

# A Power-Efficient 32 bit ARM Processor Using Timing-Error Detection and Correction for Transient-Error Tolerance and Adaptation to PVT Variation

David Bull, Shidhartha Das, *Member, IEEE*, Karthik Shivashankar, Ganesh S. Dasika, *Student Member, IEEE*, Krisztian Flautner, *Member, IEEE*, and David Blaauw, *Senior Member, IEEE*

**Abstract**—Razor is a hybrid technique for dynamic detection and correction of timing errors. A combination of error detecting circuits and micro-architectural recovery mechanisms creates a system that is robust in the face of timing errors, and can be tuned to an efficient operating point by dynamically eliminating unused timing margins. Savings from margin reclamation can be realized as per device power-efficiency improvement, or parametric yield improvement for a batch of devices. In this paper, we apply Razor to a 32 bit ARM processor with a micro-architecture design that has balanced pipeline stages with critical memory access and clock-gating enable paths. The design is fabricated on a UMC 65 nm process, using industry standard EDA tools, with a worst-case STA signoff of 724 MHz. Based on measurements on 87 samples from split-lots, we obtain 52% power reduction for the overall distribution at 1 GHz operation. We present error rate driven dynamic voltage and frequency scaling schemes where runtime adaptation to PVT variations and tolerance of fast transients is demonstrated. All Razor cells are augmented with a sticky error history bit, allowing precise diagnosis of timing errors over the execution of test vectors. We show potential for parametric yield improvement through energy-efficient operation using Razor.

**Index Terms**—Adaptive design, dynamic voltage and frequency scaling, energy-efficient circuits, parametric yield, variation tolerance.

## I. INTRODUCTION

INTEGRATED circuits within microprocessors are operated with sufficient margins to mitigate the impact of rising variations at advanced process nodes. Margins are required to cope with process variation, power delivery network limitations [16]–[18], temperature fluctuations [17], lifetime degradation [13], [14], signal integrity effects and clock uncertainty. Inaccuracies in transistor models and EDA tools combined with measurement tolerances on the tester also contribute to

the overall level of uncertainty, and consequently drive up the margin requirements further still. While margins exist for the entire duration of the processor lifetime, they are only required for the worst-case combination of conditions that occur extremely rarely, if at all, in practice. Excess margins are essentially overheads that adversely impact both power and performance. Reducing excess margins is clearly beneficial, but this is both expensive and difficult without compromising on design integrity.

Table I classifies the various sources of variations according to their spatial reach and temporal rate-of-change. Based on their spatial reach, variations can be *global* or *local* in extent. Global variations affect all transistors on die such as inter-die process variations and ambient temperature fluctuations. In contrast, local variations affect transistors that are in the immediate vicinity of one another. Examples of local variations are intra-die process variations, local resistive (IR) drops in the power-grid and localized temperature hot-spots.

Based on their rate-of-change with time, variations can be classified as being *static* or *dynamic*. Static variations are essentially fixed after fabrication such as process variations, or manifest extremely slowly over processor lifetime such as ageing effects [13], [14]. Dynamic variations affect processor performance at runtime. Slow-changing variations such as temperature hot-spots and board-parasitics induced regulator ripple have kilo-hertz time constants. Fast-changing variations such as inductive undershoots in the supply voltage can develop over a few processor cycles [16], [18]. The rate and the duration of these  $Ldi/dt$  droops is a function of package inductance and the on-chip decoupling capacitance. Coupling noise and phase-locked loop (PLL) jitter are examples of local and extremely fast dynamic variations with duration less than a clock-cycle.

Traditional adaptive techniques [9]–[12], [16]–[24] based on canary or tracking circuits can compensate for certain manifestations of PVT variations that are global and slow-changing. These circuits are used to tune the processor voltage and frequency taking advantage of available slack. Tuning is limited to the point where delay measurements through the tracking circuits predict imminent processor failure. These circuits are limited by measurement uncertainty, the degree to which current and future events correlate and the latency of adaptation. Substantial margining for fast moving or localized events, such as  $Ldi/dt$ , local IR drop, capacitive coupling, or PLL jitter must

Manuscript received May 13, 2010; revised July 21, 2010; accepted September 12, 2010. Date of publication November 18, 2010; date of current version December 27, 2010. This paper was approved by Guest Editor Kazutami Arimoto.

D. Bull, S. Das, K. Shivashankar, and K. Flautner are with ARM Inc., Cambridge CB1 9NJ, U.K. (e-mail: dbull@arm.com, sdas@arm.com, karthik.shivashankar@arm.com, krisztian.flautner@arm.com).

G. S. Dasika and D. Blaauw are with the University of Michigan, Ann Arbor, MI 48109 USA (e-mail: gdasika@eecs.umich.edu, blaauw@umich.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSSC.2010.2079410

also be present to prevent potential critical path failures. These types of events are often transient, and while the pathological case of all occurring simultaneously is extremely unlikely in a real system, it is impossible to rule this out. Tracking circuits also incur significant calibration overhead on the tester to ensure critical path coverage over a wide range of voltage and temperature conditions. The delay impact of local variations and fast-moving transients worsens at advanced process nodes due to aggressive minimum feature lengths and high levels of integration. This undermines the efficacy of tracking circuits.

Razor [1]–[4] is a hybrid technique that addresses the impact of excess margins through dynamic detection and correction of timing errors. Razor exploits the key observation that worst-case variations occur extremely rarely in practice, by speculatively operating the processor without the full timing margins. Timing speculation incurs the risk of infrequent errors due to dynamic variations. Such errors are detected using specific circuits that explicitly check for late-arriving transitions at critical path endpoints, within a detection window around the rising clock-edge. The detection window is defined relative to the setup time, and is sufficient to detect transitions that occur either in or past the setup window.

Error detection can be done either by comparing two discrete samples [1], [2] or by using explicit Transition-Detector circuits that monitor throughout the detection window [3]–[6]. Both techniques introduce a minimum-delay constraint required to disambiguate between early transitions from the current cycle and late-transitions from the previous. This constraint is met by inserting delay-buffers on short-paths that intersect critical paths being monitored for timing errors. Error correction is performed by the system using either stall mechanisms with corrected data substitution [1], [2], or by instruction/transaction-replay [3]–[6]. A combination of *in situ* error-detecting circuits and micro-architectural recovery mechanisms creates a system that is robust in the face of timing errors.

Timing-error tolerance enables a Razor system to survive both local and fast-moving transient events, and adapt itself to the prevailing conditions, allowing excess margins to be reclaimed. Savings from margin reclamation can be realized as a per device power-efficiency improvement, or parametric yield improvement for a batch of devices. Improved power-efficiency results in a higher frequency of operation at the same supply voltage, without incurring the power impact of voltage overdrive. Alternatively, the same frequency of operation can also be sustained at a lower voltage. This leads to quadratic savings in dynamic power and exponential savings in leakage due to reduced short-channel effects (SCE).

Measurements performed on a simplified Alpha pipeline [3], [4] showed 33% energy savings by scaling the supply voltage to the point of first failure (PoFF) at extremely low error rates. In [5], the authors evaluated error-detection circuits on a 3-stage pipeline imitating a microprocessor, using artificially induced voltage droops and obtained 32% throughput gain at same supply voltage (VDD), or 17% VDD reduction at equal throughput. The authors extended this work to an open-RISC microprocessor core in [6] where *in situ* error-detecting sequentials (EDS) [5], [6] and Tunable Replica Circuits [7] are used in conjunction with micro-architectural recovery support to

achieve 41% throughput gain at equal energy or a 22% energy reduction at equal throughput.

In this paper, we apply Razor to an ARM-based processor that has timing paths representative of an industrial design, running at frequencies over 1 GHz, where fast-moving and transient timing-related events are significant. The processor implements a subset of the ARM instruction set architecture (ISA) and is fabricated on a UMC [15] 65 nm process, using industry standard EDA tools, with a worst case static timing analysis (STA) signoff of 724 MHz. Silicon measurements on 87 samples, including split lots, show a 52% power reduction of the overall distribution for 1 GHz operation. Error-rate driven dynamic voltage (DVS) and frequency scaling (DFS) schemes have been evaluated.

This work extends our previous research presented in [1]–[4] with the following innovations. (a) The micro-architecture is designed with explicitly balanced pipeline stages resulting in critical memory access and clock-gating enable paths, both of which are monitored using explicit Transition-Detectors. The micro-architecture responds to all timing errors by flushing the pipeline and re-executing from the next un-committed instruction. (b) A Transition-Detector design is presented with significantly reduced minimum-delay overhead. This design, described in Section II, operates with traditional 50% duty-cycle clocking and can be easily integrated in a traditional ASIC design flow. (c) All Razor standard-cells are augmented with a sticky error history bit that allows precise diagnosis of critical path timing failures over the course of execution of test-programs. (d) Parametric yield improvement through energy-efficient operation using Razor is demonstrated based on measurements from the test samples.

The remainder of the paper is organized as follows. Section II describes the design of the transition-detector that flags late-transitions at critical path endpoints. The micro-architectural design of the processor is described in Section III. We provide the chip implementation details in Section IV. Silicon results from dynamic voltage and frequency-scaling experiments are presented in Section V. Section VI deals with the total energy savings using Razor. Section VII evaluates the potential for parametric yield improvement using Razor-based per chip tuning. Finally, we summarize this paper in Section VIII and present concluding remarks.

## II. TRANSITION-DETECTOR CIRCUIT DESIGN

Fig. 1 shows the design of the Transition-Detector augmented to a rising-edge triggered master-slave flip-flop. We use a similar design of the Transition-Detector to monitor critical memory access paths and clock-gating enables. The Transition-Detector flags late-arriving transitions at the monitored net by generating a pulse in response to the transition and capturing it within a clock-pulse, generated from the rising-edge of the clock (CK).

The Transition-Detector incorporates two conventional pulse-generators for both rising and falling transitions on the D input. The pulse-generators use skewed devices sized such that the rising transition of the output pulse is favoured over the falling, thereby generating a wide pulse at the output of the pulse-generator. The width of the data-pulse is determined by the sizing of the pMOS transistors in the p-skewed inverters

TABLE I  
CLASSIFICATION OF VARIATIONS

		STATIC		DYNAMIC	
		EXTREMELY SLOW		SLOW-CHANGING	FAST-CHANGING
SPATIAL REACH	GLOBAL	Inter-die process variations		Package/Die VDD fluctuations	PLL jitter IR drop Ldi/dt
	LOCAL	Life-time degradation (NBTI, TDDB)		Ambient temperature variations	
		Intra-die process variations		Local IR Temperature hot-spots	Capacitive Coupling Clock-tree jitter
		TEMPORAL RATE OF CHANGE			

(with minimum-sized NMOS) and the nMOS transistors in the n-skewed NAND gates (with minimum-sized PMOS). The delay chain on the internal clock network defines an implicit clock pulse that is active when transistors N1 (enabled by CK) and N2 (enabled by nCK, the delayed and inverted version of CK) are both ON. The data-pulse can be captured when the clock-pulse is active by discharging the dynamic node, DYN, thereby flagging the ERROR signal. The ERN signal is generated during pipeline recovery initiated in response to the ERROR signal being flagged. It resets the Transition-Detector by precharging the dynamic node, DYN, and enabling it to capture subsequent timing errors. Thus, DYN is conditionally precharged only in the event of a timing error.

An additional RS-latch structure acts as a sticky error history (EHIST) bit that is set whenever an error occurs. The EHIST information is extremely useful for offline diagnostics since reading out the EHIST information allows precise identification of each Transition-Detector that triggered over the course of a test. The EHIST bit adds an additional 10% area and leakage overhead to the Transition-Detector. However, the diagnostic capability of the EHIST bit is required only during the initial development phase of a design and can be excluded in a production design.

Fig. 2 shows the conceptual timing diagrams that explain the principle of operation of the Transition-Detector. The implicit clock-pulse is active in the interval between the rising edge of CK and the falling edge of nCK. As mentioned previously, the width of the clock-pulse ( $T_{CK}$ ) and the width of the data-pulse ( $T_D$ ) are determined by the internal clock-network delay and the sizing of the pulse generators, respectively. Fig. 2(a) shows the effective error-detection window. The error-detection window begins (ends) when the trailing (leading) edge of the data-pulse overlaps with the leading (trailing) edge of the clock-pulse for duration greater than the minimum overlap ( $T_{OV}$ ) required for evaluating the dynamic node, DYN. Thus, the total error-detection window width ( $T_D + T_{CK} - 2T_{OV}$ ) is the aggregate of the data-pulse and the clock-pulse widths after adjusting for the minimum overlap required at the leading and the trailing edges.

The detection-window is fixed after design and needs to be adequate such that the delay-impact due to fast-moving phenomena can be detected and recovered from. Typically, the device widths in the pulse-generators are sized so as to minimize the total power overhead of detection while allowing sufficient

detection-window width. From simulation results, on the processor described in this paper, the generation of the error-detection window resulted in the total power overhead due to the Transition-Detectors to be 5.7% of the overall processor power at the typical corner (TT/1.0V/85C).

In order that metastability in the main flip-flop is suitably detected and flagged, the error-detection window needs to cover the setup window of the main flip-flop with sufficient margin, across PVT corners. Setup coverage is ensured by appropriately sizing the pulse-generators for a sufficiently wide data-pulse. Due to the added margin on the setup window, early transitions on the D input are now flagged as errors, even before they cause actual setup violations and state-upsets in the main flip-flop. This difference between the onset of the setup window and error-detection window, shown in Fig. 2(b), is a measure of the setup pessimism ( $T_{PES}$ ) that is inherent in this design. This pessimism ( $T_{PES}$ ) was measured on silicon to be  $\sim 5\%$  of the cycle time for 1 GHz operation, compared to the actual frequency where incorrect state starts to be latched.

“Q” can become metastable when the input “D” transitions in the setup window (the onset of which is marked by point B in Fig. 2(b)). However, this is reliably detected and flagged by the Transition-Detector since the error-detection window subsumes the setup window by design. The ERROR output of the Transition-Detector can become metastable due to a partial discharge of the node, DYN, at the onset of the error-detection window (marked by point A in Fig. 2(b)). However, since this occurs before the main flip-flop setup window, the output “Q” is guaranteed to transition to its correct state without any impact on its timing. Thus, metastability at the ERROR signal does not cause state corruption within the pipeline.

Although extremely unlikely, it is possible that a metastable ERROR output can potentially propagate to the pipeline recovery circuit. We address this in the conventional manner by ensuring that the ERROR signals are eventually double-latched within the pipeline before being processed by the recovery circuit. This is subsequently discussed in greater detail in Section III along with the micro-architectural description of the design.

The Transition-Detector imposes a minimum-delay constraint to prevent early transitions from being flagged as errors. The portion of the error-detection window that exists after the clock-edge determines the minimum-delay constraint. For this

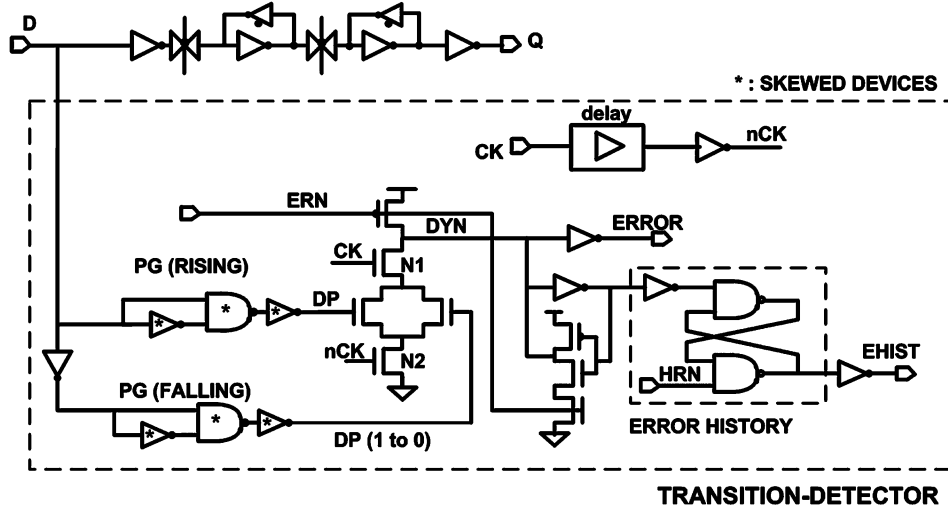


Fig. 1. Transition-Detector circuit schematic.

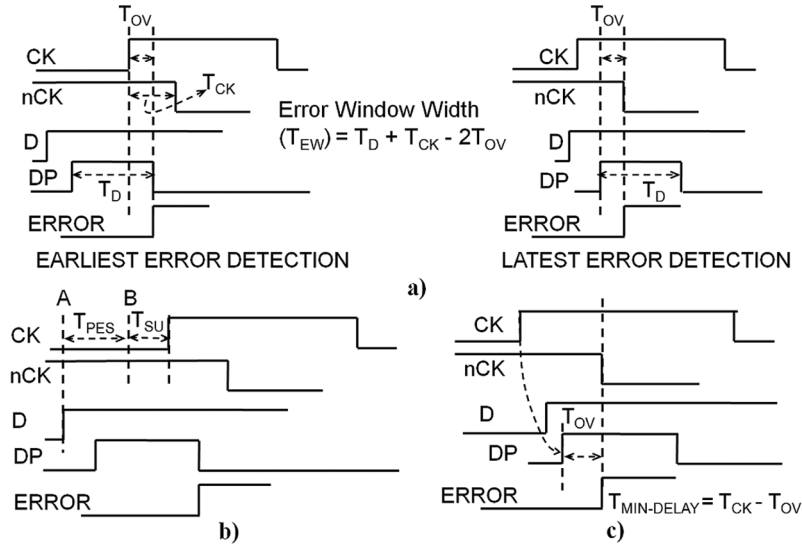


Fig. 2. Conceptual timing diagrams illustrating Transition-Detector operation a) Error-detection window is a function of the data-pulse and clock-pulse widths b) Flagging of early transition incurs a setup pessimism. c) Minimum-delay overhead is less than the clock-pulse width.

design of the Transition-Detector, the minimum-delay constraint is equivalent to the clock-pulse width after adjusting for the DYN evaluation delay, or  $T_{CK} - T_{OV}$ , as shown in Fig. 2(c). During design time, it is possible to adjust  $T_D$  and  $T_{CK}$  in order to trade-off performance penalty due to setup pessimism (determined by  $T_D$ ) for reduced minimum-delay constraint (determined by  $T_{CK}$ ). The minimum-delay constraint is met by the insertion of delay buffers on all short-paths that intersect with a critical path being monitored. This constraint for the Transition-Detector is expected to be significantly less than the high-phase of the clock used in previous designs [1]–[7]. For our processor, the power overhead of the delay buffers required to meet this constraint was 1.3% of the overall processor power at the typical corner (from simulation).

Using the high-phase of the clock as the error-detection window [1]–[7] requires a constant high-phase duration to be maintained to prevent minimum-delay violations. This requires

the generation and distribution of an asymmetric duty-cycle clock. Integrating the clock-pulse generator within the Transition-Detector precludes the need for phase truncation and conventional 50% duty-cycle clocking can be used. This makes the Transition-Detector easier to integrate in a conventional ASIC flow.

### III. MICRO-ARCHITECTURE DESIGN

The core micro-architecture is shown in Fig. 3. It is a conventional 6-stage in-order pipeline with fetch (FE), decode (DE), issue (IS), execute (EX), memory (MEM) and write-back (WB) stages. All the pipeline stages are explicitly balanced due to a combination of up-front micro-architecture design and path-equalization performed by back-end physical implementation tools, such that all stages have critical path endpoints of similar latency. The pipeline incorporates forwarding and interlock

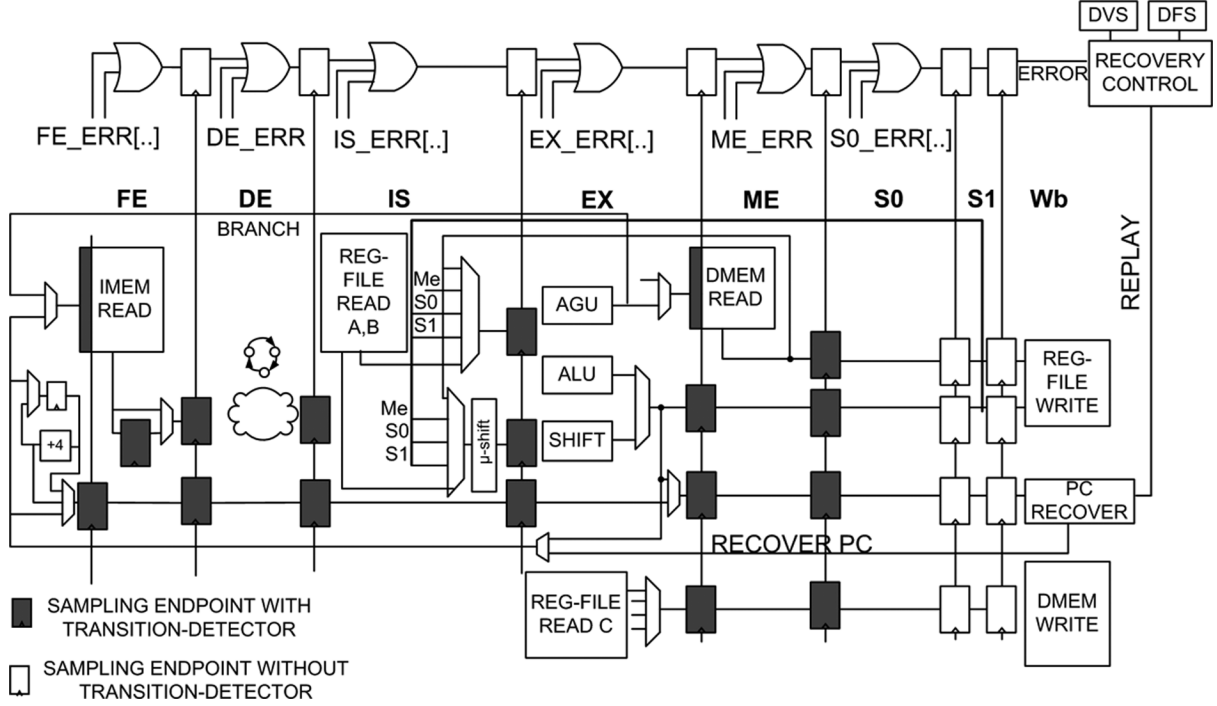


Fig. 3. Pipeline diagram of the ARM-based processor showing error-detecting Transition-Detectors and recovery control.

logic resulting in additional fanin to both dataplane and control paths.

Tightly-Coupled instruction and data memories (IRAM and DRAM), 2 KB each, hold 512 instruction and data words, respectively. As in commercial ARM microprocessor designs, the instruction and data-memory access paths are critical. Transition-detectors monitor the RAM interfaces and flag timing violations at the address and the chip-select pins during critical loads or instruction fetches. DRAM write accesses are required to be non-critical and this is guaranteed by suitably buffering store data, which eventually gets written into memory after Razor validation. Pipeline registers are aggressively clock-gated for low-power operation. Integrated Clock-gating Cells (ICGs) with critical enables are also augmented with Transition-Detectors.

The ERROR signals of individual stages are OR-ed together and registered to generate the stage error signal. This is then OR-ed with the ERROR signals from the subsequent stages and so on. The composite pipeline ERROR signal (Fig. 3) is double-latched to mitigate against potential metastability. Consequently, all instruction commits have to be postponed by two extra stabilization stages, S0 and S1, to budget for this synchronization overhead. Forwarding paths from S0 and S1 prevent pipeline interlocks and hence there is no Instruction Per Cycle (IPC) degradation due to these extra stages. From simulations performed under typical usage conditions, the power overhead due to S0 and S1 was 2.4% of the total processor power.

When an error is detected, the entire pipeline is flushed and the next un-committed instruction is replayed. Replay occurs at half-frequency such that a failing instruction does not incur repeated timing errors, thereby maintaining forward progress in the pipeline. Micro-architectural replay is a conventional technique that often already exists in high-performance pipelines

to support speculative execution such as out-of-order execution and branch-prediction. Therefore, it is possible to extend pre-existing recovery framework to support Razor timing speculation.

#### IV. CHIP IMPLEMENTATION DETAILS

Fig. 4 shows the die photograph of the processor. The processor implementation details are provided in Table II. The design is fabricated in UMC65SP [15] high-performance process with 1 V nominal supply voltage and 1.1 V as the overdrive limit. The STA sign-off frequency was 724 MHz measured at the worst-case corner (SS/0.9 V/125 C) where margins are budgeted for 10% voltage droop, slow silicon and temperature effects. We tested and measured 87 die from split-lots silicon with 30 samples from the fast (FF) lot, 37 from the typical (TT) and 20 samples from the slow (SS) lots, respectively. The prototype Razor processor is hosted on an ARM CPU sub-system as a memory-mapped peripheral. The ARM CPU is used as a test-harness for downloading code into the instruction memory through an APB [26] bus interconnect. The execution output from the general-purpose Register File (Fig. 3) and the Data Memory (DRAM) is then read-out at the end of every test and compared with a golden result set for correctness. The processor implements error rate driven dynamic frequency and voltage control (described in Sections V-A and V-B).

Out of a total of 2976 registers in the design, the top 503 most critical registers were augmented with a Transition-Detector for timing-error detection. This represents approximately 17% of the total flip-flops in the design. There are 149 ICG cells in the design, of which 27 have Transition-Detector protection. The Address and the Chip-Select pins of both the instruction and data memories are monitored using Transition-Detectors. In total, the design incorporates 550 Transition-Detectors. The timing-critical endpoints are chosen after timing analysis on a

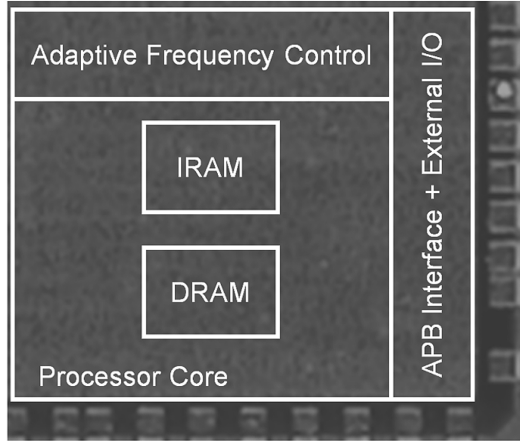


Fig. 4. Die photograph of the ARM-based Razor processor.

TABLE II  
PROCESSOR IMPLEMENTATION DETAILS

Flip-flops	2976
Flip-flops with TD	503 (17%)
ICGs	149
ICGs with TD	27
TD for RAMs	20
TD Power Overhead	5.7%
Power Overhead of Min-delay Buffers	1.3%
Stabilization Stages Power Overhead	2.4%
Total Power Overhead	9.4%
Total Area Overhead @ 70% utilization	6.9%
Measured Setup Pessimism of TD	5% @ 1GHz/1V
IRAM and DRAM size	2KB

placed-and-routed design at the slow corner. After identifying the critical path endpoints, the netlist is again taken through the implementation flow. The final design is then verified at multiple PVT corners to ensure that critical endpoints are always protected by Transition-Detectors across all corners.

A critical concern during implementation is that the design flow does not result in additional critical endpoints. Otherwise, the timing perturbation due to the incremental insertion of Transition-Detectors may lead to more timing endpoints to become critical, thus impacting design closure. We avoid multiple implementation iterations by imposing extra timing constraints on the non-critical endpoints during logic optimization and place-and-route. This ensures that the original list of critical paths is preserved and design closure is achieved.

Table II shows the total power and area overhead of Razor error detection and correction circuitry. From simulation results at the typical corner (TT/1 V/85 C), the total overhead of the 550 Transition-Detectors from simulation results was 5.7% with 1.3% overhead due to the delay buffers required to meet the minimum-delay constraint. The stabilization stages (S0 and S1)

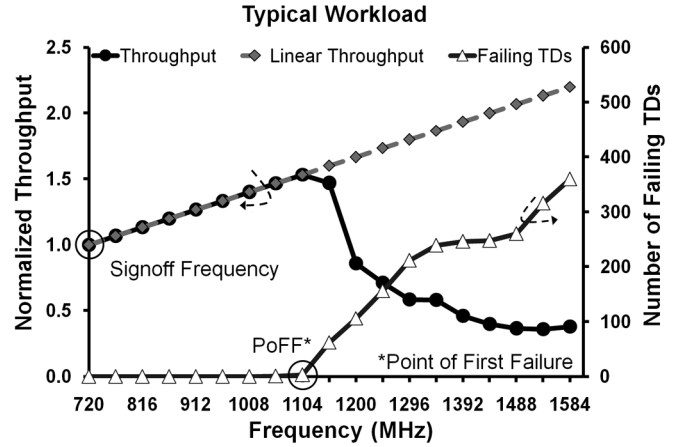


Fig. 5. Throughput versus frequency characteristics for the Typical workload at 1 V VDD on device TT9. Number of failing Transition-Detectors is also plotted against the secondary axis.

consume additional 2.4% power. Thus, the total power overhead due to Razor was 9.4% of the baseline processor. The combined Razor area overhead due to the Transition-Detectors, minimum-delay buffers and the stabilization stages was 6.9% of the total area, assuming 70% row utilization. Based on silicon measurements, the setup pessimism (Section II) of the Transition-Detectors was measured to be 5% of the cycle time for 1 GHz operation at 1 V nominal supply voltage.

## V. SILICON MEASUREMENT RESULTS

Fig. 5 shows the throughput versus frequency characteristics for a test-program executed on device TT9 at 1 V nominal VDD. This program (referred to as the Typical workload) computes the prime-factor decomposition of an array of integers and represents typical usage conditions. The throughput measured at each frequency point is normalized against the throughput at the sign-off frequency of 724 MHz. When execution completes, the EHIST information of individual Transition-Detectors is read out. The number of Transition-Detectors incurring timing errors is plotted as a function of frequency, against the secondary axis on Fig. 5.

In the absence of timing errors, the throughput increases linearly with frequency until the Point of First Failure (PoFF) at 1.1 GHz, a 50% throughput increase compared to the design point of 724 MHz. At the PoFF, there are four Transition-Detectors that incur timing errors. Thereafter, multiple failing Transition-Detectors contribute to a rapidly rising error rate due to the balanced nature of the pipeline. A combination of the rising error rate and the IPC overhead of recovery cause exponential degradation in the throughput. Consequently, it is desirable to limit operation to low-error rate regimes where the maximum benefits of energy-efficiency due to margin elimination can be claimed. Execution is correct until 1.6 GHz, after which recovery fails. This enables a safety margin of 500 MHz beyond the PoFF where the computation is still correct, albeit at an exponential loss in efficiency.

Fig. 6 shows the portion of the layout screenshot of the processor annotated with a map of failing Transition-Detectors (represented by black rectangles) for test programs executed

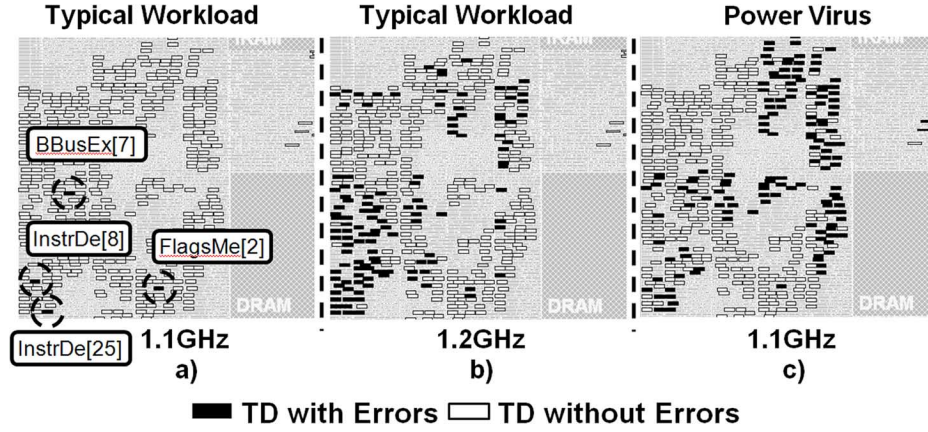


Fig. 6. Map of failing Transition-Detectors on chip TT9 at 1 V VDD: a) shows 4 failing Transition-Detectors for the Typical workload at 1.1 GHz. b) At 1.2 GHz, 122 Transition-Detectors incur timing failures indicating an increase in error rate. c) At 1.1 GHz, Power Virus workload has 249 failing endpoints compared to 4 for Typical.

on device TT9 at 1 V nominal VDD. Fig. 6(a) and (b) compares the failure map for the Typical workload at the PoFF (1.1 GHz) against that at 1.2 GHz. At 1.1 GHz, the 4 failing Transition-Detectors are in ID, EX and MEM stage buses, respectively. At 1.2 GHz, 122 Transition-Detectors fail timing. The failure map is dominated by the Transition-Detectors in the Instruction Decode bus located at the lower left-hand corner of the screenshot.

Fig. 6(a) and (c) compares the failure map for the Typical workload against that for a synthetic Power Virus workload, executed at the same operating point (1 V/1.1 GHz). The Power Virus workload is a loop of compute-intensive instructions that induces maximum on-chip activity leading to worst-case voltage droops (both IR and  $Ldi/dt$ ) in the power grid, while exercising the worst-case STA critical path. A combination of worst-case critical path sensitization and supply noise conditions causes 249 Transition-Detectors to incur timing errors compared to just 4 failures for the Typical workload. Furthermore, the failure map for the Power Virus workload is dominated by the EX stage bus located in the top right corner of the screenshot. Thus, there exists a significant variation in timing characteristics across workloads due to different critical paths being sensitized under varying voltage, temperature and noise conditions.

The failure maps in Fig. 6 illustrate the observability of critical path endpoints enabled by the EHIST information from the Transition-Detectors. This significantly enhances silicon testability since failing test-vectors can be precisely diagnosed by identifying critical path endpoints that incurred timing violations.

#### A. Razor-Based Dynamic Frequency Scaling

The adaptive frequency controller (AFC) (Fig. 4) exploits dynamic workload variations by tuning operating frequency in response to monitored error rates. In the adaptive mode, the processor clock (FCLK) is sourced from a 31-tap ring oscillator instead of the on-chip PLL. Coarse-grained frequency tuning is achieved by changing the ring-oscillator tap setting. Vernier-tuning is achieved in 24 MHz steps using a switched-capacitor ladder network. Hazard-free frequency adjustments occur during dynamic processor operation by constraining all clock-source and frequency selections to occur in the negative phase.

The pipeline error signal is double-latched to mitigate against potential metastability (Fig. 3) and accumulated in a 10 bit error register. During recovery, every alternate cycle is skipped such that the operating frequency is effectively halved, thus ensuring guaranteed forward progress within the pipeline. The frequency control algorithm is implemented in hardware and is externally programmable.

Fig. 7 shows the AFC response for a workload with three distinct phases consisting of loops of NOP, power virus and typical workloads running on device TT9 at fixed 1 V supply voltage. The AFC is programmed to reduce the operating frequency by 24 MHz for every cycle where a timing error is detected. The frequency is incremented by 24 MHz for every 1024 processor-cycles without timing errors.

The highest frequency is measured in the NOP phase (1.23 GHz). This is expected since the instruction mix is heavily dominated by lightweight NOP instructions that generate minimal switching activity within the pipeline. The most critical computations executed in the NOP phase are the address calculations for the branch instructions at the loop boundaries. When the workload transitions from the NOP to the Power Virus phase, the processor is able to survive this abrupt sensitization of worst-case critical paths, although the instantaneous throughput is impacted due to extremely high error rates. The AFC responds to the high error rate conditions by reducing the frequency in 24 MHz steps until the error rate stabilizes at approximately 1 GHz. Thus, the lowest frequency levels are measured in the Power Virus phase.

In the Typical phase, the AFC output shows 4 distinct frequencies between 1143 MHz and 1068 MHz, compared to just one for both the NOP and the Power Virus phases. This is due to paths of varying lengths being exercised during typical usage compared to relatively fixed-length paths for the synthetic NOP and Power Virus loops. The processor is able to sustain a maximum of 14% throughput gain for the Typical workload compared to the Power Virus loop.

The AFC response and the failure map experiments clearly indicate that by reclaiming worst-case margins, the device TT9 is capable of sustaining frequencies in excess of 1 GHz for most workloads, even though the actual design was signed off at 724 MHz. Hence, for the next Dynamic Voltage Control experiment,

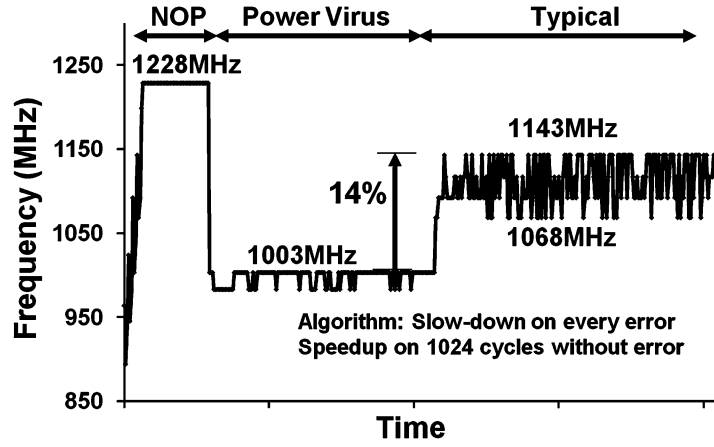


Fig. 7. Dynamic Frequency Scaling: AFC response for a 3-phase workload consisting of the NOP, Power Virus and Typical workloads at 1 V VDD. Frequency is increased or reduced in 24 MHz steps.

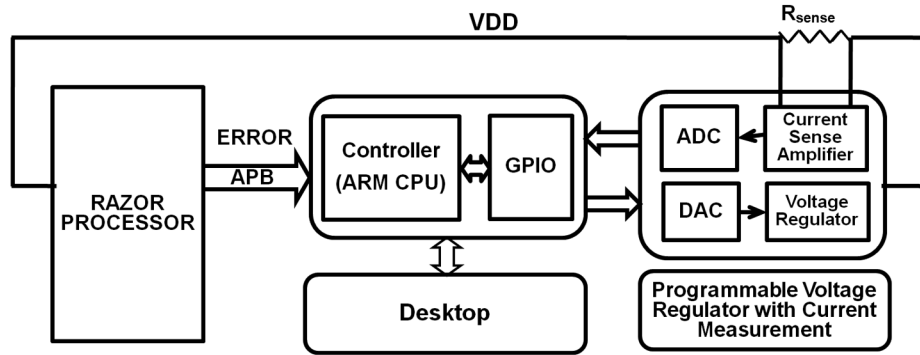


Fig. 8. Architecture of the closed-loop Razor voltage controller: The control algorithm is implemented in software running on an ARM1176. The supply voltage is set by programming an external regulator using a DAC.

we keep the frequency fixed at 1 GHz and vary the voltage as dictated by the error rates.

### B. Razor-Based Dynamic Voltage Scaling

Fig. 8 shows the architecture of the closed-loop controller implemented for dynamic voltage management based on measured error rates. The control algorithm is implemented in software on the ARM CPU that hosts the Razor processor sub-system. The voltage control decision is based upon the accumulated value of 100 samples of the on-chip error register, accessed through the APB bus interface. The supply voltage is adjusted by programming an external DC-DC regulator. The DC-DC regulator can source 800 mA current that is sufficient for the requirements of the Razor processor with maximum current consumption less than 150 mA. The response latency of the voltage control loop is measured to be 55  $\mu$ s.

The voltage controller response on device TT9 is shown in Fig. 9, for a three-phase program with loops of the NOP, Power Virus and Typical workloads, running at fixed 1 GHz frequency. The error rate for device TT9 is plotted against the secondary axis. The error rate is initially zero in the NOP phase since the supply voltage is higher than the PoFF for the relatively lightweight NOP instructions. The controller responds to the zero error rate by reducing the supply voltage to 0.92 V for device TT9, where infrequent timing errors occur. During the transition from the NOP to the Power Virus phase, the processor

experiences a surge in the error rate. The controller responds to the high error rate by increasing the supply voltage in proportional increments until the steady-state voltage is attained at 1.07 V. Conversely, the error rate drops to zero during the transition from the Power Virus to the Typical workload phase. The steady-state voltage for the Typical workload is achieved at 0.96 V.

The controller response for devices SS6 and FF5 are also plotted in Fig. 9. Device SS6 is amongst the slowest die out of the 87 devices while FF5 is amongst the fastest with maximum standby leakage. Thus, these devices represent the extremes of the distribution of devices. The steady-state voltage measured for the NOP, Power Virus and the Typical phases for each device in Fig. 9 is indicative of its native silicon-grade.

The dynamic voltage and frequency scaling experiments in Sections V-A and V-B illustrate how Razor maximizes the energy efficiency of the processor by tuning to the most efficient operating point depending upon specific workload requirements. *In situ* error detection and recovery enables the Razor processor to maintain correct operation in the presence of fast-changing dynamic variations and worst-case critical path sensitization. When dynamic variations persist, the Razor voltage controller automatically adapts to higher voltage levels so that low error rates are eventually achieved. In Section VI, we quantify the energy savings obtainable with Razor-enabled voltage tuning for 1 GHz operation.



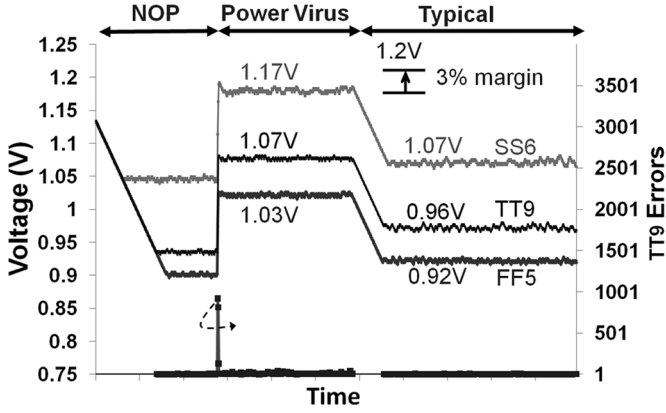


Fig. 9. Impact of process variations on Razor voltage controller response at 1 GHz frequency: Slowest device, SS6, requires the highest voltage and *vice versa* for the fastest device, FF5. SS6 requires 1.17 V for the Power Virus phase.

## VI. RAZOR ENERGY SAVINGS

From the Razor voltage controller response in Fig. 9, we observe that the slowest chip, SS6, requires a minimum voltage of 1.17 V in order to operate the Power Virus workload at 1 GHz frequency. For all our samples to operate correctly without Razor, sufficient margin is required to guarantee that the slowest device (SS6) operates correctly in the worst-case. Assuming Power Virus is the absolute worst-case code, then at a bare minimum additional margin must be added for temperature and safety. For 1 GHz operation, this translates to a worst-case voltage of 1.2 V for 3% margin. Thus, for conventional operation without Razor, the minimum required supply voltage is 1.2 V such that all die operate correctly at 1 GHz.

Fig. 10 compares the power consumption for Razor-enabled operation versus conventional operation at 1.2 V when executing the Typical workload at 1 GHz frequency for each of the three devices (FF5, SS6 and TT9). For the 1.2 V operation, leakage power is a significant contributor to the total power for the fastest device, FF5 (approximately 50%). The slowest device SS6 consumes the least power at 1.2 V due to low leakage. Even though the SS6 dynamic power is higher than that for FF5, the higher contribution of leakage causes FF5 to be the maximum power outlier for the entire distribution of devices.

With Razor-enabled voltage tuning, all devices operate at the PoFF for the Typical workload. The lower PoFF for FF5 (0.92 V) compared to that for SS6 (1.07 V) compensates for its higher leakage, leading to SS6 becoming the power outlier for the distribution. The maximum power consumption for Typical workload, considering all 87 devices, reduces from 100 mW for the baseline 1.2 V operation to 48 mW for operation with Razor. This represents a net 52% power saving at 1 GHz operation. On a per chip basis, power consumption on TT9 reduces from 71 mW at 1.2 V to 40.5 mW using Razor, a net 43% power saving due to Razor.

Fig. 10 compares Razor with a hypothetical, best-in-class adaptive technique. Adaptive techniques can be static where the supply voltage or the body bias is calibrated, margined and programmed on the tester [19]–[24]. Dynamic adaptive techniques [9]–[12], [16]–[18] rely on process and temperature

monitors to scale processor supply voltage according to prevailing PVT conditions at runtime. Due to adaptation latency, such techniques cannot respond in time to fast-changing voltage droops that manifest during abrupt processor activity changes (Fig. 9). At the minimum, margining is required to account for this latency as well as for measurement uncertainties inherent in the monitoring circuits. For our experiment, we assume a dynamic adaptive loop where voltage scaling is limited to the Power Virus voltage. Scaling voltage below this level can potentially cause incorrect execution if the processor undergoes a transition to the Power Virus workload. An additional 3% margin is added to account for measurement uncertainty.

Fig. 10 shows the power consumed for the Typical workload by the devices SS6, FF5 and TT9 using such a hypothetical best-case adaptive technique. This is a best-case comparison of the adaptive approach against Razor since only the minimum required margins are accounted for. Typically, margins for local temperature fluctuations and ageing effects are also added which will then lead to higher power consumption for the adaptive approach. The device TT9 requires 1.1 V using the best-case adaptive tuning and consumes 58.7 mW for the Typical workload. With Razor, TT9 consumes 40.5 mW that represents a 30% power saving due to Razor compared to the adaptive approach. The worst-case power, due to SS6, reduces by 25% using Razor.

Fig. 11 shows the power distribution for the 87 devices with Razor versus operation at 1.2 V and the best-case adaptive technique. The power distribution at constant 1.2 V VDD is dominated by the fast and leaky devices and therefore has large spread (37 mW). In contrast, the power distribution with Razor has a significantly narrower spread (10 mW) due to the equalization effect of a higher PoFF for the slower devices compensating for the higher leakage on the faster devices. The mean of the power distribution improves by 30 mW using Razor, a net 40% improvement over 1.2 V operation. Compared to best-case adaptive tuning, the mean of the distribution shifts by 14 mW (or 24%) when using Razor.

Sustained operation beyond the process overdrive limit of 1.1 V can have potential long-term gate-oxide reliability [14] and accelerated wear-out implications [13]. In addition, excessive overdrive exacerbates short-channel effects such as Drain Induced Barrier Lowering (DIBL) [24] leading to exponential increase in leakage, especially on the fast devices. From reliability and leakage considerations, it is desirable to limit the voltage overdrive to the process limit of 1.1 V.

SS6 requires at least 1.17 V when executing the worst-case Power Virus workload at 1 GHz. Hence, limiting the long-term overdrive operation to 1.1 V would necessarily require SS6 to be discarded when operating without Razor at 1 GHz frequency. Consequently, without Razor, operation at 1.1 V most certainly incurs a parametric yield loss for a frequency target of 1 GHz due to discarding the slow devices. In Section VII, we analyze the impact on parametric yield at 1 GHz when the maximum voltage for sustained, long-term operation is limited to 1.1 V.

## VII. PARAMETRIC YIELD IMPROVEMENT USING RAZOR

Any yield improvement technique cannot be quantitatively demonstrated with a small number of samples, however we can still illustrate the principle of how Razor can be used to improve

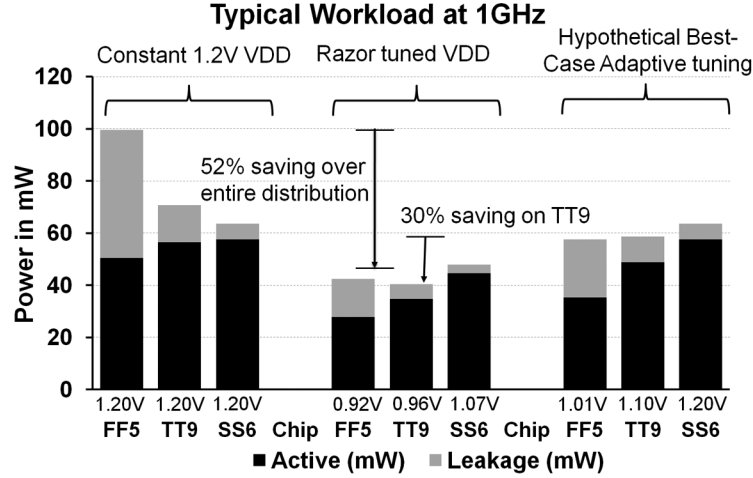


Fig. 10. Power consumption on devices FF5, TT9 and SS6: Razor is compared against constant 1.2 V VDD and best-case Adaptive-tuning. Razor enables 52% power saving overall compared to constant 1.2 V operation. Razor enables 30% power saving compared to best-case adaptive tuning for device TT9 and 43% saving compared to 1.2 V operation.

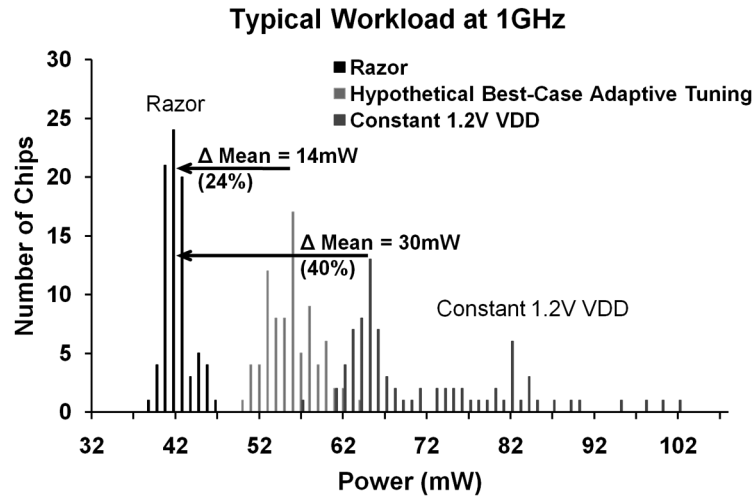


Fig. 11. Power distribution at the worst-case (WC) 1.2 V constant voltage operation versus Razor. Razor improves both the m and the s of the distribution.

the parametric yield for a distribution of devices. Functional devices are required to meet a targeted frequency specification ( $F_{max}$ ) under a given power budget ( $P_{max}$ ), before they can be shipped. In the following, we compare the parametric yield obtained using Razor versus that with conventional overdrive operation at constant 1.1 V VDD and an Adaptive Voltage Scaling (AVS) approach based on an on-chip Ring Oscillator serving as a process monitor. We have chosen the parametric yield targets of 1 GHz frequency under 65 mW power consumption for typical usage conditions.

#### A. Parametric Yield With Constant 1.1 V Overdrive

The scatter plot in Fig. 12 shows the total power consumption (dynamic and leakage) as a function of silicon-grade for all devices when executing the Typical workload at the 1.1 V/1 GHz operating point, without Razor. Operation without Razor requires margins for the worst-case. Assuming Power Virus to be the worst-case workload, we obtain the maximum frequency of operation for each die by measuring the Point of First Failure (PoFF) frequency (with 3% margin added for safety), when executing the Power Virus workload at 1.1 V VDD. Thus, the mea-

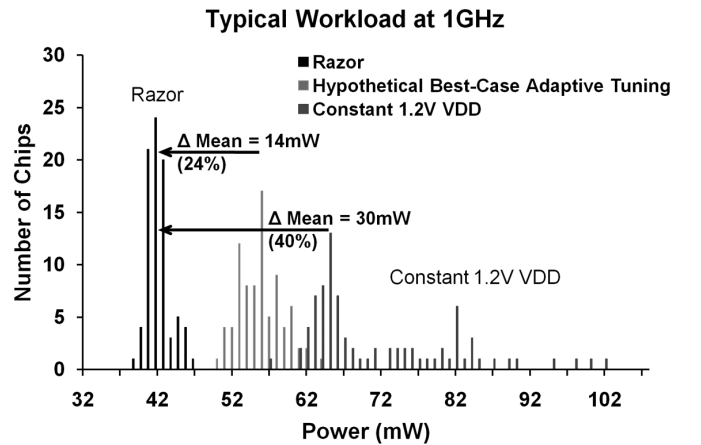


Fig. 12. Power at 1 GHz for Typical workload versus silicon-grade measured by highest frequency for correct operation without Razor at 1.1 V constant VDD. Measurements obtained on 87 die from split lots. Yield window is shown for frequency target ( $F_{max}$ ) of 1 GHz and power target ( $P_{max}$ ) of 65 mW.

sured PoFF for the worst-case Power Virus workload represents a margined frequency point under typical usage conditions.

The device, FF5, sustains the highest frequency (1127 MHz) for worst-case operation and consumes maximum power due to high leakage. Devices SS6 and TT13 from the slow and the typical lots respectively, are the slowest devices from our test samples and operate the Power Virus workload at 890 MHz. Thus, the devices follow an expected exponential trend with the fast devices with high leakage dominating the total power consumption compared to the slower devices.

Fig. 12 shows the parametric yield targets of frequency and power, labeled as “Fmax” and “Pmax” respectively. Out of 87 devices, there are 7 devices that exceed the 65 mW power criteria and 44 devices that fail the 1 GHz frequency criteria. Thus, there are 36 yielding devices (or 41% yield) out of a total of 87.

### B. Parametric Yield With Adaptive Voltage Scaling (AVS)

AVS techniques [9]–[12], [16]–[21] individually tune the supply voltage of devices according to their native speed-grade, based on delay measurements using on-chip process monitors. Per-device tuning compensates for inter-die process variations. However, extra margins are still required for fast-moving transients that are impossible to respond to in time. Such transients can trigger during abrupt processor transition from low-activity and non-critical operations to compute-intensive, heavyweight instructions. Consequently, for safe operation, AVS is required to be limited to a sufficiently margined point. We derive this safe operating limit based on the failure point for the Power Virus workload with added margin for safety (3%). In the absence of dynamic detection and correction of errors, the AVS technique cannot operate below this voltage due to potential risk of incorrect execution.

Our AVS measurements use an on-chip Ring Oscillator for estimating the worst-case processor delay. We obtain a statistical correlation function using a linear-fit model that relates the measured Ring Oscillator frequency at 1 V VDD to the minimum safe voltage requirement at 1 GHz. Due to the limited number of test devices, we measure the correlation function using data from every die. In the general case, a small number of samples from different global corners of the process distribution could be used as a training set to generate the correlation function for the entire distribution of devices.

In our measurements, we add margins to the linear-fit model only to account for possible under-estimation of the device voltage from the measured Ring Oscillator frequency. Discounting margins for temperature and ageing allows an optimistic comparison of AVS against Razor. Fig. 13 shows the scatter plot of the PoFF voltage for Power Virus workload versus the Ring-Oscillator frequency measured at 1 V for die from the fast (FF), slow (SS) and typical (TT) lots, respectively. The statistical correlation function for both data sets is computed to be 95.3% for the entire training set of devices. When measured across separate lots, this correlation is computed to be 86.2% for the FF lot, 85.1% for the SS lot and 89.1% for the TT lot, respectively. Due to the high correlation measured across global process corners, the Ring Oscillator frequency can be used to set the supply voltage for individual devices.

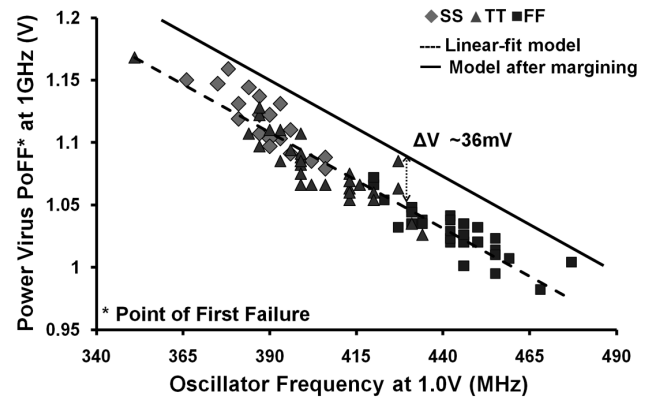


Fig. 13. Scatter-plot of the Power Virus PoFF (with 3% margin for safety) at 1 GHz versus measured Ring Oscillator frequency at 1.0 V: Ring-Oscillator frequency shows significant correlation (95%) with the minimum safe voltage, thus showing accurate tracking across global process corners. Extra margin added for voltage underestimation (36 mV).

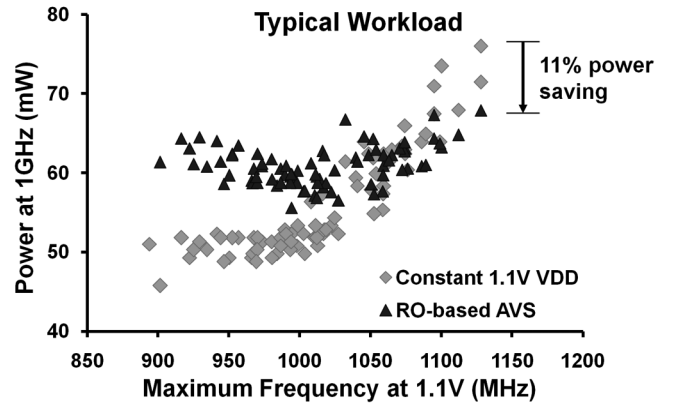


Fig. 14. Comparison of Ring-Oscillator based AVS with constant 1.1 V operation: Scatter-plot of power at 1 GHz running typical code for both techniques is shown as a function of silicon-grade. The maximum power reduces with AVS due to voltage scaling on fast chips. However, power increases on slow chips due to VDD exceeding 1.1 V limit leading to the U-shaped trend.

Fig. 13 shows the margining methodology for the Ring Oscillator based AVS. The device TT3 shows the maximum deviation from the linear-fit model, leading to a voltage underestimation of 36 mV. Consequently, this voltage difference has to be added as extra margin to the model to guarantee that the estimated voltage is always greater than the minimum voltage required for safe operation. This margin (36 mV) represents 3.2% of the nominal voltage overdrive of 1.1 V.

The scatter plot of Fig. 14 shows the power consumption of each die using the margined AVS model in Fig. 13 plotted against its native silicon-grade (maximum worst-case frequency of operation with 3% margin). The U-shaped trend of the scatter plot is a consequence of the power reduction on the faster devices due to lower voltage operation and *vice versa* for the slower devices. The maximum power consumption reduces from 76 mW at constant 1.1 V operation to 68 mW using AVS, a net 11% reduction in total power.

The voltage on the slow devices using AVS exceeds the 1.1 V overdrive limit. In addition, the extra 36 mV margin for voltage underestimation causes some of the typical devices to exceed

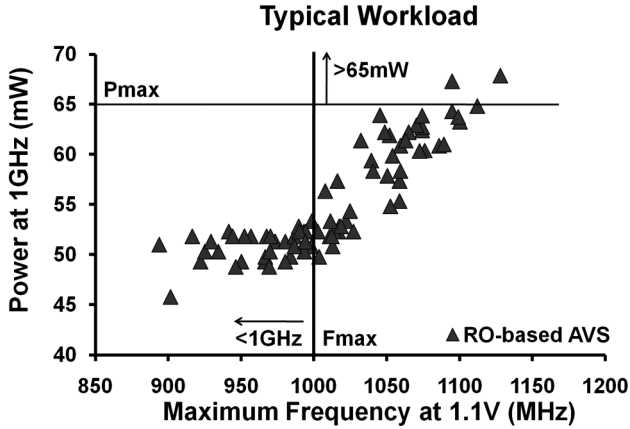


Fig. 15. Power versus silicon-grade scatter plot for AVS: Maximum supply voltage is limited to 1.1 V VDD when the AVS voltage exceeds this limit on the slow die. VDD tuning on the fast-die reduces the max power for the distribution which improves the power yield. However, frequency yield is unchanged.

the 1.1 V limit as well. Due to wearout and reliability concerns, we limit the maximum voltage to the process overdrive limit of 1.1 V for sustained, long-term operation. As a consequence, devices incapable of sustaining correct operation at 1.1 V are now discarded, leading to yield loss.

Fig. 15 shows the power versus silicon-grade scatter plot where maximum VDD is limited to 1.1 V. AVS leads to lower power consumption on the fast devices with the maximum power outlier at 68 mW. Excluding the 2 devices violating the maximum power constraint and the 44 devices fail the 1-GHz frequency constraint, there are now 41 yielding devices out of 87, or 47% yield.

### C. Parametric Yield With Razor

Fig. 16 shows the power versus silicon-grade scatter plot for Razor-enabled operation on 87 devices executing the Typical workload at 1 GHz frequency. The silicon-grade is again represented by the maximum frequency of operation, sustainable at constant 1.1 V VDD. Due to the elimination of worst-case margins using Razor, each device operates at a higher frequency when executing the Typical workload compared to the worst-case Power Virus workload. Therefore, the entire scatter plot shifts to higher frequency values. The slowest device, SS6, can execute the Typical workload at near zero error rate conditions at 1015 MHz at 1.1 V VDD, thus exceeding the 1 GHz frequency target. The highest PoFF for the Typical workload is measured to be 1397 MHz on device, FF76.

The maximum power outlier when using Razor is measured to be 48 mW which represents a 26% saving over the power target of 65 mW and a net 37% saving over the worst-case power (76 mW) at constant 1.1 V operation. Thus, all devices simultaneously meet both the power and frequency targets and 100% yield is achieved. The yield obtained for the 1 GHz/65 mW parametric targets using constant 1.1 V operation, AVS and Razor approaches is summarized in Table III.

A key observation here is that in case of Razor, the slowest device SS6 executes most workloads below the process limit

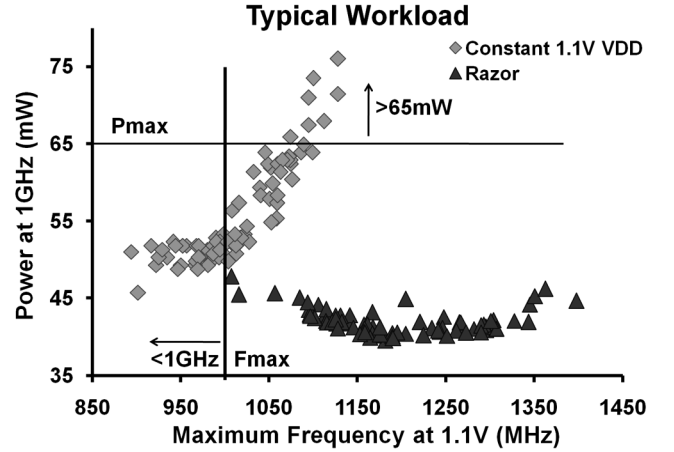


Fig. 16. Parametric yield with Razor compared against that with constant 1.1 V VDD operation. Frequency uplift through margin reclamation on slow devices and voltage scaling to PoFF for fast devices enables 100% yield through Razor.

TABLE III  
SUMMARY OF YIELD OBTAINED USING THE 3 DIFFERENT TECHNIQUES:  
CONSTANT 1.1 V OVERDRIVE, AVS AND RAZOR-TUNING FOR 87 TOTAL DIE

Freq	Technique	No. Chips at <1GHz Freq	No. Chips with >65mW power	Yield	% Yield
1008	Constant 1.1V	44/87	7/87	36/87	41%
	AVS	44/87	2/87	41/87	47%
	Razor	0	0	87/87	100%

of 1.1 V. Thus, for long-term operation the supply voltage is kept below 1.1 V for all devices, except for extremely rare use cases equivalent to the pathological worst-case Power Virus code. This is in contrast with the AVS approach where operation beyond 1.1 V is sustained on a long-term basis for the slower devices. Furthermore, safety margins and correlation uncertainties cause more devices to require greater than 1.1 V supply in the AVS approach compared to Razor.

For applications where the peak power consumption is a fundamental constraint, packaging and thermal limitations can impose absolute restrictions on the supply voltage from exceeding the 1.1 V VDD limit, even for the Power Virus workload. From our measurements, there are 22 devices out of 87 that require supply voltage in excess of 1.1 V for the Power Virus workload with Razor-enabled operation. Discarding these devices leads to 65 yielding devices (or 75% yield) when strict limits on the maximum voltage of operation are applied.

## VIII. SUMMARY AND CONCLUSION

In this paper, we presented the design of an ARM-based microprocessor that uses Razor for energy-efficient operation through the elimination of timing margins. With Razor-based voltage tuning, we achieved 52% energy savings at 1 GHz operation on a distribution of 87 devices from split-lots. We presented the design of a Transition-Detector with significantly reduced minimum-delay impact. The Transition-Detector relies on locally generated clock and data-pulses and can operate using conventional 50% duty-cycle clocking. Thus, it can be easily integrated into a conventional ASIC design flow.

We demonstrated the operation of dynamic frequency and voltage controllers that enable runtime adaptation to PVT variations and tolerance of fast transients through Razor error detection and recovery. The dynamic frequency controller was implemented in hardware on-chip and relies on a Ring-Oscillator clock-source to adjust frequency according to monitored error rates. The voltage controller was implemented in software running on a separate ARM processor that samples the error register through an APB bus interface and adjusts the voltage by programming an external voltage regulator.

Finally, we demonstrated the potential for parametric yield improvement using Razor. By trading margins for higher frequency on the slow devices and lower power on the fast devices, Razor-tuning enables more devices to meet the dual-sided parametric yield constraints of frequency and power. Further research is required to develop suitable manufacturing test methodologies before Razor can be deployed in the field. As process technology scales to ultra-small geometries, Razor mitigates the impact of rising variations by simultaneously enabling higher performance at lower power consumption.

#### ACKNOWLEDGMENT

The authors would like to thank staff at United Microelectronics Corporation (UMC) for providing, integrating, and fabricating the silicon, as well as D. Flynn, S. Idgunji, and J. Biggs at ARM for developing the "Uterior" technology demonstrator chip that hosts the Razor subsystem.

#### REFERENCES

- [1] S. Das *et al.*, "A self-tuning DVS processor using delay-error detection and correction," *IEEE J. Solid-State Circuits*, pp. 792–804, Apr. 2006.
- [2] D. Ernst *et al.*, "Razor: A low-power pipeline based on circuit-level timing speculation," in *Proc. IEEE Micro Int. Symp. Microarchitecture*, Dec. 2003, pp. 7–18.
- [3] D. Blaauw *et al.*, "RazorII: *in situ* error detection and correction for PVT and SER tolerance," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2008, pp. 400–401.
- [4] S. Das *et al.*, "Razor II: *in situ* error detection and correction for PVT and SER tolerance," *IEEE J. Solid-State Circuits*, pp. 32–48, Jan. 2009.
- [5] K. Bowman *et al.*, "Energy-Efficient and metastability-immune timing-error detection and instruction replay-based recovery circuits for dynamic variation tolerance," *IEEE J. Solid-State Circuits*, pp. 49–63, Jan. 2009.
- [6] J. Tschanz *et al.*, "A 45 nm resilient and adaptive microprocessor core for dynamic variation tolerance," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2010, pp. 282–283.
- [7] J. Tschanz *et al.*, "Tunable replica circuits and adaptive voltage-frequency techniques for dynamic voltage, temperature, and aging variation tolerance," in *IEEE Symp. VLSI Circuits Dig.*, Jun. 2009.
- [8] D. Bull *et al.*, "A power-efficient ARM ISA processor using timing-error detection and correction for timing-error tolerance and adaptation to PVT variations," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2010, pp. 284–285.
- [9] A. Drake *et al.*, "A distributed critical path timing monitor for a 65 nm high-performance microprocessor," *IEEE ISSCC Dig. Tech. Papers*, 2007.
- [10] J. Tschanz *et al.*, "Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2007, pp. 292–293.
- [11] M. Nakai *et al.*, "Dynamic voltage and frequency management for a low power embedded microprocessor," *IEEE J. Solid-State Circuits*, pp. 28–35, Jan. 2005.
- [12] K. Nowka, "A 32-bit PowerPC system-on-a-chip with support for dynamic voltage scaling and dynamic frequency scaling," *IEEE J. Solid-State Circuits*, Nov. 2002.
- [13] S. Rangan, N. Mielke, and E. Yeh, "Universal recovery behavior of negative bias temperature instability," in *IEEE Int. Electron Devices Meeting Dig.*, Dec. 2003, pp. 341–341.
- [14] A. M. Yassine *et al.*, "Time dependent breakdown of ultrathin gate oxide," *IEEE Trans. Electron Devices*, pp. 1416–1420, Jul. 2000.
- [15] UMC United Microelectronics Corporation [Online]. Available: <http://www.umc.com/>
- [16] T. Fischer *et al.*, "A 90-nm variable frequency clock system for a power-managed titanium architecture processor," *IEEE J. Solid-State Circuits*, pp. 218–228, Jan. 2006.
- [17] R. McGowen *et al.*, "Power and temperature control on a 90-nm titanium family processor," *IEEE J. Solid-State Circuits*, pp. 229–237, Jan. 2006.
- [18] N. James *et al.*, "Comparison of split- versus connected-core supplies in the POWER6TM microprocessor," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2007, pp. 297–298.
- [19] J. Kao, M. Miyazaki, and A. R. Chandrakasan, "A 175-mV multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *IEEE J. Solid-State Circuits*, pp. 1545–1554, Nov. 2002.
- [20] J. Tschanz *et al.*, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," *IEEE J. Solid-State Circuits*, pp. 1396–1402, Nov. 2002.
- [21] J. Tschanz *et al.*, "Effectiveness of adaptive supply voltage and body bias for reducing impact of parameter variations in low power and high performance microprocessors," *IEEE J. Solid-State Circuits*, pp. 826–829, May 2003.
- [22] G. Gammie *et al.*, "A 45 nm 3.5 G baseband-and-Multimedia application processor using adaptive body-bias and ultra-low-power techniques," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2008, pp. 258–259.
- [23] C. Neau and K. Roy, "Optimal body bias selection for leakage improvement and process compensation over different technology generations," in *Proc. Int. Symp. on Low Power Electronic Devices (ISLPED)*, Aug. 2003, pp. 116–121.
- [24] A. Hokazono *et al.*, "Forward body biasing as a bulk-si CMOS technology scaling strategy," *IEEE Trans. Electron Devices*, pp. 2657–2664, Oct. 2008.
- [25] R. Kaur *et al.*, "Unified subthreshold model for channel-engineered sub-100-nm advanced MOSFET structures," *IEEE Trans. Electron Devices*, pp. 2475–2486, Sep. 2007.
- [26] AMBA APB Bus Specification Reference Manual [Online]. Available: [http://arminfo/help/topic/com.arm.doc.ih0024c/IH0024C\\_amba\\_apb\\_protocol\\_v2\\_0\\_spec.pdf](http://arminfo/help/topic/com.arm.doc.ih0024c/IH0024C_amba_apb_protocol_v2_0_spec.pdf)



**David Bull** received the B.Sc. degree in computer science from Royal Holloway College, University of London, U.K., in 1991.

He is a consultant engineer at ARM Ltd., Cambridge, U.K. He joined ARM in 1995, and spent nine years working on various aspects of processor development including micro-architecture and circuits. He has worked on the ARM9 and ARM11 processor families processor, and was the design lead for the ARM926EJ-S. Since 2004 he has focused on research into advanced circuit and micro-architectural techniques, and has led the ARM RAZOR research project.



**Shidhartha Das** (S'03–M'08) received the B.Tech degree in electrical engineering from the Indian Institute of Technology, Bombay, India, in 2002 and the M.S. and Ph.D. degrees in computer science and engineering from the University of Michigan at Ann Arbor in 2005 and 2009.

His research interests include micro-architectural and circuit techniques for low-power and variability-tolerant digital IC design. Currently, he is a Staff Engineer working for ARM Ltd., Cambridge, U.K., in the Research and Development group.



**Karthik Shivashankar** received the B.E. degree in electronics and communications from The National Institute of Engineering, Mysore, India, in 2006 and the M.Sc. degree in microelectronics from University of Liverpool, U.K., in 2008.

His research interests include design methodologies for DVFS controller algorithms. Currently, he is working as an Engineer at ARM Ltd., Cambridge, U.K., in the Research and Development group.



**Ganesh S. Dasika** (S'01) received the B.S.E. degree in computer engineering from the University of Michigan at Ann Arbor, where he is now a Ph.D. student in the Department of Electrical Engineering and Computer Science.

His research interests mainly include designing and compilation for power-efficient, domain-specific processors. He is a student member of the IEEE.



**Krisztian Flautner** (S'96–M'01) received the Ph.D. degree in computer science and engineering from the University of Michigan at Ann Arbor, where he is currently appointed as a visiting scholar.

He is the Vice President of research and development at ARM. ARM designs the technology that lies at the heart of advanced digital products with more than fifteen billion processors deployed. He leads a global team which is focused on the understanding and development of technologies relevant to the proliferation of the ARM architecture. The group's activities

cover a wide breadth of areas ranging from circuits, through processor and system architectures to tools and software. Key activities are related to high-performance computing in energy-constrained environments.

Dr. Flautner is a member of the ACM and the IEEE.



**David Blaauw** (M'94–SM'07) received the B.S. degree in physics and computer science from Duke University, Durham, NC, in 1986, the M.S. degree in computer science from the University of Illinois, Urbana, in 1988, and the Ph.D. degree in computer science from the University of Chicago at Urbana-Champaign in 1991.

Until 2001, he was with Motorola, Inc., Austin, TX, where he was the Manager with the High Performance Design Technology Group. Since 2001, he has been on the faculty at the University of Michigan,

Ann Arbor, where he is currently a Professor. His work has focused on very large scale integration design with particular emphasis on ultralow power and high performance design. His current research interests include high-performance and low-power VLSI circuits, particularly addressing nanometer design issues pertaining to power, performance, and robustness.

Dr. Blaauw was the Technical Program Chair and General Chair for the International Symposium on Low Power Electronic and Design. He was also the Technical Program Co-Chair of the ACM/IEEE Design Automation Conference and a Member of the International Solid-State Circuits Conference (ISSCC) Technical Program Committee.