the concept of thermal benchmark software as a software concept (and tool) for run-time system level thermal characterization.

Key-Words: - thermal benchmark, thermal characterization, thermal sensor

1. Introduction

Power consumption of modern processors is continuing increasing, as a consequence of the increase in integration area, number of integrated units and working frequencies. Because of the increase in power consumption, new problems arise, which are related to the limitation of the temperature at the semiconductors' level, in order to avoid the physical damages of the circuits [1, 2]. This is the reason why temperature is used often during the different phases of developing the component parts of the computing systems, the most important of them being thermal design, thermal testing and thermal management [2].

The most important consequences of increasing the circuits' operating temperature are [2, 3]:

- decreased speed – the speed decreases with 0.15% for each 1 ^oC increment of temperature. Increased temperature affects the commutation time of logic gates;
- decreased reliability – the life span is diminished by 50% given a 10-15 ^oC increase of temperature. Increased temperature is affecting and degrading the silicon structure, therefore leading to an early failure of the circuit;
- increased leakage currents - I_{OFF} rises with temperature. There is a great possibility for the "thermal runaway" process to occur because of residual currents' increase, that further implies warming of the circuit, which triggers again the growth of the current and so on;
- increased probability of software errors.

Therefore, all integrated circuits are designed to operate reliably within a defined temperature range. Outside of this range, there is no assurance that the integrated circuits will continue to function correctly [2]. Hence, in the context of increasing in temperature,

Proceedings of the 10th WSEAS International Conference on COMPUTERS, Vouliagmeni, Athens, Greece, July 13-15, 2006 (pp1220-1223)

thermal management methods have to be applied. The ultimate goal of these techniques is to keep the processor at or below its maximum operating temperature regardless of running application or environment temperature.

Proper thermal management depends on two major elements:

- **thermal packaging** including heatsink properly mounted to the processor and effective fans directing airflow through the system chassis [2,3,4].
- **dynamic thermal management** (DTM) strategies. To reduce packaging cost without unnecessarily limiting performance, the package is designed for the worst typical application and any application that dissipate more heat should activate an alternative, run-time thermal management technique [3,4].

As a conclusion for thermal management, DTM schemes are designed as solutions to deal with the worst-case applications where the thermal package deals with the average or typical applications.

This article does not intend to presented DTM techniques, good overview of them is part of articles [4] and [5]. What we want to remark is that DTM strategies are mainly evaluated by simulation or using performance benchmarks (usually SPEC CPU2000 [7]). Therefore we try to fill this gap by proposing a new kind of benchmark, we called **thermal benchmark**. Thermal benchmark concept is not new but is not very often used in the context of thermal management. The only one context we found for thermal benchmark is the work of Szekely et. all [6], regarding a thermal benchmark circuit, designed for thermal characterization of circuit level management. But there is no reference, as we know at this moment, about thermal benchmark software, therefore we intend to introduce and define such one here.

Thermal benchmark applications are intended to be used to evaluate and/or calibrate DTM solutions, thermal hardware sizing related to applications the system is expected to support, on-line thermal testing and monitoring, system level thermal control.

The paper is organized as follows. In the next section we define the thermal benchmark concept. Section 3 of this paper describes our thermal benchmark software implementation. Experimental results and conclusions are presented in Section 5 and Section 6 respectively.

2. Thermal benchmark definition

A computer benchmark is typically a computer program that performs a strictly defined set of operations (a workload) and returns some form of result (a metric) describing how the tested computer performed. Computer benchmark metrics usually measure speed or throughput. Running the same computer benchmark on multiple computers allows a comparison to be made [7].

We propose to extend the concept of benchmarking with a new metric: the temperature, and we name it thermal benchmark.

Definition 1 - Thermal benchmark is defined as a software program that describes the thermal behavior of the system with respect to certain stimulus (workload).

A thermal benchmark must by able to distinguish the way a hardware device temperature is increasing with workload and the way it’s temperature decreases when the workload is finished. Therefore, a thermal benchmark is composed by three components (Fig. 1):

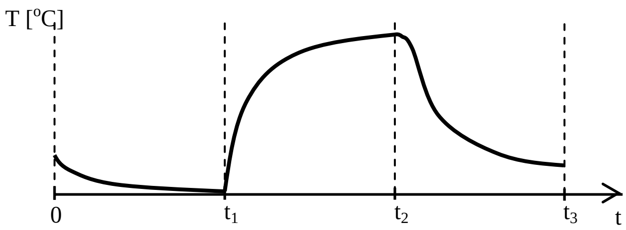

Fig. 1 Thermal benchmark definition

- the first range [0-t_1), is intended for idle mode steady temperature. In this step, the component does not execute anything, but the power saving mechanisms are prevented to occur.
- the second range [t_1-t_2) represents the warming phase, when a certain workload is executed. SPEC CPU2000 or any type of other applications can be executed as workload.
- the last range [t_2-t_3) represents the cooling phase intended for the component to reach again the idle state temperature. In this step, the component does not execute anything, but the power saving mechanisms are prevented to occur.

A thermal benchmark in not intended only for the microprocessor but it also can be applied for every hardware device with built-in thermal monitoring capability (such as hard disk).

3. Implementation

Run-time thermal management requires real-time temperature sensing. The majority of microprocessors produced in last years include at least one built-in thermal sensor to aid in thermal management of servers or workstation systems. This thermal sensor is connected to a thermal diode on the processor core. The thermal diode temperature changes with the junction temperature very closely. Thus, the thermal sensor provides the earliest indication of thermal variation [10].

The thermal sensor is composed of control logic, SMBus interface logic, a precision analog-to-digital converter, and a precision current source. Analog-to-digital conversions happen continuously and the single byte result of the most recent conversion is stored in a register on the thermal sensor. This register can be read from the applications using the serial interface protocol of the SMBus [10]. SMBus is the System Management Bus defined by Intel Corporation in 1995 and is used in personal computers and servers for low-speed system management communications [9].

In a computer, there is a circuit, called hardware monitor, which implements the SMBus controller and the interface between applications and processor thermal sensor. The hardware monitor circuit can be used to monitor power supply voltages, temperatures, and fan speeds. Actual values for these inputs can be read at any time from device drivers or user level applications.

More detailed information about temperature sensors reading from applications running under Windows NT based operating systems was published by the same author in [8].

We implemented the processor thermal benchmark in Visual C++ 6.0 and it was tested under Windows 2000 and Windows XP operating systems. The application implements in code the three phases of a thermal benchmark, presented in previous section. The t_1, t_2 and t_3 time values in Fig. 1 are parameters configurable in the application.

4. Test cases and experimental results

In order to validate our proposed thermal benchmark software solution we elaborate a set of test cases, the most important of them being presented in this section. The test were run a large number of hours (summing almost one month) on 6 different hardware systems:

- AMD Athlon 1200 MHz, 256 MB RAM, chipset and thermal monitor VIA686 (3 systems)
- AMD Duron 800 MHz, 256 MB RAM, chipset and thermal monitor VIA686
- Intel Pentium II 400 MHz, 128 MB RAM, chipset Intel 82801AA/ICH, thermal monitor ADM 1025
- Intel Pentium IV 2800 MHz, 512 MB RAM, chipset Intel 82801EB/ICH5, thermal monitor ADM 1027

4.1 Thermal benchmark stability

The first test was intended to show that thermal benchmark provides the same results on the same system when the same workload is applied. Fig. 2 shows two thermal signatures for the same system.

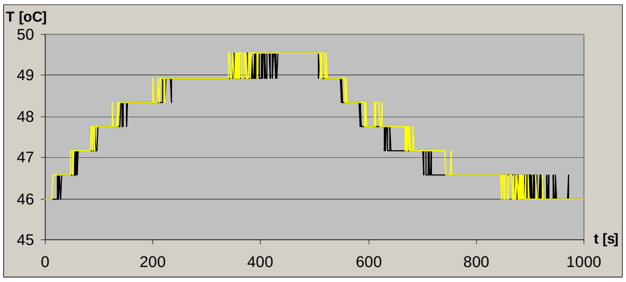

Fig.2 Thermal measurements

Running the benchmark on the same system a large number of times and averaging the measured temperatures we obtains the standard thermal signature of a certain processor for the selected workload.

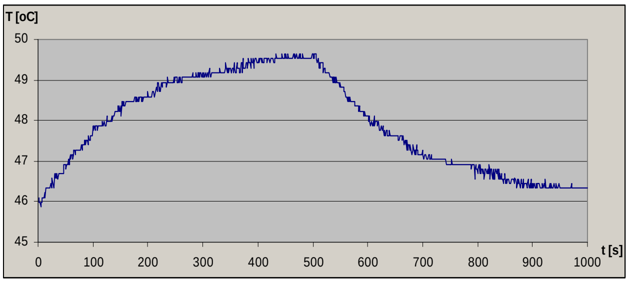

Fig. 3 Thermal standard signature for a processor

4.2 Thermal benchmark similarity

The second test was intended to distinguish the thermal signatures for the same type or similar types of processors placed in different systems (case, fans). In Fig. 4 can be observed that thermal response of similar processors is the same for the same workload, but is influenced by cooling components used inside the case.

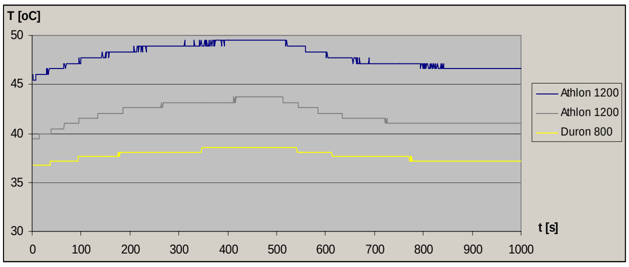

Fig. 4. Thermal benchmark results for similar processors

4.3 Thermal benchmark

The third test was selected to distinguish between thermal signature of different processors. Fig. 5 shows big differences in thermal responses for Pentium II, Pentium IV and Athlon for the same workload.

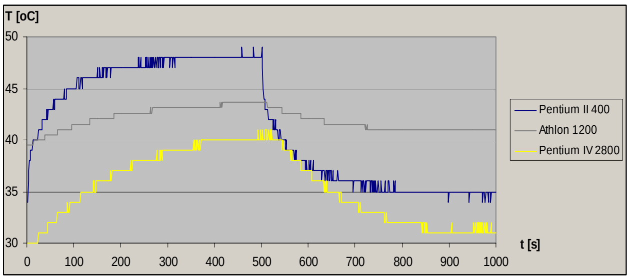

Fig. 5. Thermal benchmark results for different processors

4.4 Thermal benchmark workloads

The forth test presents the thermal response of the same processor when different workload types were applied: integer, memory and floating point operations.

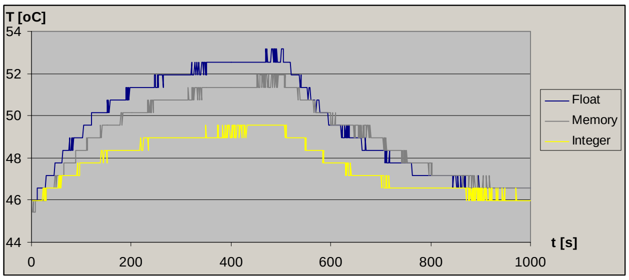

Fig. 6. Thermal benchmark for different workloads

5. Conclusions

This paper proposed and introduced a new type of benchmark: the thermal benchmark. Thermal benchmarks are software applications intended for run-time system components thermal characterization, with possible applicability in evaluation and calibration of DTM solutions, thermal hardware sizing related to applications, on-line thermal testing and monitoring, system level thermal control. The new thermal benchmark concept was validated by a set of test cases presented in this paper.

Future work on thermal benchmarking will be directed to battery powered systems and dual-core microprocessors.

References:

[1] V. Szekely, M. Rencz, and B. Courtois, "Tracing the Thermal Behaviour of ICs", *IEEE Design&Test of Computers*, Vol. 15, No. 2, April-June 1998, pp. 14-21.

[2] S.H. Gunther, F. Binns, D.M. Carmean, and J.C. Hall, "Managing the Impact of Increasing Microprocessor Power Consumption", *Intel Technology Journal*, Q1, 2001.

[3] K. Skadron, M.R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-aware Microarchitecture", *Proceedings of the 30th ACM/IEEE International Symposium on Computer Architecture*, San Diego, CA, June 2003, pp. 2-13.

[4] Intel Pentium 4 Processor Thermal Management, http://support.intel.com/support/processors/pentium4/sb/CS-007999.htm

[5] D. Brooks and M. Martonosi, "Dynamic thermal management for high-performance microprocessors", *7th International Symposium on High-Performance Computer Architecture (HPCA'01)*, Jan. 2001, pp. 171-182.

[6] V. Szekely, C. Marta, M. Rencz, G. Vegh, Z. Benedek, and S. Torok, "A Thermal Benchmark Chip: Design and Application", *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, Part A. Vol. 21. No. 3, Sept., 1998, pp. 399-405.

[7] Standard Performance Evaluation Corporation, SPEC CPU2000 Benchmarks. http://www.specbench.org/osg/cpu2000.

[8] M. Marcu, H. Moldovan, V. Tomescu, M. Zilahi, M. Ionas, "Windows XP Issues for Real-Time Temperature Monitoring", *22nd IEEE Instrumentation and Measurement Technology Conference IMTC2005*, Ottawa, Canada, May 2005.

[9] System Management Bus (SMBus) Specification Version 2.0, Aug. 2000.

[10] Pentium II Xeon Processor SMBus Thermal Reference Guidelines, Application Note AP-831, June 1998.