Leveraging CPU Electromagnetic Emanations for Voltage Noise Characterization

Zacharias Hadjilambrou* Shidhartha Das+ Marco A. Antoniades* Yiannakis Sazeides*
University of Cyprus * ARM Research+

The final version of this paper appears in 51st Annual IEEE/ACM International Symposium on Microarchitecture, 2018.

ABSTRACT

Worst-case dI/dt voltage noise is typically characterized post-silicon using direct voltage measurements through either on-package measurement points or on-chip dedicated circuitry. These approaches consume expensive pad resources or suffer from design-time and run-time overheads. This work proposes an alternative non-intrusive, zero-overhead approach for post-silicon dI/dt voltage noise generation based on sensing CPU electromagnetic emanations using an antenna and a spectrum analyzer. The approach is based on the observation that high amplitude electromagnetic emanations are correlated with high voltage noise. We leverage this observation to automatically generate voltage noise (dI/dt) stress tests with a genetic-algorithm that is driven by electromagnetic signal amplitude and to obtain the first-order resonance-frequency of the Power-Delivery LC-tank network. The generality of the approach is established by successfully applying it to three different CPUs: two ARM multi-core mobile CPU clusters hosted on a big.LITTLE configuration and an x86-64 AMD desktop CPU. The efficacy of the proposed methodology is validated through V_{MIN} and direct voltage noise measurements.

1 INTRODUCTION

The combination of higher current demand at scaled supply-voltages [1], high operating frequencies, aggressive low-power techniques [56] and increasing core-counts exacerbate supply-voltage noise for CPUs both in mobile [6][47][64] and server/desktop [2][31][60] market segments. Large voltage noise is a threat to robust execution because when the supply voltage drops below a certain threshold, timing violations or bit-flips may occur [1][31][56]. This may lead to silent data corruption (SDC), application or system crashes and general system instability [2][16].

Manufacturers budget voltage margins (or guardbands) to ensure robustness even in the presence of worst-case voltage noise conditions [1]. Consequently, production systems are typically operated at a higher supply voltage (and/or lower clock-frequency) than necessary under nominal conditions of operation. Accurate determination of voltage margins is critical since optimistic margining (where the added margins are not adequately provisioned for the rare worst-case noise event) can cause abrupt system-failures in the field. In contrast, excessive margining adversely impacts CPU energy-efficiency [1][2][29][31][45][59].

A key aspect of margining production systems is the determination of the worst-case inductive component (referred to as "LdI/dt" or "dI/dt"), of the voltage noise [31] that typically dominates over the resistive component (referred to as " IR") in the Power Delivery Network (PDN) of modern computing systems[6][29][43]. As discussed in Section 2, the PDN is a distributed system composed of the chip, the package and the Printed Circuit Board (PCB) whose equivalent-circuit model consists of multiple LC-tanks, each with its own distinct resonance frequency. The tank circuit formed by on-chip capacitance and the package parasitic-inductance has the highest resonance frequency, referred to as the "1st-order resonance frequency". Abrupt changes in CPU current demand, such as due to branch-misprediction or cache-misses, causes large-magnitude voltage noise oscillations excited at the 1st-order resonance frequency. In comparison with aperiodic or isolated dI/dt events, periodic current modulations at this frequency reinforces resonant noise even further [53], thereby maximally stressing system-margins.

Commercial Electronic Design Automation (EDA) tools [70][71][72] cannot accurately model the time-varying CPU current due to the complex hardware/software interactions, particularly in multi-core configurations [31]. Consequently, design-time PDN optimization is inadequate and post-silicon

[1] Voltage margins are also necessary for variation effects such as temperature hot-spots, circuit-aging and process-variation effects [53]. However, system-margins are typically stressed most due to LdI/dt or inductive transients. Their fast-moving nature [27][29] renders them difficult to compensate for using traditional adaptive techniques.

characterization is essential for margining production systems [31] [64].

Post-silicon characterization typically relies upon synthetic virus workloads, referred to as dI/dt stress tests [31]. Due to the inherent complexity of manually crafting these tests, previous work [2][31][39] introduced frameworks for automated generation of stress tests based on optimization techniques such as Genetic Algorithms (GA). These approaches rely upon the capability of the platform-under-test to support high-bandwidth monitoring of on-chip voltage rails or direct voltage measurements.

There are two main approaches for direct voltage measurement: 1) specialized on-chip circuitry integrated into the system at design-time [5][21][32][47][65][66] and 2) voltage sense pins located on the package [1][2][48][49] (also known as Kelvin measurement points). Unfortunately, these capabilities are not yet mainstream features, particularly in cost- and resource-constrained mobile platforms. Moreover, on-chip approaches incur the Non-Recurring Engineering (NRE) cost of hardware development and system-integration at design-time. In cases where the voltage monitor is integrated into the system as a peripheral device, they require additional software support (in terms of a device driver) to configure, calibrate and query. In contrast, on-package measurement points directly connected to on-chip voltage rails do not incur design-time NRE overheads. Nonetheless, they require a dedicated pair of Controlled Collapsible Chip Connection (C4) [50] bumps for each voltage-domain. This consumes valuable C4 resources that could otherwise be used for direct power-delivery. Consequently, such support is not usually provided in resource-constrained platforms such as mobile CPUs (e.g. the Cortex-A53 CPU used in this paper).

1.1 Contributions of this work

In this work, we propose an alternative approach for post-silicon dI/dt stress test generation and PDN resonance frequency measurement. The proposed approach relies upon sensing CPU electromagnetic (EM) emanations using an antenna and a spectrum analyzer connected to the antenna. Compared to direct-measurement, our approach offers the following unique advantages for resonant voltage noise analysis that stresses safety-margins worse than isolated or aperiodic dI/dt events [2][31][56][62]: a) is non-intrusive, as no physical connection to the CPU is required, b) has zero-overhead, as it does not require design time, development effort, on-package and on-chip resources, and c) is cross-platform, as it can be applied to virtually any platform.

Due to its general applicability, we believe that our approach is a fundamentally new way of benchmarking commercial systems that democratizes PDN characterization and voltage noise research. Voltage noise visibility is not a standard feature supported in motherboards and researchers do not usually have access, when available, to proprietary on-chip voltage noise circuits. Consequently, voltage noise visibility requires a chip and a motherboard that exposes high bandwidth voltage measurements points. The proposed EM methodology removes these requirements by allowing basic PDN characterization to be performed on any CPU and motherboard without the need for direct fine-grained voltage measurements. As our main contributions in this work, we:

- Explain the theoretical basis and provide conclusive evidence for the correlation between on-chip voltage noise and emanated EM power. Our measurements demonstrate that both on-chip voltage noise and EM-signal power are maximized at the 1st-order resonance frequency.
- Leverage the above observation to propose a convenient, zero-overhead, cross-platform and non-intrusive way for PDN characterization. We demonstrate that with the proposed EM approach, it is possible to: a) monitor periodic voltage noise of large amplitude b) generate dI/dt stress tests within a GA framework that optimizes towards a maximum EM signal amplitude, and c) rapidly measure the 1st-order resonance frequency and d) detect resonance frequency shifts due to capacitance changes in multi-core configurations, e.g. due to dynamically switching on or off cores in a CPU cluster.
- Establish the cross-platform applicability of the EM approach by successfully applying it to three different CPUs spanning multiple Instruction Set Architectures (ISA). We characterize the PDN for individual CPUs across separate platforms and distinct processor-clusters integrated on the same die. In particular, the EM methodology is applied on three different CPUs: two ARM multi-core CPU clusters (dual-core Cortex-A72 and quad-core Cortex-A53) hosted on a Juno Board [13] and one x86-64 AMD desktop CPU (Athlon II X4 645). Thus, the proposed approach is shown to work across CPUs of different market segments (mobile and desktop/server), different ISAs (ARM and x86), different CPU micro-architectures, different technology nodes and on CPUs that do not offer direct voltage measurements such as the Cortex-A53 cluster on the Juno platform. The efficacy of the proposed approach is validated through direct voltage measurements (where it is feasible) and V_{MIN} determination (minimum stable operational voltage for a given frequency).

The remainder of the paper discusses background on PDN and the theoretical basis linking PDN voltage noise and EM emanations (Section 2), the GA framework used for stress test generation (Section 3), the experimental details (Section 4), the EM methodology validation and evaluation with a Cortex-A72 CPU (Section 5), EM methodology evaluation with a Cortex-A53 CPU and an AMD Athlon II X4 645 CPU (Sections 6 and 7), general insights from the analysis (Section 8), related work (Section 9) and conclusions and future work (Section 10).

2 THEORETICAL BASIS

This section describes the PDN fundamentals and explains why large CPU voltage noise causes high amplitude EM emanations.

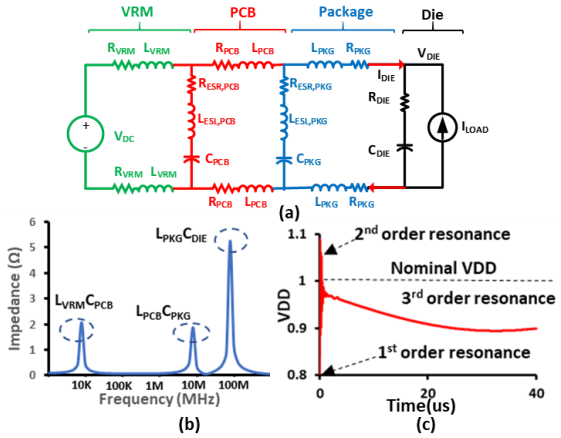

Figure 1. (a) A simplified model of the PDN [43]. The impedance as seen by the die has multiple resonance frequencies, shown in the frequency-domain response in (b) and time-domain response to a step-current excitation in (c)

2.1 Power Delivery Network (PDN) Fundamentals

Figure 1 (a) shows a simplistic representation of the PDN of a die-package-PCB system [6][43]. The current demand due to on-chip switching transistors is modelled as a lumped current source, I_{LOAD}. Explicit decoupling capacitors (henceforth, referred to as decaps) and non-switching, but powered-on, transistors act as localized charge reservoirs that provide the high-frequency component of the demand current, I_{LOAD}. The on-chip power-grid resistance is modelled as a lumped resistor, R_{DIE}, connected in series with C_{DIE}. The total die current (I_{DIE}) is sourced through the inductive power-line traces of the package and the PCB, represented by a series R-L (resistor, inductor) equivalent circuit. The discrete decaps on the PCB and package are represented by an ideal capacitance (C_{PKG}, C_{PCB}) in series with its effective series inductance (ESL) and effective series resistance (ESR). Figure 1(b) shows the input impedance of the distributed RLC network as seen from the die. The impedance spectrum shows multiple resonance peaks due to multiple LC-tank circuits. The highest impedance peak, referred to as the 1st-order resonance peak is attributed to the die-capacitance (C_{DIE}) interacting with its counterpart inductance (L_{PKG}). The 1st-order resonance also occurs at the highest frequency (50MHz-200MHz) compared to the 2nd- (~1-10MHz) and 3rd-order (~10KHz) resonances that are due to downstream capacitor networks.

The resonance frequencies also manifest in the time-domain when the PDN is excited by a step-current excitation (Figure 1 (c)). Micro-architectural events such as branch mispredictions [6] can trigger these oscillations in the PDN. Power-supply oscillations of larger magnitudes can be set off

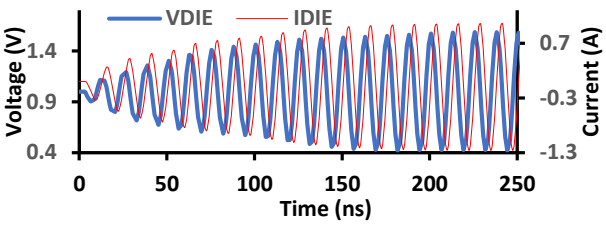

Figure 2. Simulated waveforms showing the die voltage (V_{DIE}) and die current (I_{DIE}) in the simplified PDN model in Fig. 1. I_{LOAD} triggers the first-order resonance by pulsing at 80MHz. This causes both V_{DIE} and I_{DIE} to undergo large-magnitude oscillations, maximizing the radiated EM power.

within the supply network due to sustained program activity with alternating periods of high-current and low-current consumption within a loop [2][16]. When the frequency of the time-varying current aligns closely with the 1st-order resonance frequency, voltage oscillations are maximized in amplitude (Figure 2). High voltage oscillations can lead to bit-flips in arrays, timing errors in logic paths [1][2][7][16] and reliability issues due to gate-oxide stress [7][8]. Such periodic events often result in system/application crashes and/or incorrect execution output [2][45].

2.2 Relationship Between CPU EM Emanations and On-Chip Voltage Noise

It is well-known that metallic conductors act as transmitting antennae that emanate EM radiation under oscillating voltage and current stimulation [17][20]. On-chip interconnections and transistors act as distributed radiating antennae due to time-varying current consumption induced through normal program execution. Simple periodic activity, such as that due to instruction loops, cause periodic variations in CPU power (i.e. sequence of DIVs followed by ADDs) that manifest as visible spikes in the EM spectrum, at a frequency F equal to 1/T (where T is the loop period) [9].

Fundamental antenna theory (say, for a traditional Hertzian dipole) states that the component of the radiated power for the transmitting antenna, at a specific frequency, varies quadratically with the amplitude of the oscillating feed current [20] at the corresponding frequency and the so-called radiation resistance[2]. Periodic current load (I_{LOAD}), pulsing at the first-order resonance frequency, can trigger sustained oscillations of large magnitude in V_{DIE} and I_{DIE}.

We simulate the simplified PDN model in Figure 1 (a) with a persistently pulsing current excitation (I_{LOAD}) at 80MHz which matches the 1st-order resonance frequency (Figure 1 (b)). This sets of resonant oscillations in the PDN as illustrated by HSPICE [52] simulations in Figure 2. At resonance, both voltage and current oscillations maximize in amplitude. This, in turn, maximizes the radiated EM power

[2] The radiating resistance of a conductor can be differentiated from its loss resistance, in that the former is a function of the geometry of the conductor and determines the magnitude and the directivity of the radiated power [20]. The loss resistance, in contrast, manifests as ohmic losses dissipated through the conductor.

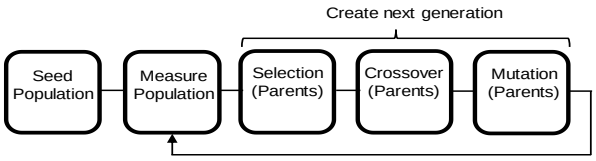

Figure 3. A Typical GA flow.

from the on-chip distributed antennae, due to the quadratic dependence with oscillatory current amplitude. Therefore, measuring the frequency at which the amplitude of the emanated EM power is maximized directly reveals the 1st-order resonance frequency. We leverage this relationship between radiated EM power and on-chip voltage-noise to maximize the voltage-noise by maximizing the amplitude of EM signals.

We validate the above theory that links CPU EM emanations with on-chip voltage noise in Section 5 using the ARM Juno [13]. The Juno board supports fine-grained, voltage-noise measurements in the time-domain. The measurements presented in Section 5 confirm that a) maximization of EM power is strongly correlated with higher amplitude voltage noise, and b) emanated EM power is maximized at the 1st-order resonance frequency. We establish the generality of the approach in Sections 6, 7 by applying it on a Cortex-A53 and an AMD Athlon CPU.

3 GENETIC ALGORITHM STRESS TEST GENERATION FRAMEWORK

Central to our work is a GA-based optimization framework that automatically produces code that maximizes EM radiation amplitude. GA for dI/dt stress-test (virus) generation is proposed in previous work [2][39]. We do not claim novelty for the use of GA for virus generation but for showing that EM emanations can be leveraged for dI/dt virus-generation. Next, we provide the GA framework flow implementation details for the paper's reproducibility and readability purposes.

3.1 GA Basics

GAs typically optimize a target metric by using operators inspired by bio-logical evolution, such as crossover (exchange of genes), mutation and selection of the fittest individuals for breeding [12]. Previous work has demonstrated the efficacy of GAs at generating synthetic stress tests that maximize power consumption [3][4][23] and voltage noise [2][16][39]. The difference with prior work is in our usage of the maximum EM-power amplitude as an optimization metric to drive the GA . Figure 3 shows a typical GA flow that is adopted as follows for our study:

a) **Initial Seed Population:** The first step is to create an initial seed population (generation). This can be either a new random initial population or a population from a previous GA run. In our case, the initial population is a set of random assembly instruction sequences (either ARM or x86). In GA terminology, each sequence of assembly instructions represents an **individual** of the population. We empirically find that population size of 50 individuals works well for our optimization goals.

b) **Measure Individuals:** The second step involves compiling each individual instruction sequence, executing the resulting binary and measuring the optimization metric of interest. In this work, for the EM optimization, the metric of interest is maximum EM amplitude at any frequency in the spectrum of 50-200MHz (the spectrum where the 1st order PDN resonant frequency is typically located). The metric used for maximum EM amplitude is the mean root square of 30 samples. This work also performs dI/dt virus generation based on voltage feedback (on CPUs that provide this capability) for validation and comparison purposes. For voltage driven optimizations, the target metric is either maximum voltage droop or peak to peak (maximum – minimum) voltage amplitude.

c) **Creating next generation:** The algorithm creates a new population after all individuals are measured. The new population is created by selecting the fittest (e.g. the ones that scored the highest EM amplitude) individuals as parents, exchanging instructions among the two parents (crossover) and performing mutation. A mutation operation converts an instruction or an instruction-operand (such as a register) into another, with a conversion probability, referred to as the "mutation rate". We empirically determined that the following mutation rate, crossover and parent selection operands work well for our case study: a) 2-4% mutation rate, b) one-point crossover, and c) tournament selection.

3.2 GA Implementation and Configuration Details

The GA framework is developed with Python. The framework is executed on a separate workstation different from the optimization target CPU. This workstation creates the seed population, and applies the parent selection, crossover and mutation operators. Communication between the workstation and the target machine is achieved using the Secure Shell (SSH) protocol. The workstation sends the source code of an individual to the target machine, and, the target compiles and runs the binary. While the binary is running, the workstation drives the measurement instrument (e.g. spectrum analyser) to measure and record each individual. Upon completion of a measurement, the workstation terminates the binary execution on the target machine. Empirically, we observe that satisfactory results are obtained after running the GA for at least 60 generations. The algorithm execution is typically limited by the measurement latency per individual. For instance, when optimizing for EM amplitude, approximately 18 seconds are needed to take 30 measurements which translates to an execution time of ~15 hours for 60 generations (and 50 individuals per generation).

The assembly instructions that are used in the GA optimization are described by the user in an XML input file. The user can also specify what registers each instruction may

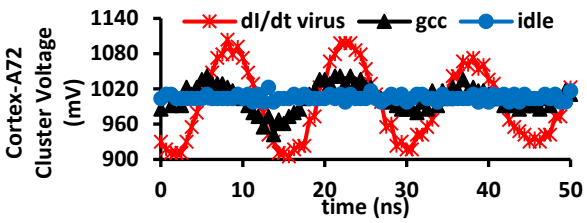

Figure 4. Voltage waveforms obtained from OC-DSO for 3 different workloads. dI/dt virus causes the largest voltage noise.

use as well as the memory addresses for the memory instructions. Essentially, the GA framework optimization evolves around finding the right mix, dependencies and order of instructions that maximize the target optimization metric.

3.3 GA Instruction and Data Mix

The GA framework relies on a user-specified template consisting of pre-initialized registers and an empty loop body that is eventually filled with the assembly instructions composing an individual. The GA optimization, relies on representative instructions that deliberately target diversity in latency (both single and multi-cycle) and instruction-type (integer, floating-point, SIMD and load/store) to facilitate rapid convergence. In particular, for the ARM ISA [40] we use: a) short latency integer instructions such as move (MOV), add (ADD) b) multi-cycle long latency integer instructions such as MUL and DIV, c) floating point equivalents of the above arithmetic instructions d) equivalent SIMD instructions using SIMD registers e) unconditional dummy branches pointing to the next instruction (conditional branches are difficult to incorporate as they can introduce non-determinism), and f) load and store memory instructions. For the x86 instruction set, the same instruction mix selection principles as with ARM are used with some minor modifications. Since x86 does not have explicit load-store instructions, memory operations are implemented by using memory address operands for integer instructions. For SIMD operations, SSE2 [41] instructions are used. As shown subsequently, in Section 8, the viruses make use of nearly all instruction types to maximize voltage noise. This clearly illustrates that it is essential to have diverse set of instruction types to select from during GA optimization.

We deliberately avoid cache misses due to the timing non-determinism introduced by them. The GA should give preference to instruction sequences with periodic current swings triggering first-order resonant oscillations in the PDN. Thereby, events such as cache misses that introduce time variability should be avoided as they result in significant jitter to the GA algorithm, which in turn impedes its convergence. Nonetheless, memory references, even if they are always hits, are found to be essential for maximizing voltage noise due to engaging the memory subsystem (pipeline resources and L1 cache).

4 MEASUREMENT SETUP

Table 1 shows an overview of the ARM and the AMD platforms used in this study. The ARM Juno [13] platform hosts a heterogeneous multiprocessing System-on-Chip (the so-called big.LITTLE configuration) consisting of separate clusters of the dual core Cortex-A72 and a quad core Cortex-A53 [42]. The platform integrates an on-chip power-supply monitor configurable as a digital storage oscilloscope (OC-DSO) [5] that is ideal for validating our proposed EM methodology. The OC-DSO provides fine-grained sampling (up to 1.6GHz bandwidth) of the voltage rails supplying the dual-core Cortex-A72 cluster. The capability of OC-DSO to capture voltage noise is illustrated in Figure 4 with the dI/dt virus causing much larger noise than a regular SPEC2006 benchmark and CPU idle state. The JUNO board also offers a synthetic current load (SCL) [16] block integrated in the OC-DSO. The SCL loads the Cortex-A72 PDN with a square-wave current excitation at various frequencies. This is useful for detecting the Cortex-A72 PDN resonant frequency [16] and we also use the SCL in this work to validate the EM methodology. The Cortex-A53 cluster does not benefit from OC-DSO or SCL circuit because it is in a separate voltage domain. Cortex-A53 voltage domain lacks any explicit support for voltage-noise measurement. The Juno board runs a Debian OS with a 4.4.0-135-arm64 kernel. The DS-5 debugger [26] is used to access OC-DSO, sweep CPU frequency, change supply-voltage and power-gate both the Cortex-A72 and Cortex-A53 clusters, orchestrated through a system control processor (SCP) that enables this functionality [5].

For the AMD setup, an Athlon II X4 645 CPU is used that is hosted on an ASUS M5A78L LE motherboard and Windows 8.1 OS. AMD Overdrive application [25] is used to change the voltage and the frequency of the CPU. This application also includes a stability test that is evaluated and compared against the GA generated dI/dt viruses. The

Table 1. Experimental platform details.

MB	CPU	# of Cores	ISA	uArch	Highest Freq,Vol Point	Technology (nm)	OS	Voltage noise visibility
Juno Board R2	Cortex-A72	2	ARM	Out of Order	1.2GHz,1V	16	Debian	OC-DSO
Juno Board R2	Cortex-A53	4	ARM	In-Order	0.95GHz,1V	16	Debian	None
Asus M5A78L LE	Athlon II X4 645	4	x86-64	Out of Order	3.1GHz,1.4V	45	Windows 8.1	On-package pads

Figure 5. Experimental setup for the ARM Juno board (left) and AMD desktop CPU (right).

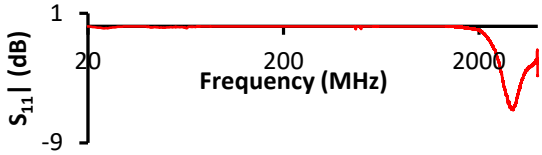

Figure 6. Measured $|S_{11}|$ for the square loop antenna indicating a self -resonance around 2.95 GHz.

motherboard integrates on-package Kelvin measurement pads that enable direct external monitoring of the on-chip voltage rails using differential probes connected to a bench-top oscilloscope.

Figure 5 shows both the ARM Juno and AMD desktop PC experimental setups. We use a square loop antenna (3 cm side length) as a receiver for the emanated EM radiation. We measure the frequency response of the antenna to monitor for self-resonance frequencies in the range of interest (50 MHz – 200 MHz). Figure 6 shows the single-port scattering parameter (S11) measurement of the antenna for a wide-frequency range [51]. The antenna has a relatively flat frequency response from DC until 1.2 GHz, with a self-resonance frequency at 2.95 GHz. Thus, we confirm that the antenna does not modulate the received signal in the frequency range where we expect the first-order resonance frequency of the PDN to lie (50 – 200 MHz). Furthermore, even though the antenna is not well matched in the frequency range of 50 – 200 MHz, it is still able to receive the emanated EM radiation in close proximity to the CPUs. The antenna is connected to a spectrum analyzer through a low-loss coaxial cable to receive the emanated waves from the experimental platforms. The spectrum analysers Agilent E4402B (Juno setup) and Agilent N9332C (AMD setup) are used to measure the EM signals. Cheaper commercial software-defined radio receivers should also work [10]. The antenna is placed at a stable position 5-10cm close to the monitored CPUs. We record strong EM signals on either side of the PCB but prefer the lower side due to proximity to the die. If required to increase the strength of the signal pre-amplifiers and antenna matching networks can be used. Also, instead of spectrum analyser oscilloscope can be used for time-domain measurements (and for frequency domain if it supports real time FFT analysis).

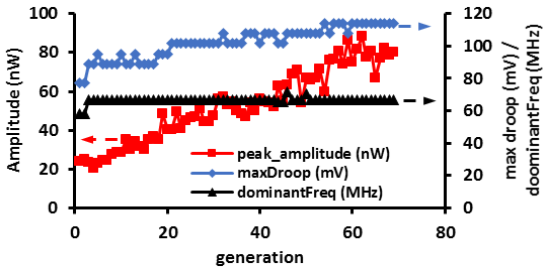

Figure 7. EM driven GA run on Cortex-A72. Peak amplitude (left axis) and maximum droop / dominant frequency (right axis) for the best individual of each GA generation.

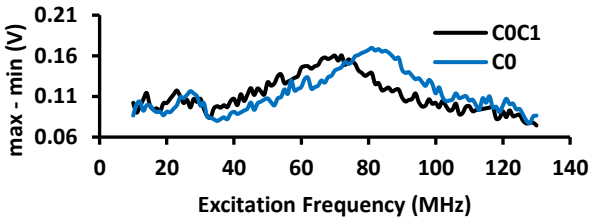

Figure 8. SCL stimulus reveals a resonant frequency in the range of 66-72MHz with two powered cores (C0C1) and 80-86MHz with one powered core (C0).

5 EM METHODOLOGY VALIDATION

5.1 EM Emanations and Voltage-Noise Correlation

A GA search is performed on the Cortex-A72 with target to maximize EM amplitude (in the 1st order resonant frequency range of 50MHz-200MHz). Figure 7 shows how the EM amplitude and dominant frequency of the strongest individual of each generation varies as the GA progresses. The figure also plots the maximum voltage droop caused by the strongest individual per generation (we obtain the droop using OC-DSO by re-running and measuring each individual after the GA search has finished). It is clearly seen that as the signal amplitude increases from generation to generation during the GA search the voltage droop increases as well. Therefore, it is safe to say that the GA search driven by EM signal amplitude essentially maximizes voltage noise. Furthermore, we observe that from the very first generations, the GA prefers individuals that have a dominant frequency at 67MHz (the frequency with the highest EM amplitude). To check whether this frequency is the Cortex-A72 PDN 1st order resonant frequency we use the methodology described in [16]. Particularly, we load the PDN with a square-wave current at various frequencies in steps of 1MHz using the SCL circuit. We record the peak-to-peak voltage oscillation at each frequency with the OC-DSO. The highest voltage oscillation reveals the resonant frequency [2][16]. The results of the sweep are shown in Figure 8 according to which the first-order resonance frequency lies in the range between 66-72MHz (we observe a relatively flat frequency response around resonance) when both cores in the cluster are powered

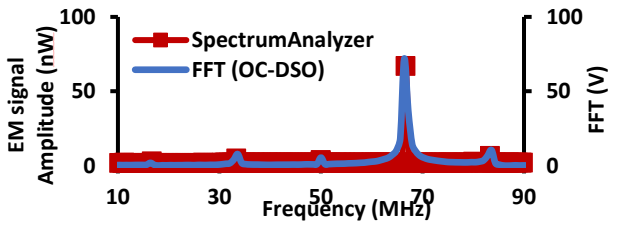

Figure 9. Comparison of spectrum analyzer readings (left axis) with FFT of OC-DSO voltage readings (right axis) during execution of EM dI/dt virus. The two measurements agree as they reveal spikes at the same frequencies.

up (indicated by the label "C0C1" in the plot). This agrees with the dominant frequency identified by the GA. This shows that GA guided by EM amplitude successfully identifies the resonant frequency. Moreover, this provides strong evidence that voltage noise and EM signal amplitude are both maximized at the resonant frequency.

To confirm the EM amplitude and voltage noise correlation further, we obtain the frequency-domain representation (using the Fast Fourier Transform (FFT) algorithm) of the voltage samples from the OC-DSO while executing the EM dI/dt virus. Figure 9 compares the spectrum analyzer readings of the EM power captured by the receiver antenna with the FFT of OC-DSO voltage readings. The dominant frequency of both frequency-domain representations is exactly aligned at 67MHz.Moreover, the two instruments agree on other less dominant spikes as well, such as the virus's base loop frequency (1/loop period) located at 16.66MHz.

5.2 V_{MIN} Tests on Cortex-A72

The virus generated in the previous section must ultimately limit the stability of the overall system due to the magnitude of voltage oscillations it generates. We quantify system-stability due to a workload by measuring the minimum operational voltage (V_{MIN}) at which the workload is executed correctly. Figure 10 compares the V_{MIN} of the EM virus against that of the SPEC2006 benchmarks and the V_{MIN} of a virus generated by the GA framework when optimizing for maximum voltage droop measured by the OC-DSO. All workloads are executed on both the Cortex-A72 cores, with each core running a separate instance of the workload. Each experiment is started at a high voltage and the voltage is progressively lowered in steps of 10mV until a system crash is observed. The workloads are run until completion and then the output is checked for SDC (by comparing the output against a golden reference obtained at nominal operating voltage of 1.0V). Figure 10 reports the highest voltage at which any deviation from the nominal execution is observed, either due to a SDC, an application crash or a system crash. We have observed (not shown in the figure) that typically, workloads suffer SDC or application crash approximately 10mV above the system crash. Both EM and OC-DSO viruses clearly cause higher voltage droop (in excess of 25mV compared to the "lbm", the SPEC benchmark with the highest voltage droop) and have higher V_{MIN} compared to the other workloads (20mV higher V_{MIN} compared to "lbm"). Both viruses (generated by targeting EM power or maximum voltage droop) stress the PDN in approximately similar manner.

These results support the claim that EM-driven GA is a feasible and reliable method for generating dI/dt viruses for post-production characterization and voltage margin determination. For statistical confidence in our measurements, we perform 30 V_{MIN} tests for each virus and two V_{MIN} tests for each SPEC benchmark. SPEC benchmarks are executed with reference inputs, and, therefore, total V_{MIN} experimentation time is equal to about two days. Thereby, the SPEC benchmarks run for significant amount of time at voltages lower than the viruses' V_{MIN} without any failure.

5.3 EM Methodology for Quick Determination of the PDN Resonant Frequency

As shown in Section 5.1, the GA framework is an effective approach for maximizing voltage-noise and obtaining the resonant frequency, based only on external EM readings. However, the algorithm requires multiple generations for convergence and may require many hours to terminate. Therefore, having a quick independent method for quickly finding the resonant frequency is useful for various reasons such as a) to validate GA results, b) to constrain the spectrum analyser measurements during EM GA search to a smaller band of frequencies to minimize the measurement time and, hence, the GA search time, c) for post-production purposes like PDN simulation validation, tampering detection etc. To speed-up 1st order resonant frequency detection we propose

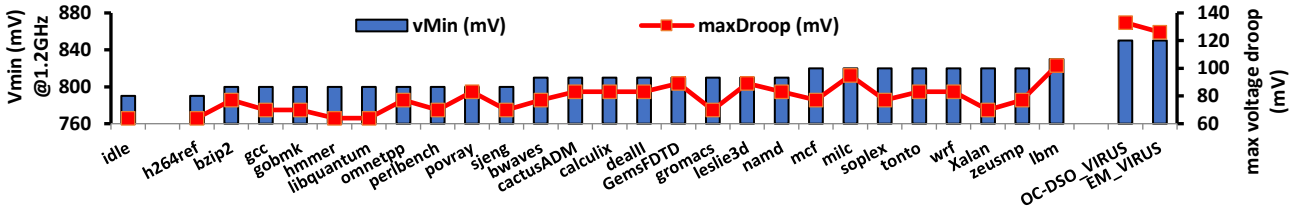

Figure 10. V_{MIN} (blue bar, left axis) and maximum voltage droop (red curve, right axis) of various workloads for dual core runs. Viruses (rightmost workloads) cause higher droop and have higher V_{MIN} than typical benchmarks.

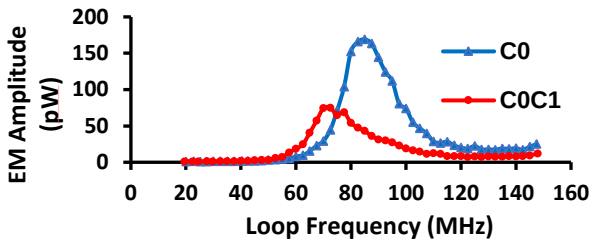

Figure 11. EM resonant frequency exploration for Cortex-A72 PDN with workload loop frequency modulated by CPU frequency.

an alternative EM based method that in our setup completes in approximately 15 minutes (instead of many hours). Description of this approach is given next.

The first step is to manually design a simple instruction loop composed of separate high and low current consuming sequences. This is not supposed to be a proper dI/dt stress test but a loop that causes merely enough current variation to result in a visible EM spike at a frequency equal to the loop frequency (which is equal to the inverse of the loop iteration time). We then run the loop and sweep the CPU frequency which consequently modulates the loop frequency and the EM spike frequency. The spike amplitude is maximized when the loop frequency matches the resonant frequency because the fluctuating loop-current will trigger resonant oscillation in the PDN [6]. Therefore, after the frequency sweep is over, the frequency at which the highest EM amplitude occurs reveals the resonant frequency.

In this specific case-study, we used a loop with the high current consuming sequence consisting of eight ADD instructions that are executed in 4 CPU cycles and a low current consuming sequence consisting of a single DIV instruction that takes 4 CPU cycles to execute. The difference in power consumption can be attributed to the fact that the core sustains an issue rate of two instructions per CPU cycle for the single-cycle integer instructions whereas the multi-cycle DIV instruction achieves 0.25 instructions executed per CPU cycle. The period of execution of the overall loop (with both the high-current and the low-current consuming portions) is 8ns at 1.2GHz CPU frequency. This corresponds to a loop frequency of 150MHz. To modulate the loop frequency, we sweep the CPU frequency from 1.2GHz down to 120MHz in steps of 20MHz (the frequency step is limited by the multiplier which defaults to 20) and we record the EM signal amplitude at each frequency point. Figure 11 shows the results of the frequency sweep. The amplitude is maximized at around 70MHz loop frequency when both cores are powered up (labelled by "C0C1") and at 85MHz when just one core is powered up (labelled by "C0"). These results confirm the 1st order resonant frequency ranges for C0C1 and C0 scenarios determined in Section 5.1 and Figure 8 using the SCL circuit and the OC-DSO. This proves the effectiveness of the proposed approach in quickly identifying the 1st order resonant frequency. Note that is expected the 1st order resonance frequency to increase with less powered cores because it is inversely proportional to the die capacitance [43].

To conclude, the findings in this section provide strong support for the claims in Section 2.2 about the relationship between CPU EM emanations and PDN voltage noise. Also, the section shows that with the EM approach is possible to generate worst case dI/dt stress tests and identify the resonant frequency. We proceed next to establish the generality of the proposed methodology by applying it to different CPU cores and different platforms. In the next section, we apply the EM methodology on the Cortex-A53 cluster on the Juno platform and in Section 7 to an AMD CPU.

6 EM METHODOLOGY ON CORTEX-A53

Cortex-A53 cluster does not provide any support for direct voltage-noise measurements rendering dI/dt virus generation and resonant frequency identification impracticable. This section shows that the EM methodology circumvents this shortcoming to obtain a) a virus that stresses voltage margins, and b) the first-order resonance frequency. This underlines the effectiveness and the generality of the proposed methodology.

Both the Cortex-A53 and Cortex-A72 clusters implement the same version of the ARM ISA. Hence, we conduct a GA optimization run, with the same optimization parameters as in Section 5.1, but with the objective of obtaining a voltage-noise virus for the Cortex-A53 cluster. Figure 12 shows the inter-generational progression of the GA (left-axis showing received EM-power and the right-axis showing the dominant frequency of the strongest individual per generation). The GA successfully maximizes the EM amplitude. Since, Cortex-A53 does not support voltage noise measurements to test the effectiveness of the GA we compare the V_{MIN} of the strongest individual across all generations (labelled "EM virus") against the V_{MIN} of SPEC2006 benchmarks.

Figure 14 shows the V_{MIN} of the EM virus (rightmost) compared to SPEC2006 benchmarks and idle (leftmost). The V_{MIN} is obtained with four active cores at a 950MHz CPU frequency using the V_{MIN} test methodology described in Section 5.2 but applied to the Cortex-A53. The V_{MIN} of the generated EM virus stands out (50mV higher) compared to the rest of the benchmarks which demonstrates the effectiveness of the EM approach in generating dI/dt viruses.

The GA converges to 75MHz dominant frequency. We use the fast methodology described in Section 5.3 to validate that this is the first-order resonance frequency of the Cortex-A53 cluster. The results of the sweep are shown in Figure 13. For four powered cores (C0C1C2C3 scenario) the sweep reveals a resonance frequency at 76MHz which matches closely the GA results. The agreement of the two independent approaches gives confidence that the resonant frequency is correctly identified.

Furthermore, Figure 13 provides insight about how power-gating can affect significantly the PDN characteristics. The

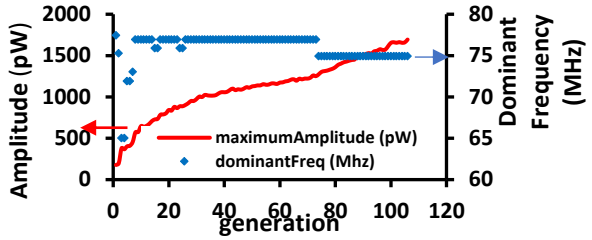

Figure 12. GA EM amplitude driven optimization for Cortex-A53.

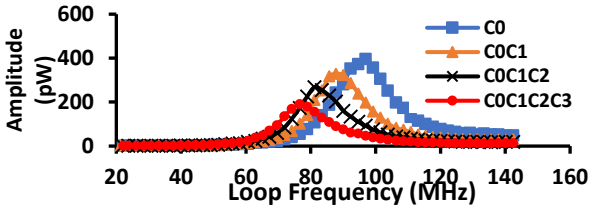

Figure 13. Resonant frequency exploration on Cortex-A53. For four powered cores (C0C1C2C3) the resonant frequency is 76.5MHz.

Cortex-A53 quad-core cluster has the highest die capacitance when all four cores are powered up ("C0C1C2C3"). The first-order resonance frequency is inversely proportional to the square-root of the die capacitance [43], hence, the resonance frequency increases from 76.5MHz when all cores are powered up (labelled as "C0C1C2C3") to 97MHz with just one core powered up (labelled as "C0"). Note that the amplitude of the EM emanations is affected by the number of powered cores in addition to the resonance frequency. Since we kept stable current consumption across all four scenarios by having only the first core active, the EM amplitude (and hence the voltage noise) is maximized in the scenario where the least PDN capacitance is present ("C0"). These results confirm prior work [58] that shows that with more cores connected under the same PDN, the capacitance increases and voltage noise smooths out. Moreover, the results indicate that power-saving techniques, such as power-gating individual cores, whilst being beneficial from a leakage perspective, can affect power-delivery adversely. Power-gating not only reduces the available useful capacitance that can mitigate high-magnitude voltage-droops, but also makes the frequency of voltage-noise oscillations higher. This has detrimental implications on voltage-noise mitigation mechanisms such as adaptive-clocking [21][29], that are extremely sensitive to response-latency.

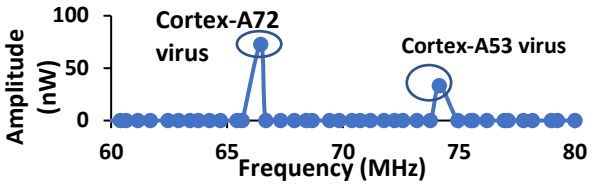

Figure 15. Simultaneous monitoring of voltage emergencies across multiple voltage domains through EM emanations.

6.1 Simultaneous Voltage Noise Monitoring of Multiple Voltage Domains

We next illustrate the capability of the EM based methodology to monitor multiple voltage domains simultaneously. This is impossible with an on-chip or off-chip oscilloscope that has a direct physical probing on a single voltage domain. In contrast, an antenna can detect voltage emergencies happening at the same time on both the Cortex-A72 and Cortex-A53. To demonstrate this capability, we run the Cortex-A72 and Cortex-A53 dI/dt viruses at the same time and capture the spectrum analyzer readings as shown in Figure 15. The frequency-domain signatures of both viruses are clearly visible. This shows that the EM methodology offers an effective detection mechanism for voltage-noise oscillations occurring across multiple voltage domains, thereby underlining its applicability to heterogeneous System-on-Chips (SoCs).

7 EM METHODOLOGY ON AMD CPU

This section extends the evaluation from low-power mobile CPUs and the ARM ISA to high power x86-64 desktops (AMD Athlon II X4 645). The fast EM frequency sweep methodology for finding the resonant frequency (Section 5.3) is performed on the AMD CPU and the results are shown in Figure 16. The sweep reveals the first-order resonance frequency to be at 78MHz. An EM amplitude driven GA run shows excellent agreement converging to nearly the same resonant frequency (77MHz) as shown in Figure 17. The EM amplitude during the GA search follows the same trends as in the Juno board CPUs (Figure 7, Figure 12), it increases with each generation until it eventually converges.

For V_{MIN} comparison, the GA auto-generated EM virus is compared against common Windows (and Desktop CPU) workloads. The benchmark suite includes CPU intensive

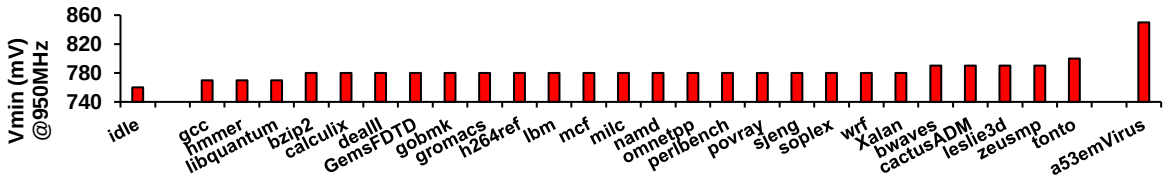

Figure 14. V_{MIN} measurements on Cortex-A53.

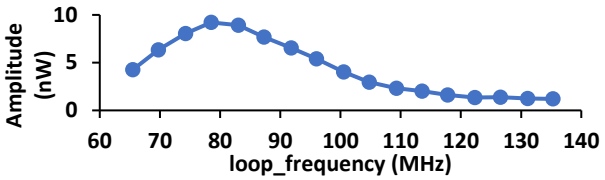

Figure 16. Loop frequency sweep on Athlon II X4 645 reveals a resonant frequency at 78MHz.

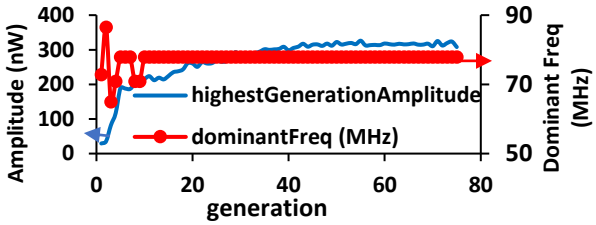

Figure 17. GA EM amplitude driven run on AMD CPU.

video rendering workloads such as Blender [33], Cinebench [34], scientific workloads such as Euler 3D [35] and all-around benchmark suites such as WEBXPRT [36] (mimics browser workloads) and GeekBench [37] (set of common workloads e.g. encryption, database queries etc.). Moreover, the EM virus is compared against the well-known Prime95 [38] stability test, AMD's own stability test application [25], and a GA virus generated through the voltage feedback from on-package Kelvin measurement pads (denoted as OscVirus). We monitor on-die voltage noise using a differential probe connected to an oscilloscope. The V_{MIN} and voltage noise results are shown in Figure 18. Unless noted otherwise, all measurements are with all four cores active.

The GA viruses (EMvirus, OscVirus) cause much higher voltage noise and have higher V_{MIN} as compared to the rest of the workloads. The EM driven GA approach again is effective in generating voltage-noise viruses. The EM virus has a V_{MIN} of 1.3625V, 37.5mV below the nominal voltage at 3.1GHz. It is interesting to point out that the EM based virus running on only two active cores is more severe than the AMD stability test and Prime95 on four active cores. To gain confidence in the V_{MIN} results we have run the AMD stability test and Prime95 for 24 hours at 1.287V and 1.28V respectively. They both pass the test whereas the EM virus causes immediate system-crash at 1.3V or even higher voltages.

8 DISCUSSION OF CROSS-PLATFORM FINDINGS

Sections 5, 6 and 7 demonstrated the successful application of the proposed EM-based approach for generating voltage-noise viruses and measuring the PDN first-order resonance frequency for three different CPU micro-architectures across two different ISAs. This section discusses cross-platform findings to provide insight on the generated viruses. In particular, we focus the discussion on the measured

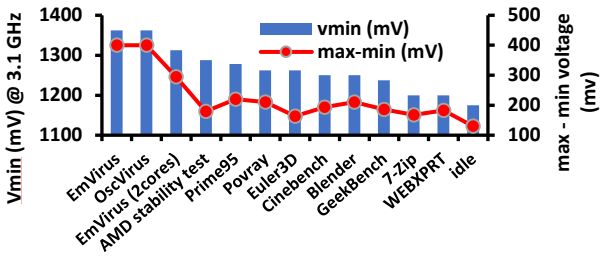

Figure 18. V_{MIN} and voltage noise measurements on the AMD CPU.

resonance frequencies, the potential for energy-efficiency improvements, the efficacy of the GA-optimization and the implications of instruction mix in virus generation.

Table 2 provides a comparison about the viruses generated by GA for the different platforms in terms of average instructions per cycle (IPC), instruction loop frequency, dominant frequency (the one where the highest EM amplitude is observed), voltage margin (the difference between the nominal voltage and virus V_{MIN}) and instruction-mix breakdown. Note that all viruses have a loop length of 50 instructions empirically found to work well for our optimization goals.

8.1 PDN Resonant Frequency and Voltage Margins

The first-order resonance frequency of processors is typically in the range between 50-200MHz [2][16] which is confirmed by our experimental results. The lowest resonance is observed at 66MHz (Cortex-A72, both cores powered) and the highest at 96MHz (Cortex-A53, with one core powered).

The analysis also quantifies the potential for energy-efficiency improvements through the elimination of voltage margins. Specifically, the viruses exhibit between 20 to 75mV higher V_{MIN} compared to standard benchmarks or previously proposed stress tests (e.g. Prime95) and, hence, can be used to determine better operating points. The Cortex-A72 and Cortex-A53 on the Juno platform can benefit considerably from margin elimination (the estimated V_{MIN} is at least 150mV lower than nominal voltage specifications [13]).

8.2 Dominant vs Loop Frequency in the GA Optimization

An interesting insight from Table 2 is that that the dominant frequency (at which highest voltage oscillations occur) does not equal the instruction loop frequency (1/loop period). All ARM CPU viruses have long loop periods that includes faster periodic events that stress the 1st order resonant frequency (e.g. a53em has 6 times slower loop frequency than dominant frequency). In contrast, the two viruses for the AMD CPU have equal dominant and loop frequencies.

We believe this difference is mainly due to CPU operating frequency. In particular, for the same number of instructions (50) it is easier for GA to construct a virus that has dominant

Table 2. dI/dt virus comparison. SL denotes short latency and LL denotes long latency. Voltage margin = Nominal voltage – VirusVmin.

Virus	Loop instructions	IPC	Loop period (ns)	Loop Freq (1/loop period) (MHz)	Dominant Freq (MHz)	Voltage margin (mV)	Instruction Type Mix							
							Branch (ARM Only)	SL int Register only	LL int Register only	SL int Mem (x86 only)	LL int Mem (x86 only)	Float	SIMD	MEM (ARM Only)
a72OC-DSO	50	1.45	30.43	32.86	65.73	150	4%	28%	10%	-	-	32%	18%	8%
a72em	50	0.74	60.00	16.67	66.66	150	0%	32%	8%	-	-	36%	18%	6%
a53em	50	0.69	81.17	12.32	74.95	150	0%	20%	8%	-	-	42%	24%	6%
amdEm	50	1.32	13.00	76.92	76.92	37.5	0%	24%	8%	30%	2%	10%	26%	-
amdOsc	50	1.35	12.69	78.8	78.8	37.5	0%	28%	8%	26%	2%	4%	32%	-

frequency equal to the resonant frequency when the CPU frequency is higher. Simply put, the faster the clock the lower it is the virus IPC needed for dominant and resonant frequencies to match. And, in general, it is difficult to construct sequences with high average IPC and yet with high and low power phases. For example, for Cortex-A72 the minimum IPC needed for dominant frequency to match resonant is nearly 3 whereas for AMD is 1.3 (minIPC= (resonant Frequency x Loop Instructions)/clock Frequency). Therefore, for the slower ARM CPUs the GA ends up constructing instruction sequences within the loop that cause periodic current fluctuations at a higher frequency compared to the loop-frequency itself. This underlines the effectiveness of the GA optimization to identify the 1st resonant frequency while fundamentally agnostic to CPU implementation (uArch, CPU frequency etc).

Moreover, the similarity of a72OC-DSO and a72em in dominant frequency and $V_{MIN,}$ despite the different IPC and loop period, underlines that there are multiple instruction sequences that can stress a CPU for voltage noise.

8.3 Virus Instruction Mix Breakdown

Table 2 shows the instruction breakdown of the viruses. All instruction types, apart from branch instructions are used in the instruction-mix of the viruses. Typically, a virus requires a combination of high-current and low-current-consuming instructions to create modulations in CPU current demand that can match the PDN's 1st order resonance frequency. Single-cycle instructions and those that engage the memory sub-system typically increase current consumption in the pipeline due to higher switching activity. The ARM viruses use plenty of short latency operations whereas the AMD viruses include many short latency integer instructions with operands in memory (denoted as SL-int-Mem).

Longer latency instructions are found in all the viruses as they create explicit pipeline stalls/interlocks that reduce current consumption. For stalling the SIMD/floating point functional units we have observed by code inspection that viruses tend to use long latency instructions like FSQRT (square root).

9 RELATED WORK

Previous work has exploited EM radiation for various objectives. EM emanations are a known security side channel for snooping information [9][11][15][63]. Other work leverages EM for non-malicious uses. In particular, [10] proposed non-obtrusive software profiling and [14] a malware detection scheme based on EM emanations. Our work also leverages EM radiation, but for addressing a different problem: voltage noise and PDN characterization in high-performance system-design. Other work [74] proposes architectural and compiler changes to reduce CPU EM interference.

Prior work has proposed various voltage margin elimination and voltage noise oscillation damping techniques. Adding capacitance helps in decreasing the voltage droop magnitude [56][58]. Some chips feature on-chip circuits that detect a voltage droop and react to it with a throttling mechanism [21][22][44][46] (e.g. adaptive clocking). Techniques based on voltage emergency signature prediction [30], destructive scheduling [1][57][60], on-die point-of-load regulation [47], micro-architectural throttling [54][55][56][61], recovery mechanisms [75] and dynamic determination of the available timing margin based on error correction or critical path monitor (CPM) [27][28][29][57][59] have also been proposed.

Representative dI/dt stress-tests are required to test the effectiveness of the above techniques and prior work has emphasized the need of post-silicon dI/dt stress tests for revealing PDN weaknesses and for voltage margin determination [2][31]. GA for dI/dt virus generation based on direct voltage measurements feedback is proposed in [2]. This work also uses GA for dI/dt virus generation, but based on EM emanations.

10 CONCLUSIONS AND FUTURE WORK

This work proposes a novel methodology for post-silicon dI/dt stress-test generation and resonant frequency detection based on sensing modulations in CPU EM emanations. The proposed approach has the advantage of being non-intrusive to system-software and does not incur design-time overheads and complexities. The basic premise for this methodology is

the presence of a correlation between the radiated EM power and on-chip voltage noise. The experimental analysis clearly establishes this correlation. Additionally, we demonstrate the generality of the proposed approach by successfully applying it to different CPUs to generate voltage-noise viruses for them and to obtain their PDN 1st order resonance frequency. The proposed EM based approach for quickly identifying resonant frequency can be particularly useful for validating pre-silicon simulation estimations with actual post-silicon product. The validation is desired for various reason e.g. for tampering detection.

For future work, we aim to extend our methodology to GPU PDNs, complementing recent studies on GPU voltage noise [18][19]. Other possible work directions are: a) secure-system design where on-the-fly PDN characterization can be utilized to thwart malicious side-channel attacks, b) development of an EM based PDN characterization procedure that is integrated in high-end products that can help improve their quality and energy efficiency, and c) voltage margin prediction based on EM emanations during conventional workload execution.

ACKNOWLEDGMENT

This work is funded by the H2020 Framework Program of the European Union through the UniServer Project (Grant Agreement 688540) – http://www.uniserver2020.eu. Part of this work has been conducted during an internship of the first author at ARM Research.

11 REFERENCES

[1] Reddi, Vijay Janapa, Svilen Kanev, Wonyoung Kim, Simone Campanoni, Michael D. Smith, Gu-Yeon Wei, and David Brooks. "Voltage noise in production processors." IEEE micro 31, no. 1 (2011): 20-28.

[2] Kim, Youngtaek, Lizy Kurian John, Sanjay Pant, Srilatha Manne, Michael Schulte, William Lloyd Bircher, and Madhu Saravana Sibi Govindan. "AUDIT: Stress testing the automatic way." In Microarchitecture (MICRO), 2012 45th Annual IEEE/ACM International Symposium on, pp. 212-223. IEEE, 2012.

[3] Polfliet, Stijn, Frederick Ryckbosch, and Lieven Eeckhout. "Automated full-system power characterization." IEEE Micro 31.3 (2011): 46-59.

[4] Ganesan, Karthik, and Lizy K. John. "MAximum Multicore POwer (MAMPO): an automatic multithreaded synthetic power virus generation framework for multicore systems." Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis. ACM, 2011.

[5] Whatmough, Paul N., Shidhartha Das, Zacharias Hadjilambrou, and David M. Bull. "14.6 An all-digital power-delivery monitor for analysis of a 28nm dual-core ARM Cortex-A57 cluster." In Solid-State Circuits Conference-(ISSCC), 2015 IEEE International, pp. 1-3. IEEE, 2015.

[6] Das, Shidhartha, Paul Whatmough, and David Bull. "Modeling and characterization of the system-level Power Delivery Network for a dual-core ARM Cortex-A57 cluster in 28nm CMOS." Low Power Electronics and Design (ISLPED), 2015 IEEE/ACM International Symposium on. IEEE, 2015.

[7] Reddi, Vijay Janapa, Meeta S. Gupta, Krishna K. Rangan, Simone Campanoni, Glenn Holloway, Michael D. Smith, Gu-Yeon Wei, and David Brooks. "Voltage noise: Why it's bad, and what to do about it." In 5th IEEE Workshop on Silicon Errors in Logic-System Effects (SELSE), Palo Alto, CA. 2009.

[8] Alam, M., B. Weir, and A. Silverman. "A future of function or failure?[CMOS gate oxide scaling]." IEEE circuits and devices magazine 18, no. 2 (2002): 42-48.

[9] Callan, Robert, Nina Popovic, Alenka Zajić, and Milos Prvulovic. "A new approach for measuring electromagnetic side-channel energy available to the attacker in modern processor-memory systems." In Antennas and Propagation (EuCAP), 2015 9th European Conference on, pp. 1-5. IEEE, 2015.

[10] Sehatbakhsh, Nader, Alireza Nazari, Alenka Zajic, and Milos Prvulovic. "Spectral profiling: Observer-effect-free profiling by monitoring EM emanations." In Microarchitecture (MICRO), 2016 49th Annual IEEE/ACM International Symposium on, pp. 1-11. IEEE, 2016.

[11] Genkin, Daniel, Lev Pachmanov, Itamar Pipman, and Eran Tromer. "Stealing keys from PCs using a radio: Cheap electromagnetic attacks on windowed exponentiation." In International Workshop on Cryptographic Hardware and Embedded Systems, pp. 207-228. Springer Berlin Heidelberg, 2015.

[12] Mitchell, Melanie. An introduction to genetic algorithms. MIT press, 1998.

[13] ARM, http://infocenter.arm.com/help/topic/com.arm.doc.100114_0200_03_en/arm_versatile_express_juno_r2_development_platform_(v2m_juno_r2)_technical_reference_manual_100114_0200_03_en.pdf

[14] Nazari, Alireza, Nader Sehatbakhsh, Monjur Alam, Alenka Zajic, and Milos Prvulovic. "EDDIE: EM-Based Detection of Deviations in Program Execution." In Proceedings of the 44th Annual International Symposium on Computer Architecture, pp. 333-346. ACM, 2017..

[15] Callan, Robert, Alenka Zajic, and Milos Prvulovic. "A practical methodology for measuring the side-channel signal available to the attacker for instruction-level events." Microarchitecture (MICRO), 2014 47th Annual IEEE/ACM International Symposium on. IEEE, 2014.

[16] Whatmough, Paul N., Shidhartha Das, Zacharias Hadjilambrou, and David M. Bull. "Power Integrity Analysis of a 28 nm Dual-Core ARM Cortex-A57 Cluster Using an All-Digital Power Delivery Monitor." IEEE Journal of Solid-State Circuits 52, no. 6 (2017): 1643-1654.

[17] Stutzman, Warren L., and Gary A. Thiele. Antenna theory and design. John Wiley & Sons, 2012.

[18] Thomas, Renji, Naser Sedaghati, and Radu Teodorescu. "EmerGPU: Understanding and mitigating resonance-induced voltage noise in GPU architectures." Performance Analysis of Systems and Software (ISPASS), 2016 IEEE International Symposium on. IEEE, 2016.

[19] Leng, Jingwen, Yazhou Zu, and Vijay Janapa Reddi. "GPU voltage noise: Characterization and hierarchical smoothing of spatial and temporal voltage noise interference in GPU architectures." High Performance Computer Architecture (HPCA), 2015 IEEE 21st International Symposium on. IEEE, 2015.

[20] Jordan, Edward C., and K. G. Balmain. "EM Waves & Radiating Systems." (2006).

[21] Grenat, Aaron, Sanjay Pant, Ravinder Rachala, and Samuel Naffziger. "5.6 adaptive clocking system for improved power efficiency in a 28nm x86-64 microprocessor." In Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International, pp. 106-107. IEEE, 2014.

[22] Ravezzi, Luca, and Hamid Partovi. "Clock and synchronization networks for a 3 GHz 64 Bit ARMv8 8-core SoC." IEEE Journal of Solid-State Circuits 50.7 (2015): 1702-1710.

[23] Joshi, Ajay M., Lieven Eeckhout, Lizy K. John, and Ciji Isen. "Automated microprocessor stressmark generation." In High Performance Computer Architecture, 2008. HPCA 2008. IEEE 14th International Symposium on, pp. 229-239. IEEE, 2008.

[24] National Instruments drivers, http://www.ni.com/downloads/drivers/

[25] AMD overdrive, https://www.amd.com/en/technologies/amd-overdrive

[26] DS-5 debugger, https://developer.arm.com/products/software-development-tools/ds-5-development-studio/ds-5-debugger/overview

[27] S. Das, "Razor: A Variation-Tolerant Design Methodology for Low-Power and Robust Computing", Doctoral Dissertation, University of Michigan, 2009.

[28] Ernst, Dan, Shidhartha Das, Seokwoo Lee, David Blaauw, Todd Austin, Trevor Mudge, Nam Sung Kim, and Krisztián Flautner. "Razor: circuit-level correction of timing errors for low-power operation." IEEE Micro 24, no. 6 (2004): 10-20.

[29] Lefurgy, Charles R., Alan J. Drake, Michael S. Floyd, Malcolm S. Allen-Ware, Bishop Brock, Jose A. Tierno, and John B. Carter. "Active management of timing guardband to save energy in POWER7." In Microarchitecture (MICRO), 2011 44th Annual IEEE/ACM International Symposium on, pp. 1-11. IEEE, 2011.

[30] Reddi, Vijay Janapa, Meeta S. Gupta, Glenn Holloway, Gu-Yeon Wei, Michael D. Smith, and David Brooks. "Voltage emergency prediction: Using signatures to reduce operating margins." In High Performance Computer Architecture, 2009. HPCA 2009. IEEE 15th International Symposium on, pp. 18-29. IEEE, 2009.

[31] Bertran, Ramon, Alper Buyuktosunoglu, Pradip Bose, Timothy J. Slegel, Gerard Salem, Sean Carey, Richard F. Rizzolo, and Thomas Strach. "Voltage noise in multi-core processors: Empirical characterization and optimization opportunities." In Microarchitecture (MICRO), 2014 47th Annual IEEE/ACM International Symposium on, pp. 368-380. IEEE, 2014.

[32] E. Alon, V. Stojanovic, and M. A. Horowitz, "Circuits and techniques for high-resolution measurement of on-chip power supply noise," J. SolidState Circuits, vol. 40, no. 4, pp. 820–828, Apr. 2005.

[33] Blender, https://www.blender.org/

[34] Cinebench, https://www.maxon.net/en/products/cinebench/

[35] Euler3d benchmark.www.caselab.okstate.edu/research/euler3dbenchmark.html

[36] http://www.principledtechnologies.com/ benchmarkxprt/webxpr

[37] GeekBench, https://www.geekbench.com/

[38] Prime 95, https://www.mersenne.org/download/

[39] Kim, Youngtaek, and Lizy Kurian John. "Automated di/dt stressmark generation for microprocessor power delivery networks." In Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design, pp. 253-258. IEEE Press, 2011.

[40] ARM V8 ISA, http://infocenter.arm.com/help/topic/com.arm.doc.den0024a/DEN0024A_v8_architecture_PG.pdf

[41] Oracle x86 assembly language reference manual, https://docs.oracle.com/cd/E18752_01/html/817-5477/docinfo.html

[42] big.LITTLE Whitepaper ARM, https://www.arm.com/files/pdf/big_LITTLE_Technology_the_Futue_of_Mobile.pdfbig.LITTLE Whitepaper ARM, https://www.arm.com/files/pdf/big_LITTLE_Technology_the_Futue_of_Mobile.pdf

[43] S. Pant, "Design and Analysis of Power Distribution Networks in VLSI Circuits", Doctoral Dissertation, University of Michigan, 2007.

[44] Kurd, Nasser A., Subramani Bhamidipati, Christopher Mozak, Jeffrey L. Miller, Timothy M. Wilson, Mahadev Nemani, and Muntaquim Chowdhury. "Westmere: A family of 32nm IA processors." In Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International, pp. 96-97. IEEE, 2010.

[45] Papadimitriou, George, Manolis Kaliorakis, Athanasios Chatzidimitriou, Dimitris Gizopoulos, Peter Lawthers, and Shidhartha Das. "Harnessing voltage margins for energy efficiency in multicore CPUs." In Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture, pp. 503-516. ACM, 2017.

[46] Fischer, Tim, Jayen Desai, Bruce Doyle, Samuel Naffziger, and Ben Patella. "A 90-nm variable frequency clock system for a power-managed itanium architecture processor." IEEE Journal of Solid-State Circuits 41, no. 1 (2006): 218-228.

[47] Mair, H.T., Gammie, G., Wang, A., Lagerquist, R., Chung, C.J., Gururajarao, S., Kao, P., Rajagopalan, A., Saha, A., Jain, A. and Wang, E., 2016, January. 4.3 A 20nm 2.5 GHz ultra-low-power tri-cluster CPU subsystem with adaptive power allocation for optimal mobile SoC performance. In Solid-State Circuits Conference (ISSCC), 2016 IEEE International (pp. 76-77). IEEE.

[48] http://download.intel.com/design/mobile/datashts/31407804.pdf

[49] http://support.amd.com/TechDocs/31412.pdf

[50] DeHaven, Keith, and Joel Dietz. "Controlled collapse chip connection (C4)-an enabling technology." In Electronic Components and Technology Conference, 1994. Proceedings., 44th, pp. 1-6. IEEE, 1994.

[51] Bockelman, David E., and William R. Eisenstadt. "Combined differential and common-mode scattering parameters: Theory and simulation." IEEE transactions on microwave theory and techniques 43, no. 7 (1995): 1530-1539.

[52] HSPICE circuit simulation tool, https://www.synopsys.com/verification/ams-verification/circuit-simulation/hspice.html

[53] Reddi, Vijay Janapa, and Meeta Sharma Gupta. "Resilient architecture design for voltage variation." Synthesis Lectures on Computer Architecture 8.2 (2013): 1-138.

[54] M.D. Powell and T. N.Vijaykumar. Pipeline muffling and a priori current ramping: architectural techniques to reduce high-frequency inductive noise. In Proc. International Symposium on Low Power Electronics and Design, 2003. DOI: 10.1145/871506.871562 69, 70

[55] M. D. Powell and T. N. Vijaykumar. Pipeline damping: A microarchitectural technique to reduce inductive noise in supply voltage. In Proc. International Symposium on Compute Architecture, 2003. DOI: 10.1145/871656.859628 70

[56] R. Joseph, D. Brooks, and M. Martonosi. Control techniques to eliminate voltage emergencies in high performance processors. In Proc. International Symposium on High-Performance Computer Architecture, 2003

[57] Zu, Yazhou, Charles R. Lefurgy, Jingwen Leng, Matthew Halpern, Michael S. Floyd, and Vijay Janapa Reddi. "Adaptive guardband scheduling to improve system-level efficiency of the POWER7+." In Proceedings of the 48th International Symposium on Microarchitecture, pp. 308-321. ACM, 2015.

[58] James, Norman, Phillip Restle, Joshua Friedrich, Bill Huott, and Bradley McCredie. "Comparison of split-versus connected-core supplies in the POWER6 microprocessor." In Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International, pp. 298-604. IEEE, 2007.

[59] Bacha, Anys, and Radu Teodorescu. "Using ECC feedback to guide voltage speculation in low-voltage processors." Proceedings of the 47th Annual IEEE/ACM International Symposium on Microarchitecture. IEEE Computer Society, 2014.

[60] Miller, Timothy N., Renji Thomas, Xiang Pan, and Radu Teodorescu. "VRSync: Characterizing and eliminating synchronization-induced voltage emergencies in many-core processors." In Computer Architecture (ISCA), 2012 39th Annual International Symposium on, pp. 249-260. IEEE, 2012.

[61] Powell, M. D., & Vijaykumar, T. N. (2004, June). Exploiting resonant behavior to reduce inductive noise. In Computer Architecture, 2004. Proceedings. 31st Annual International Symposium on (pp. 288-299). IEEE.

[62] Gupta, Meeta Sharma, Krishna K. Rangan, Michael D. Smith, Gu-Yeon Wei, and David Brooks. "Towards a software approach to mitigate voltage emergencies." In Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium on, pp. 123-128. IEEE, 2007.

[63] Callan, Robert, Alenka Zajić, and Milos Prvulovic. "FASE: finding amplitude-modulated side-channel emanations." Computer Architecture (ISCA), 2015 ACM/IEEE 42nd Annual International Symposium on. IEEE, 2015.

[64] Mair, H., Wang, E., Wang, A., Kao, P., Tsai, Y., Gururajarao, S., Lagerquist, R., Son, J., Gammie, G., Lin, G. and Thippana, A., 2017, February. 3.4 A 10nm FinFET 2.8 GHz tri-gear deca-core CPU complex with optimized power-delivery network for mobile SoC performance. In Solid-State Circuits Conference (ISSCC), 2017 IEEE International (pp. 56-57). IEEE.

[65] O'Mahony, F., 2013 February. Tutorial 6 - On-chip voltage and timing-diagnostic circuits. In Solid-State Circuits Conference (ISSCC), 2013 IEEE International. IEEE.

[66] Mansuri, M., Casper, B. and O'Mahony, F., 2012, June. An on-die all-digital delay measurement circuit with 250fs accuracy. In VLSI Circuits (VLSIC), 2012 Symposium on (pp. 98-99). IEEE.

[67] Xu, J., Hazucha, P., Huang, M., Aseron, P., Paillet, F., Schrom, G., Tschanz, J., Zhao, C., De, V., Karnik, T. and Taylor, G., 2007, February. On-die supply-resonance suppression using band-limited active damping. In Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International (pp. 286-603). IEEE.

[68] Sathe, V. and Das, S. Taming the Dark Horse: Voltage-Margin Minimization for Modern “Real-World” Energy-Efficient Computing. Tutorial in IEEE Design Automation Conference (DAC), Austin, TX, June 2016.

[69] Gu, J., Eom, H. and Kim, C.H., 2007, June. A switched decoupling capacitor circuit for on-chip supply resonance damping. In VLSI Circuits, 2007 IEEE Symposium on (pp. 126-127). IEEE.

[70] https://www.cadence.com/content/dam/cadence-www/global/en_US/documents/tools/ic-package-design-analysis/sigrity-systemsi-technology-ds.pdf

[71] https://www.apache-da.com/products/sentinel/sentinel-psi

[72] https://www.ansys.com/en-gb/products/electronics/ansys-hfss

[73] Bowman, Keith A., Carlos Tokunaga, Tanay Karnik, Vivek K. De, and James W. Tschanz. "A 22 nm all-digital dynamically adaptive clock distribution for supply voltage droop tolerance." IEEE Journal of Solid-State Circuits 48, no. 4 (2013): 907-916.

[74] Gorman, Daphne I., Matthew R. Guthaus, and Jose Renau. "Architectural opportunities for novel dynamic EMI shifting (DEMIS)." Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture. ACM, 2017.

[75] Gupta, Meeta S., Krishna K. Rangan, Michael D. Smith, Gu-Yeon Wei, and David Brooks. "DeCoR: A delayed commit and rollback mechanism for handling inductive noise in processors." In High Performance Computer Architecture, 2008. HPCA 2008. IEEE 14th International Symposium on, pp. 381-392. IEEE, 2008.