A Dynamic Reliability Management Framework for Dark Silicon

Taeyoung Kim*, Xin Huang†, Valeriy Sukharev‡, Sheldon X.-D. Tan†
* Department of Computer Science and Engineering, University of California, Riverside, CA 92521
† Department of Electrical and Computer Engineering, University of California, Riverside, CA 92521
‡ Mentor Graphics Corporation, Fremont, CA 94538

***Abstract*—In this paper, we introduce the dynamic reliability management (DRM) framework for dark silicon, which is considered in one of the emerging computing platforms for the future many-core systems due to the diminishing of Dennard's scaling. We focus on the electromigration-induced reliability problem as it is the dominant failure effect for on-chip interconnects. To make our framework more realistic, our approach is based on newly proposed physics-based electromigration (EM) reliability model, interval core-based microarchitecture model, and X86-based power model. We also apply HotSpot thermal model for the temperature estimation. We consider thermal design power (TDP) as the power constraint for a dark silicon chip. For DRM, we use the recently proposed resource-based reliability model in which each core of dark silicon chip can spend the time-to-failure (TTF) resources at different rates specified by the temperature and its power consumption. We employ both dynamic voltage and frequency scaling (DVFS) and task migration as the control knobs. A large class of multi–threaded applications is used as the benchmark to validate our reliability-aware dark silicon framework and the proposed DRM method. Experimental results on a 64-core dark silicon chip show that the DRM on the proposed framework can effectively manage and optimize the lifetime of the chips under the given power budget on a dark silicon chip.**

I. Introduction

Technology scaling has led to the continuous integration of devices, and future many-core chips will have more cores integrated. However due to the diminishing of Dennard's scaling [1], power density of chip starts to increase for current and future technology nodes, the consequence is the emerging of so-called dark silicon many-core processors as the chip can only be powered at a certain percentage of cores due to the power and temperature limitations. Recently, the architecture researcher has begun focusing on a many-core processor such as 100-core and 1000-core dark silicon chips on the die, such many-core systems pose new challenges and opportunities for power/thermal and reliability management of those chips [2].

On the other hand, reliability is becoming a limiting constraint in high-performance nanometer VLSI chip designs due to the high failure rates in deep submicron and nanoscale devices. It was expected that the future chips will show sign of reliability-induced age much faster than the previous generations. Among of many reliability effects, electromigration (EM)-induced reliability has become a major design constraint due to aggressive transistor scaling and increasing power density. For dark silicon, the reliability can become worse as cores will experience more thermal cycles during the on-off operations. Many core also may operate in the very low or even near threshold voltage region, which also hurts the soft-error induced reliability. For the EM effects, however, dark silicon can provide one more knob (turn on or off for a core) to manage the EM-induced lifetime of the chip, which will be explored in this work.

This work is supported in part by NSF grant under No. CCF-1017090, in part by NSF Grant under No. CCF-1255899, in part by Semiconductor Research Corporation (SRC) grant under No. 2013-TJ-2417.

Existing studies mainly focus on many-core or dark silicon architecture, such as core organization, topology, and optimal number of cores, and workload management, such as task allocation, migration, and scheduling [2]–[5]. However, most of those existing works focus on performance scaling and energy efficiency for dark silicon chips. Recently, reliability management with dark silicon effect on many-core scaling were proposed [6]–[9], All of these works considered conventional reliability models, which will not be accurate for specific failure mechanism.

In this paper, we propose the dynamic reliability management (DRM) framework for dark silicon considering the electromigration (EM)-induced reliability. We focus on the electromigration induced reliability problem, which is the dominant failure effect for interconnects. To make our framework more realistic, our approach is based on newly proposed physics-based electromigration (EM) reliability model, interval core-based microarchitecture model, and X86-based power model. We also apply the HotSpot thermal model for the temperature estimation. We consider thermal design power (TDP) as a power constraint for a dark silicon chip. For DRM, we use the recently proposed resource-based reliability model in which each core of dark silicon chip can spend the time-to-failure (TTF) resources at different rates specified by the temperature and its power consumption. The reliability management uses two variance reduction techniques. First, we use the dynamic voltage and frequency scaling (DVFS) techniques for low power mode, which can improve the reliability without adding design margins or spare components. Second, we use task migration, which also can improve the reliability of the multi-core platform. We employ both dynamic voltage and frequency scaling (DVFS) and task migration as the control knobs. A large class of multi–threaded applications is used for the benchmark to validate our framework and DRM. Experimental results on a 64-core dark silicon chip show that the DRM on the proposed framework can effectively manage and optimize the lifetime of the chips under the given power budget on a dark silicon chip.

II. Dynamic reliability management framework for dark silicon

In this section, we will describe our dynamic reliability management framework for dark silicon as shown in Fig. 1. We implement a full simulation-based framework for dark silicon chip and develop a new reliability management technique. We first describe the major component models of the framework such as microarchitecture, power estimation, thermal and reliability models.

A. dark silicon microarchitecture, power, and thermal models

Our proposed framework uses Sniper as an microarchitecture model, which is an accurate and fast application-level interval-based microarchitecture simulation [10]. The interval simulation is a recently proposed multi-/many-core simulation framework at a

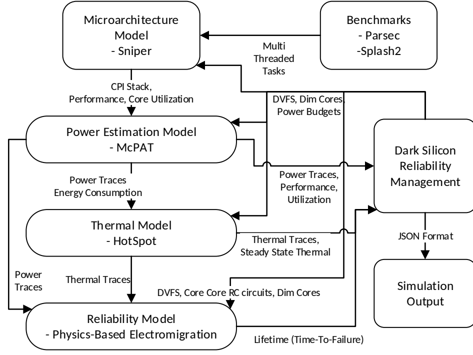

Fig. 1. Reliability-Aware dark silicon framework

higher level of abstraction which is faster than cycle-accurate full-system simulation. The interval simulation uses mechanistic analytical model, which is constructed from the mechanism of a superscalar processor core. The cycle-accurate full-system simulator, such as gem5 (full-system mode) [11], GEMS [12], MARSSx86 [13] and SimFlex [14] can run both application and operating system (OS). These frameworks have the merit of having an accurate evaluation of I/O activities and OS extensive kernel function. However, these full-time simulations are extremely slow and not very suitable for our framework because they rely on the existing OS systems, which currently do not support many-core and dark silicon architectures in their simulators [15]. Thus, to support dark silicon and may-core processor, we choose application-level Sniper simulator. This Sniper interval-based model is accurately matching well with the Intel x86 multi-core architecture [10]. For our dark silicon framework, PARSEC [16] and Splash2 [17] benchmarks are used. Both benchmarks are recently released multithreaded benchmarks, which provide an up-to-date collection of modern workloads for multi-/many-core systems. We use both workloads to evaluate our proposed framework and algorithm in Section III.

For the power estimation, we use McPAT (Multicore Power, Area and Timing), which was recently proposed full integration modeling framework. McPAT can provide dynamic and static, even short-circuit power dissipation and provides multi-threaded and multi-core processor models. The timing and area models are derived from CACTI [18]. The dynamic power model is close to Wattch [19]. But it supports short-circuit and leakage power models. At the each step of performance measurement in Sniper, McPAT can estimate the power and energy consumption. For the thermal model, we use HotSpot to accurately characterize the thermal traces from the given multithreaded task run in each core [20]. To enable dark silicon feature, the floor plan and power trace are dynamically controlled by the dark silicon reliability management module in Fig. 1

Our new proposed framework uses the dynamic voltage and frequency scaling (DVFS) techniques, which can improve the reliability without adding design margins or spare components. Moreover, we use coordinated multi-threaded task migration, which also can improve the reliability of the multi-core platform. Dark silicon reliability management module has these features to control core voltage and frequency settings and allocate the task into each core.

As shown Fig. 1, once the CPI performance stack and power/energy traces are achieved in the microarchitecture model with power model, the thermal model can generate thermal trace for given task run. With each core's power trace, thermal trace, core voltage, core frequency, and active cores, we can perform EM reliability analysis and the system-level assessment for processor lifetime based on the reliability models. Fig. 2(a) and 2(b) show the results from the proposed framework, which are the power traces samples and thermal measurement in a 64-core dark silicon chip. There are 64 cores running at the normal DVFS setting (2.0Ghz, 1.2V) and 8 multi-threaded tasks (2x cholesky, 2x radix, 2x raytrace, 2x volrend). Each task have 8 multithreads, so 16 threads are running on a 64-core dark silicon chip. Fig. 1 only shows the core area. Power budgeting is not applied here.

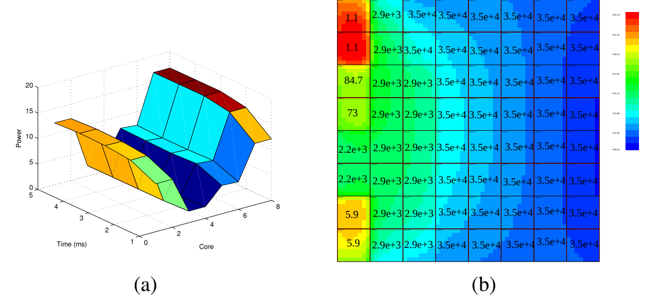

Fig. 2. (a) SPLASH2 benchmark 8 multithreaded tasks power traces (b) Thermal (color:degree) and EM lifetime (number:yrs) analysis on 64 cores

B. System-Level EM reliability model for dark silicon

An existing EM models including the new physics-based model can only take a constant temperature. Previous study shows that whole system time-to-failure (TTF) or lifetime under different temperature can be approximated by [21]: For the dark silicon, here we add core selection variable $a_{i,k}$ to describe if the core is active or dimmed. Dark silicon EM reliability system-level model can be expressed as follows

$$TTF(i) = \frac{1}{(\sum_{k=1}^{n}(a_{i,k}\Delta t_{i,k}\frac{1}{TTF_{i,k}}))/T} \quad (1)$$

where $TTF_{R,k}$ is the actual TTF under the k-th power and temperature settings for Δt_k period, assuming each core works through n different power and temperature settings and $T = \sum_{k=1}^{n} \Delta t_{i,k}$. $a_{i,k}$ is core selection variable, which is zero when i-th core is dimmed at the k-th execution cycle time as dark silicon is enabled. Each $TTF_{i,k}$ will be computed based on the recently proposed physics-based EM model and assessment techniques [22].

To explain a system-level reliability on a dark silicon-enabled many-core processor, we define *performability* as the ratio of number of non-failure cores over total number of cores [6]. The performability ratio is 1 when all cores are not failed. The many-core processor lifetime can be defined as the time taken to reach the certain performability threshold. It depends on the design and process.

III. Dynamic Reliability Management for Dark Silicon

We use the task migration method to balance the reliability resources for dark silicon chips, which is different from the conventional multi-core task migration method that targets at improving on-chip temperature profile. Based on the (1), we can treat the lifetime of the processor specified by TTF as a resource that could be consumed as the core works. We first define the specified TTF as a nominal value, denoted as TTF_N, which is the intended or required life of the core under a typical temperature and power traces in a core.

Depending on different workload settings, the consumption rate could be either higher or lower than its nominal rate, and we define *consumption rate* for workload k as

$$cr_k = \frac{TTF_N}{TTF_k} \quad (2)$$

in which the lifetime in real case (TTF_k under the kth workload) could be estimated by (1). In the nominal case, the core is working under its specified temperature and power setting, and it has lifetime given by TTF_N. Hence, the amount of lifetime consumed by the core in each second is 1 EM second, that is to say, the nominal average consumption rate is $cr_N = 1$. According to the definition of TTF consumption rate defined by (2), if $cr_k > cr_N$ persistently, it will introduce excessive consumption of TTF, which would possibly lead to early failure of the core if no compensation is made. We define *TTF resource slack* as the accumulative TTF consumption difference between real case and nominal case over all different task execution periods, which is calculated through

$$S_d(i) = \sum_{k=1} (cr_N - cr_k)\Delta p_k \tag{3}$$

$$= \sum_{k=1} (\frac{TTF_k - TTF_N}{TTF_k})\Delta p_k \tag{4}$$

$$= \int \frac{TTF(t) - TTF_N}{TTF(t)} d(t) \tag{5}$$

where cr_k is the average consumption rate during the k-th execution cycle at i-th core, cr_N is the nominal average consumption rate, In the continuous time, $\Delta p_k = ECT$ is defined one cycle time. $TTF(t)$ is TTF at time t based on the temperature and current densities at t, $a_{k,i}$ is core selection variable, which shows i-th core is selected at the k-th execution cycle time as dark silicon is enabled.

Given (3), we consider the following scenarios:

- If $S_d = 0$, it indicates the overall consumption of the core would lead to its intended TTF. It is easy to verify $lifetime = TTF_N$ by using (1) in this case.
- If $S_d < 0$, it indicates that the lifetime is excessively consumed for the past execution periods, and it requires compensations in future to avoid early failure.
- If $S_d > 0$, it indicates that the lifetime is consumed less than its nominal rate for the past execution periods, and it allows increased consumptions in future without causing early failure.

In a dark silicon system, for each core i, we could calculate TTF resource slack for core i at the end of each task execution cycle, and denote it as $S_d(i)$. We could also characterize the average power of the tasks in the coming execution cycle for each core. Assuming that the dark silicon processor has N cores, and the average power of the tasks on each core are denoted as P_1,P_2,...,P_N, and the TTF resource slack for each core are denoted as $S_d(1)$, $S_d(2)$,...,$S_d(N)$.

To balance the EM-reliability of all the cores, one simple solution is to use a greedy based algorithm which is to sort out the order of power consumptions and that of the TTF resource slacks, and assign the highest power to the core with highest value of S_d, and assign the second highest power to the core with the second highest value of S_d, and so on. Our task migration can be allocated with active core selection switch for dark silicon. In this way, the TTF consumption of different cores could be balanced, which means that all the cores will be targeting at the similar length of lifetime, avoiding early failure of some cores due to continuously heavy load assignment.

However, the task migration would not always be able to compensate all the excessively consumed TTF if all the cores are loaded with heavy tasks, thus, we use DVFS technique to enable low power mode so that it can compensate the overly consumed TTF, thus, the core could maintain its intended lifetime.

IV. Numerical Result and Discussions

A. Experimental setup

The proposed new dynamic reliability management framework for dark silicon has been implemented in Python 2.7.9 with numerical libraries, such as Numpy 1.9.2 and Scipy 0.15.1. We modified the architecture simulator (Sniper 6.1), power model simulator (McPAT 1.0.32), and thermal simulator (HotSpot 5.02) to estimate reliability-aware performance and lifetime task models on top of new physics-based EM models with the dark silicon chip configuration. A large class of multi–threaded applications (PARSEC and SPLASH-2) is used for the benchmark to validate our proposed ideas. Our framework support five P-state but in this experiment, we choose two P-state for DVFS, one is full power mode (2.0GHz, 1.2V setting) and another is low power mode (1.0Ghz, 0.9V setting).

B. Dark silicon power budget and performance analysis

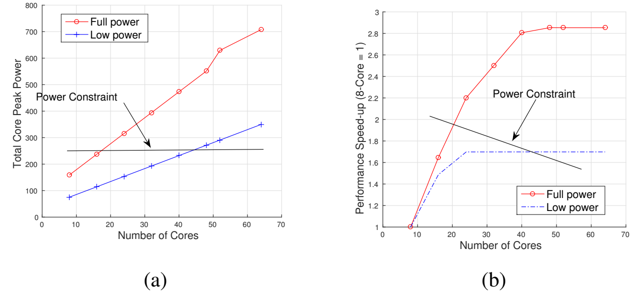

Fig. 3. (a) Active core total peak power and power budget for full power and low power mode (b) Performance analysis on number of active core

As seen in Fig. 3, to evaluate our dark silicon framework, the power consumption and performance results are shown based on the number of active cores (8,16,24,32,40,48,52, and 64). The dark silicon power constraint is obtained from the certain power to ensure that all the cores' executions do not exceed the critical temperature based on the realistic Intel Xeon Processor [23] (our framework power model is based on x86 architecture). As a result, we find a suitable TDP value (250W) for our 64-core dark silicon chip, which can be changed by the design and process. Based on the given power constraint, our framework can choose the maximum number of active cores for full power mode (highest DVFS performance state), and low power mode (lowest DVFS performance state) as shown in Fig. 3(a). Based on our power constraint, we find out that 16 cores can be active for the full power mode, and 40 cores can be active for the low power mode.

C. EM-induced dynamic reliability management analysis

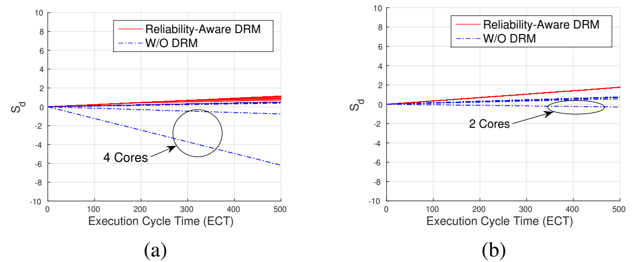

Fig. 4. Resource-based model for dynamic reliability management effect on (a) full power (b) low power

We then proceed to evaluate our dynamic reliability management (DRM) method on the dark silicon environment. Fig. 4 shows the comparison results with DRM and without DRM. In dark silicon, if we do not consider reliability, some of cores still can be excessively used and consumed, resulting in the decrease of performability. As we

can see from Fig. 4(a), at the full power level, four cores' lifetimes are excessively consumed. However, DRM can effectively manage to avoid early failure and maintain good performability. At the low power level, 2 cores' lifetimes are excessively consumed and our DRM can balance each core lifetime well.

Further, we compare thermal-aware DRM method with our EM-aware DRM for full and low power modes. Fig. 5 shows 2 cores at the full power mode (a) and 1 core for low power model (b) excessively consume the lifetime resources. But, the reliability-aware DRM lead much better results in both cases. As a result, the new approach can achieve higher performability for dark silicon.

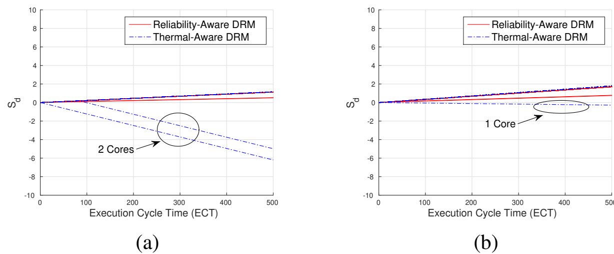

Fig. 5. Resource-based model for dynamic reliability management and dynamic thermal management on (a) full power (b) low power

V. Conclusion

In this paper, we propose the dynamic reliability management (DRM) framework for dark silicon considering the electromigration (EM)-induced reliability. We focus on the electromigration induced reliability problem, which is the dominant failure effects for interconnects. To make our framework more realistic, our approach is based on newly proposed physics-based electromigration (EM) reliability model, interval core-based microarchitecture model, and X86-based power model. We also apply the HotSpot thermal model for the temperature estimation. We consider thermal design power (TDP) as a power constraint for a dark silicon chip and use realistic TDP from Intel Xeon Processor. For DRM, we use the recently proposed resource-based reliability model in which each core of dark silicon chip can spend the time-to-failure (TTF) resources at different rates specified by the temperature and its power consumption. The reliability management uses two variance reduction techniques. First, we use the dynamic voltage and frequency scaling (DVFS) techniques for low power mode, which can improve the reliability without adding design margins or spare components. Second, we use task migration, which also can improve the reliability of the multi-core platform. We employ both dynamic voltage and frequency scaling (DVFS) and task migration as the control knobs. A large class of multi–threaded applications is used for the benchmark to validate our framework and DRM. The experimental results on a 64-core dark silicon chip show that the DRM on the proposed framework can effectively manage and optimize the lifetime of the chips under the given power budget on a dark silicon chip. As a result, our new approach can achieve higher performability for dark silicon.

References

[1] R. Dennard, F. Gaensslen, H. Yu, V. Rideout, E. Bassous, and A. LeBlanc, "Design of ion-implanted mosfet's with very small physical dimensions," *IEEE Journal of Solid-State Circuits*, vol. 9, pp. 256–268, October 1974.

[2] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Dark silicon and the end of multicore scaling," in *Proceedings of the 38th Annual International Symposium on Computer Architecture*, ISCA '11, (New York, NY, USA), pp. 365–376, ACM, 2011.

[3] K. Chakraborty, *Over-provisioned Multicore Systems*. PhD thesis, University of Wisconsin-Madison, Madison, WI, USA, 2008. AAI3327881.

[4] S. Cho and R. Melhem, "Corollaries to amdahl's law for energy," *IEEE Comput. Archit. Lett.*, vol. 7, pp. 25–28, Jan. 2008.

[5] M. D. Hill and M. R. Marty, "Amdahl's law in the multicore era," *Computer*, vol. 41, pp. 33–38, July 2008.

[6] W. Song, S. Mukhopadhyay, and S. Yalamanchili, "Architectural reliability: Lifetime reliability characterization and management of many-core processors," *Computer Architecture Letters*, vol. PP, no. 99, pp. 1–1, 2014.

[7] B. Raghunathan, Y. Turakhia, S. Garg, and D. Marculescu, "Cherry-picking: Exploiting process variations in dark-silicon homogeneous chip multi-processors," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2013*, pp. 39–44, March 2013.

[8] S. Feng, S. Gupta, A. Ansari, and S. Mahlke, "Maestro: Orchestrating lifetime reliability in chip multiprocessors," in *Proceedings of the 5th International Conference on High Performance Embedded Architectures and Compilers*, HiPEAC'10, (Berlin, Heidelberg), pp. 186–200, Springer-Verlag, 2010.

[9] H. Kim, A. Vitkovskiy, P. V. Gratz, and V. Soteriou, "Use it or lose it: Wear-out and lifetime in future chip multiprocessors," in *Proceedings of the 46th Annual IEEE/ACM International Symposium on Microarchitecture*, MICRO-46, (New York, NY, USA), pp. 136–147, ACM, 2013.

[10] T. E. Carlson, W. Heirman, and L. Eeckhout, "Sniper: Exploring the level of abstraction for scalable and accurate parallel multi-core simulations," in *International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, pp. 52:1–52:12, Nov. 2011.

[11] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, "The gem5 simulator," *SIGARCH Comput. Archit. News*, vol. 39, pp. 1–7, Aug. 2011.

[12] M. M. K. Martin, D. J. Sorin, B. M. Beckmann, M. R. Marty, M. Xu, A. R. Alameldeen, K. E. Moore, M. D. Hill, and D. A. Wood, "Multifacet's general execution-driven multiprocessor simulator (gems) toolset," *SIGARCH Comput. Archit. News*, vol. 33, pp. 92–99, Nov. 2005.

[13] K. Ghose and et al, "Marssx86: Micro architectural systems simulators," in *ISCA Tutorial Session*, 2012.

[14] T. F. Wenisch, R. E. Wunderlich, M. Ferdman, A. Ailamaki, B. Falsafi, and J. C. Hoe, "Simflex: Statistical sampling of computer system simulation," *IEEE Micro*, vol. 26, pp. 18–31, July 2006.

[15] J. H. Ahn, S. Li, O. Seongil, and N. Jouppi, "Mcsima+: A manycore simulator with application-level+ simulation and detailed microarchitecture modeling," in *Performance Analysis of Systems and Software (ISPASS), 2013 IEEE International Symposium on*, pp. 74–85, April 2013.

[16] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: Characterization and architectural implications," in *Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques*, PACT '08, (New York, NY, USA), pp. 72–81, ACM, 2008.

[17] S. Woo, M. Ohara, E. Torrie, J. Singh, and A. Gupta, "The splash-2 programs: characterization and methodological considerations," in *Computer Architecture, 1995. Proceedings., 22nd Annual International Symposium on*, pp. 24–36, June 1995.

[18] P. Shivakumar and N. P. Jouppi, "CACTI 3.0: An integrated cache timing, power, and area model," 2001.

[19] D. Brooks, V. Tiwari, and M. Martonosi, "Wattch: A framework for architectural-level power analysis and optimizations," in *Proc. Int. Symp. on Computer Architecture (ISCA)*, pp. 83–94, 2000.

[20] K.Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-aware microarchitecture," in *International Symposium on Computer Architecture*, pp. 2–13, 2003.

[21] Z. Lu, W. Huang, J. Lach, M. Stan, and K. Skadron, "Interconnect lifetime prediction under dynamic stress for reliability-aware design," in *Proc. IEEE/ACM Int. Conf. on Computer Aided Design (ICCAD)*, pp. 327–334, IEEE, November 2004.

[22] X. Huang, T. Yu, V. Sukharev, and S. X.-D. Tan, "Physics-based Electromigration Assessment for Power Grid Networks," in *Proceedings of the 51st Design Automation Conference*, DAC '14, (New York, NY), pp. 1–6, ACM Press, Jun. 2014.

[23] S. Huck, "Measuring processor power tdp vs. acp," *White Paper, Revision*, vol. 1, 2011.