A Dynamic Voltage Scaled Microprocessor System

Thomas D. Burd, *Student Member, IEEE*, Trevor A. Pering, Anthony J. Stratakos, and Robert W. Brodersen, *Fellow, IEEE*

***Abstract*—A microprocessor system is presented in which the supply voltage and clock frequency can be dynamically varied so that the system can deliver high throughput when required while significantly extending battery life during the low speed periods. The system consists of a dc-dc switching regulator, an ARM V4 microprocessor with a 16-kB cache, a bank of 64-kB SRAM ICs, and an I/O interface IC. The four custom chips were fabricated in a standard 0.6-μm 3-metal CMOS process. The system can dynamically vary the supply voltage from 1.2 to 3.8 V in less than 70 μs. This provides a throughput range of 6–85 MIPS with an energy consumption of 0.54–5.6 mW/MIP yielding an effective energy efficiency as high as 26 200 MIPS/W.**

***Index Terms*—Adaptive processor, energy efficient, low power, variable voltage.**

I. Introduction

MICROPROCESSOR systems are found in a variety of portable electronic devices which span a broad range of performance with respect to throughput and energy consumption. To lower energy consumption, existing low-power design techniques generally sacrifice throughput [1]–[4]. For example, personal digital assistants (PDAs) have a much longer battery life than notebook computers, but deliver proportionally less throughput to achieve this goal. Reducing the supply voltage is an effective technique to decrease energy consumption, as it is a quadratic function of voltage; however, the delay of CMOS gates scales inversely with voltage, so this technique reduces throughput as well. This paper will describe a new design technique that dynamically varies the supply voltage to only provide high throughput when required. This technique can decrease the system's average energy consumption by up to 10x, without sacrificing perceived throughput, by exploiting the time-varying computational load that is commonly found in portable electronic devices.

Shown in Fig. 1 is an example of the microprocessor system's desired throughput (e.g., million instructions per second, or MIPS) as a function of time. The computational requirements can be considered to fall into one of three categories: compute-intensive, low-speed, and idle. Compute-intensive and short-latency tasks (e.g., video decompression, speech recognition, complex spreadsheet operations, etc.) utilize the full throughput of the processor. Low speed and long-latency tasks (e.g., text entry, address book browsing, playing music, etc.) only require a fraction of the full processor throughput to adequately run. Executing these tasks faster than the desired throughput rate has no discernible benefit. In addition, there are system idle periods because single-user systems are often not actively computing. The key design objective for the processor systems in these applications is to provide the highest possible peak throughput for the compute-intensive tasks while maximizing the battery life for the remaining low speed and idle periods.

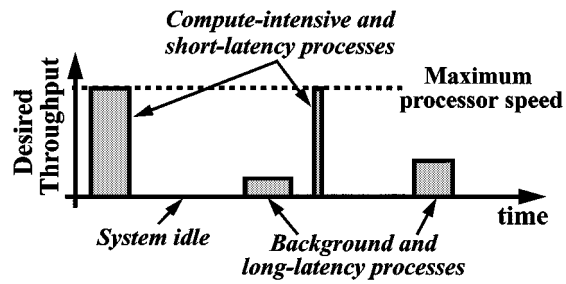

Fig. 1. Processor usage model.

A common power-saving technique is to reduce the clock frequency during non-compute-intensive activity [5]. This reduces power, but does not affect the total energy consumed per task, since energy consumption is independent of clock frequency to a first order approximation [6]. Conversely, reducing the voltage of the processor improves its energy efficiency, but compromises its peak throughput. If, however, both clock frequency and supply voltage are dynamically varied in response to computational load demands, then the energy consumed per task can be reduced for the low computational periods, while retaining peak throughput when required. When a majority of the computation does not require maximum throughput, then the average energy consumption can be significantly reduced, thereby increasing the computation that can be done with the limited energy supply of a battery. This strategy, which achieves the highest possible energy efficiency for time-varying computational loads, is called dynamic voltage scaling (DVS). This paper describes a prototype DVS-enabled chip-set which contains a voltage converter, a microprocessor, SRAM memory chips, and an interface chip for connecting to commercial I/O peripherals.

A technique for minimizing the supply voltage to reduce energy consumption utilizing a voltage regulator was proposed for digital circuits at fixed throughput [7], and subsequently demonstrated on a microprocessor core [8]. This idea was enhanced to dynamically scale the supply voltage for variable-rate digital signal processing [9], [10], and for variable-rate I/O interfaces

Manuscript received March 21, 2000; revised June 17, 2000. This work was supported by Defense Advanced Research Projects Agency and ARM Ltd.

T. D. Burd and R. W. Brodersen are with the Berkeley Wireless Research Center, University of California, Berkeley, CA 94704 USA (e-mail: burd@eecs.berkeley.edu).

T. A. Pering is with Intel Corporation, Hillsboro, OR 97124 USA.

A. J. Stratakos is with Volterra, Fremont, CA 94538 USA.

Publisher Item Identifier S 0018-9200(00)09434-8.

0018–9200/00$10.00 © 2000 IEEE

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

Petitioner Lenovo (United States) Inc. - Ex. 1036

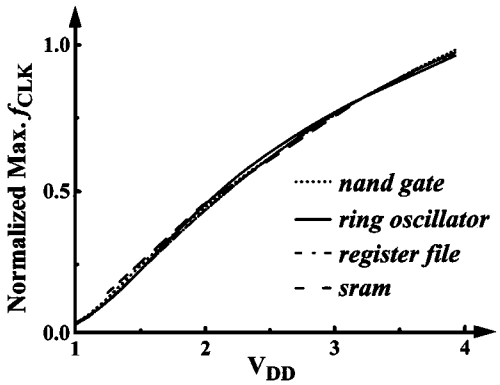

Fig. 2. Simulated maximum clock frequency for four circuits in 0.6-μm CMOS.

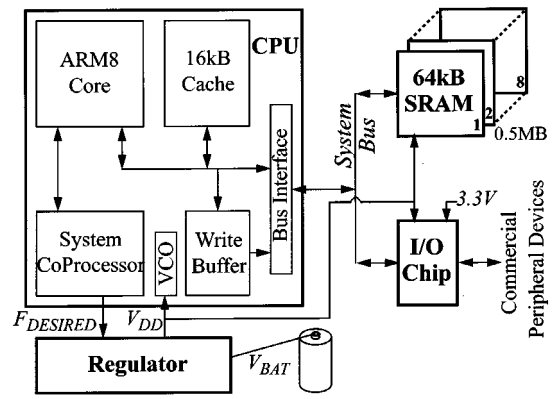

Fig. 3. System architecture—four custom chips.

[11]. This work extends these efforts to dynamic voltage supply scaling of a general-purpose microprocessor, under direct operating system control, and over a complete system chip-set.

II. DVS Overview

There are three key components for implementing DVS in a general-purpose microprocessor system: an operating system that can intelligently vary the processor speed, a regulation loop that can generate the minimum voltage required for the desired speed, and a microprocessor that can operate over a wide voltage range.

A critical characteristic of CMOS circuits is shown in Fig. 2, which plots simulated maximum clock frequency versus supply voltage for various circuits in our 0.6-μm CMOS process. Whether the circuits are simple (NAND gate, ring oscillator) or complex (register file, SRAM), their circuit delays track extremely well over a broad range of supply voltage. Thus, as the processor's supply voltage varies, all of the circuit delays scale proportionally making CMOS processor implementations amenable to DVS. However, subtle variations of circuit delay with voltage do exist and primarily effect circuit timing, as discussed in Section VI.

Control of the processor speed must be under software control, as the hardware alone cannot distinguish whether the currently executing instruction is part of a compute-intensive task or a non-speed-critical task. The application programs cannot set the processor speed because they are unaware of other programs running in a multitasking system. Thus, the operating system must control processor speed, as it is aware of the computational requirements of all the active tasks. Applications may provide useful information regarding their load requirements, but should not be given direct control of the processor speed.

As processor speed varies, so too must the supply voltage in order to optimize the energy consumption. However, the software is not aware of the minimum required supply voltage for a desired clock frequency since it is a function of the underlying hardware implementation, process variation, and operating temperature. A ring oscillator, which attempts to scale over voltage with the critical paths of the processor, provides the translation from supply voltage to operating frequency.

A conventional voltage regulation system samples the output voltage and compares it to an input reference voltage within a negative feedback loop. In this DVS system, the regulation loop was modified so that the output voltage drives the ring oscillator whose output clock signal can be readily converted to a digitally measured clock frequency. The operating system controls the loop by providing a desired clock frequency from which the measured clock frequency is subtracted to calculate the feedback error. This approach allows the software to directly set the operating frequency, and lets the hardware loop determine the minimum required supply voltage to meet this desired frequency.

DVS introduces two new performance parameters, transition time and transition energy. Transition time is defined as the duration required to alter the clock frequency and supply voltage. This time impacts both interrupt latency and wake-up latency when the system is in its lowest-energy sleep state. Transition energy is the extra energy consumption due to switching losses that is required to change the system supply voltage.

III. System Architecture

The complete microprocessor system is comprised of four custom chips, as shown in Fig. 3, all of which were designed for DVS to maximize system energy efficiency. The regulator chip, discussed in detail in Section V, converts the battery voltage (V_{BAT}) to the variable supply voltage (V_{DD}) which powers the microprocessor, the processor bus, the external memory bank, the I/O interface chip, and the front-end of the regulator. The four chips have been fabricated in a 0.6-μm 3-metal $V_T \approx 1$-V CMOS process.

The CPU chip, shown in Fig. 4, contains a custom implementation of an ARM8 processor core [12]. The core, which implements the ARM IV instruction set architecture, contains a five-stage scalar integer pipeline and an eight-word prefetch unit that performs static branch prediction. A 16-kB 32-way set-associative unified cache operates at the core clock rate. The cache contains 16 physical blocks in which a CAM tag array provides the line decoding for an SRAM data array which is logically organized into 32 lines with 8 words per line. A 12-element write buffer multiplexes address and data into a single register file and supports a variable number of data words per address to accommodate the store multiple registers (STM) instruction. A bus interface unit connects the CPU to the processor bus and contains a memory controller that provides all the signal gen-

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

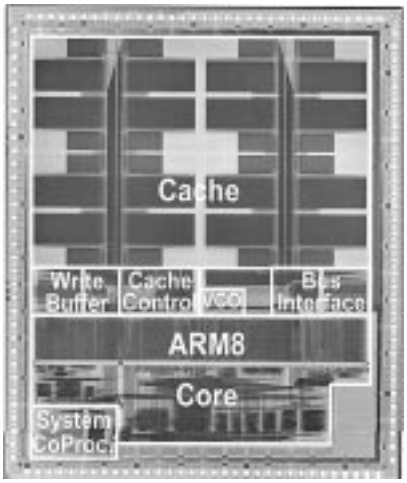

Fig. 4. CPU die photo (7.5 × 9.0 mm, 1.3 M transistors).

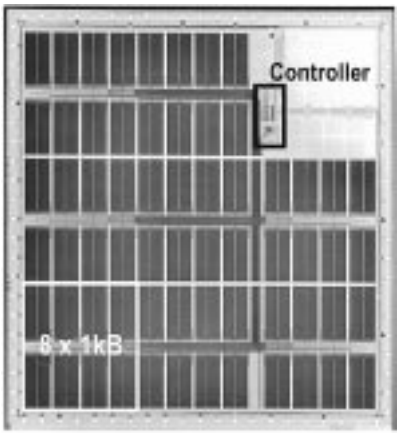

Fig. 5. SRAM die photo (9.6 × 10.4 mm, 3.4 M transistors).

eration for the external memory system. The bus side of the interface, the external bus, and the external memory system all operate at one-half the internal clock rate. At the center of the chip is the voltage-controlled oscillator (VCO) which drives an approximate H-tree clock network that has a maximum clock skew of 80 ps (simulated). The chip also contains a system coprocessor consisting of the desired clock frequency register, the regulator interface, a real-time counter, dynamic performance counters, and other system control state.

The SRAM chip, shown in Fig. 5, contains 64 kB of memory organized into two levels (8 × 8 × 1 kB). The data width of the chip is 32 bits so that only one external memory chip is activated per access, thereby minimizing the energy consumption of main memory. To prevent an excessive pin count, the address pins are multiplexed onto the same bus as the data. This reduces the bandwidth of single-word accesses by a factor of 2, but since external memory accesses are predominantly cache-line reloads, the average bandwidth reduction is closer to 11% (1 address word per 8 data words). The controller block on the SRAM chip supports these burst-mode accesses.

The primary function of the I/O chip is to convert the variable voltage processor bus to a fixed 3.3 V bus for communication with commercial peripheral devices. In addition, the I/O chip performs simple data flow control and supports direct memory accesses (DMA) from an I/O device. To facilitate testing, the entire I/O subsystem was emulated in hardware using a processor system board and an FPGA which connected directly to the I/O chip. This virtual I/O subsystem can stream data in regular intervals modeling real input devices, as well as collect and verify data destined for output peripheral devices. This emulation system allowed the execution of benchmark programs, typical of those run on PDA-like devices, to adequately demonstrate DVS.

IV. Voltage Scheduler

The voltage scheduler is a new operating system component for use in a DVS system. It controls the processor speed by writing the desired clock frequency to a system control register. The register's value is used by the regulation loop to adjust the CPU clock frequency and regulated voltage. By optimally adjusting the processor speed, the voltage scheduler always operates the processor at the minimum throughput level required by the currently executing tasks and thereby minimizes system energy consumption.

The implemented voltage scheduler runs as part of a simple real-time operating system. Since the job of determining the optimal frequency and the optimal task ordering are independent of each other, the voltage scheduler can be separate from the temporal scheduler. Thus, existing operating systems can be straightforwardly retrofitted to support DVS by adding in this new, modular component. The overhead of the scheduler is quite small such that it requires a negligible amount of throughput and energy consumption [13].

The basic voltage scheduler algorithm determines the optimal clock frequency by combining the computation requirements of all the active tasks in the system, and ensuring that all latency requirements are met given the task ordering of the temporal scheduler. Individual tasks supply either a completion deadline (e.g., video frame rate), or a desired rate of execution in MHz. The task's workload (e.g., processing an MPEG frame), measured in processor cycles, is automatically estimated by the voltage scheduler. While the optimal clock frequency in a single-tasking system is simply workload divided by the deadline time, a more sophisticated voltage scheduler is necessary to determine the optimal frequency for multiple tasks. Workload predictions are empirically calculated using an exponential moving average, and are updated by the voltage scheduler at the end of each task. Other features of the algorithm are a graceful degradation when deadlines are missed, the reservation of cycles for future high-priority tasks, and the filtering of tasks that cannot possibly be completed by a given deadline [14].

Fig. 6 plots V_{DD} for two seconds of a user-interface task, which generally has long-latency requirements. Since clock frequency increases with V_{DD}, processor speed can be inferred from this scope trace. The top trace demonstrates the microprocessor running in the typical full-speed/idle operation. A high voltage indicates the processor is actively running at full speed, and low voltage indicates system idle. This trace shows that the user-interface task has bursts of computation, which can be exploited with DVS. The lower trace shows the same task running

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

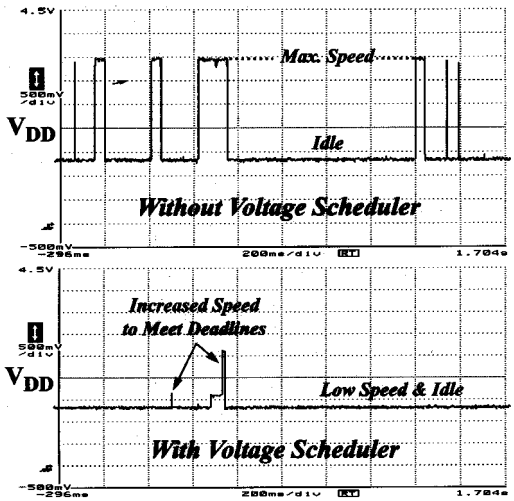

Fig. 6. DVS improvement for UI process.

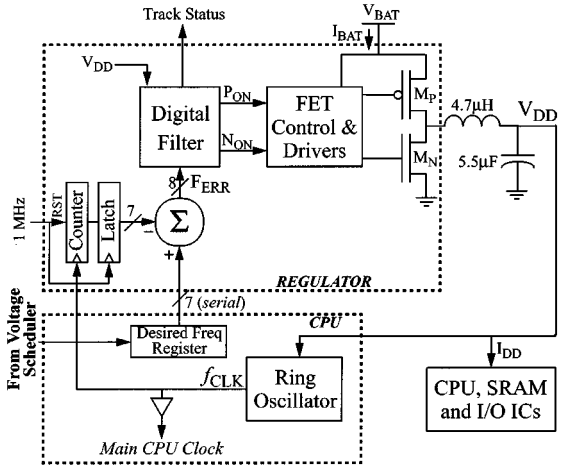

Fig. 7. Frequency to voltage feedback loop.

with the voltage scheduler enabled. In this mode, low voltage indicates both system idle and low-speed/low-energy operation. The voltage spikes indicate when the voltage scheduler has to increase the processor speed in order to meet required deadlines. This comparison demonstrates that much of the computation for this application can be done at low voltage, greatly improving the system's energy efficiency.

V. Voltage and Frequency Regulation Feedback Loop

The two-chip regulation feedback loop is shown in Fig. 7. The ring oscillator on the CPU chip outputs a clock signal whose frequency is a function of the supply voltage V_{DD}. The clock signal is sent to the regulator chip and drives a counter which is latched and reset at 1 MHz intervals to quantize the frequency into a 7-bit word. This value is subtracted from the desired frequency (in MHz) as set by the operating system, to create an 8-bit frequency error, $F_{\rm ERR}$. The loop filter circuit implements a hybrid pulse-width/pulse-frequency modulation (PWM/PFM) algorithm which generates signals to enable the power FETs via $P_{\rm ON}$ and $N_{\rm ON}$. The buck circuit consisting of M_P, M_N, and the *LC* tank down-converts $V_{\rm BAT}$ (3.3–6.0 V) to the regulated

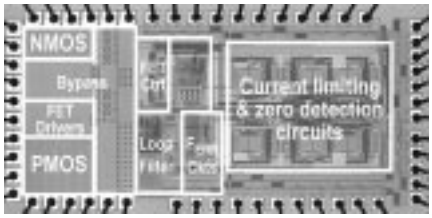

Fig. 8. Regulator die photo (1.6 × 3.4 mm).

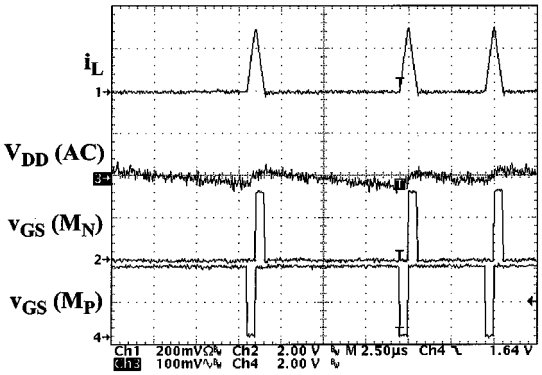

Fig. 9. Converter waveforms in regulation mode ($V_{DD} = 1.47$ V, $I_{DD} = 22$ mA).

voltage V_{DD}, which is fed back to the CPU chip, thus closing the loop.

The only external components required are a 4.7-μH inductor placed next to the regulator, and 5.5-μF of capacitance distributed near the chips' V_{DD} pins. The ring oscillator is placed on the CPU chip, and is designed to track the critical paths of the microprocessor over voltage. A beneficial side effect is that the ring oscillator will also track the critical paths over process and temperature variations. The rest of the loop is integrated onto the regulator die as shown in Fig. 8.

The regulation loop operates discontinuously to improve its stability and decrease the voltage transition time by pulsing the inductor current to transfer discrete quantities of charge to, or from, the capacitor. This is demonstrated in Fig. 9 which plots the buck circuit waveforms when the converter is regulating a constant V_{DD}, in which case it is adding charge to the capacitor. M_P is enabled first, which begins ramping up the inductor current (i_L) for a duration specified by the loop filter. At the end of the duration, M_P is turned off, and M_N is turned on, which ramps down i_L until it returns to zero. When the converter is ramping up V_{DD}, the i_L pulses will be larger and more frequent, and when it is ramping down V_{DD}, i_L will be reversed in polarity and the timing of the power FETs will switch so that M_N is enabled before M_P. Because i_L returns to zero at the end of each pulse, the inductor's current flow is not continuous, and the two-pole *LC* filter reduces to a single dominant pole which is set by the capacitor and the effective load resistance (V_{DD}/I_{DD}). Loop stability is then ensured by setting the gain-bandwidth of the loop to be well below the sampling frequency of 1 MHz over the range of V_{DD} and I_{DD}.

The regulator chip contains two additional components to support this discontinuous mode of operation. Current limiting circuits restrict the maximum output current to 1 A to protect the power FETs and external filter elements. These circuits consist

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

of two offset-cancelled comparators, one for each of the power FETs. Zero-detection circuits, implemented as offset-cancelled comparators, are required to detect when the rectifying power FET's current crosses zero, so that the FET can be turned off at the end of the charge pulse [15]. Additionally, to minimize power dissipation due to detector inaccuracy, an integral feedback loop, similar to adaptive dead-time control [16], is used to null the comparator, logic, and power FET gate-drive delays. This loop can detect the zero current crossing to within 2 mA.

A. Tracking Performance

The voltage converter required for DVS is fundamentally different from a standard voltage regulator because in addition to regulating voltage for a given speed, it must also change the voltage when a new speed is requested. A large regulator output capacitance reduces the dominant pole frequency, thereby reducing supply ripple, and increases low-voltage conversion efficiency, making the loop a better voltage regulator. A small capacitance reduces transition time and energy, making the loop a better voltage tracking system. Hence, the fundamental trade-off in DVS system design is to make the processor more tolerant of supply ripple so that the capacitance can be reduced to minimize transition time and energy [17]. The peak-to-peak ripple constraint for this system was relaxed to 5%, with a maximum measured value of 3.8%.

To further improve transition speed, the loop filter utilizes feed-forward control. F_{ERR} is first multiplied by a gain term, then a feed-forward value is added to it which is solely a function of the desired clock frequency. A 16 $\times$ 16-bit SRAM contains the look-up table for the feed-forward value as well as the frequency-dependent gain term, and is indexed by the upper four frequency bits. The feed-forward provides quick, but approximate loop adaptation, while the feedback loop locks onto the desired clock frequency.

B. Optimizing Conversion Efficiency

The key design challenge of this loop was to maintain good conversion efficiency with over 100x variation in load power dissipation, while keeping the output capacitance sufficiently small to maintain the loop's tracking performance. A hybrid PWM/PFM algorithm is utilized which combines the high efficiency that PWM can provide at high loads with the high efficiency that PFM can provide at low loads [15].

The converter operates in one of two modes, tracking and regulation. Tracking mode is initiated by a new frequency request. Charge is either delivered to, or removed from the capacitor depending upon the sign of F_{ERR}, and is delivered via a variable-width pulse which has 4 bits of control. When the error magnitude is less than 4 MHz, the converter switches to regulation mode in which the converter will still deliver energy to the output when F_{ERR} is greater than zero, but only the microprocessor system can remove charge. When F_{ERR} is less than zero, the loop filter is disabled to suppress the charge pulse that would otherwise remove charge and drive F_{ERR} to zero in a strictly PWM system. Thus, the only part of the loop that is continuously running is the front-end which calculates F_{ERR}. To improve low-voltage conversion efficiency, these circuits are powered by the variable V_{DD}, while the rest of the chip is powered by V_{BAT}.

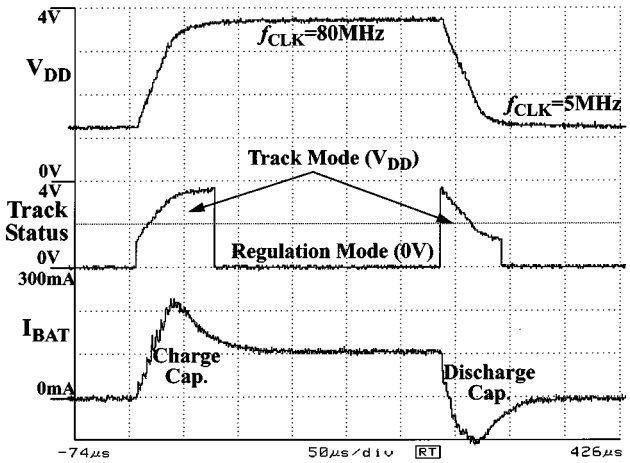

Fig. 10. Transient response of the regulation loop.

To minimize the sum of on-state and conduction losses, there is an optimum power FET gate width for fixed load current [16]. Since the load current varies by 50x, the power FETs are dynamically sized to minimize losses over a broad range of load current and maximize conversion efficiency. The filter's SRAM look-up table also contains two bits for each power FET for independent, binary-weighted, sizing control. The gate-width of the nMOS and pMOS least-significant bits (LSB) are 10 and 20 mm, respectively.

C. Transient Loop Response

Fig. 10 shows a scope trace for the system's maximum low-to-high and high-to-low speed transitions. The V_{DD} signal transitions from 1.2 to 3.8 V, then back down to 1.2 V. The Track Status signal indicates whether the loop is operating in the tracking or regulation mode. This signal demonstrates that the maximum transition time is 70 μs for the 5–80 MHz transition under full system load, while smaller voltage transitions are executed in less time. During this entire transition period, the processor system can continue to execute instructions.

The signal labeled I_{BAT} is the battery current measured going into the regulator, but after the battery's bypass capacitor. There is a current spike on a low-to-high transition which is required to charge up the loop's output capacitor to the required voltage. The negative current spike on the high-to-low transition occurs because the power pMOS is removing charge from the capacitor and placing it back onto the battery's bypass capacitor. The conversion loss of the regulator while charging and discharging the output capacitor becomes the transition energy, and is proportional to the size of the capacitor. This transition energy is a maximum of 4 μJ for a 5–80 MHz transition, which equals the energy consumed by the processor running at 80 MHz for 712 full-load cycles.

VI. Digital Circuit Design for DVS

One approach to designing a processor system that switches voltage dynamically is to halt processor operation during the

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

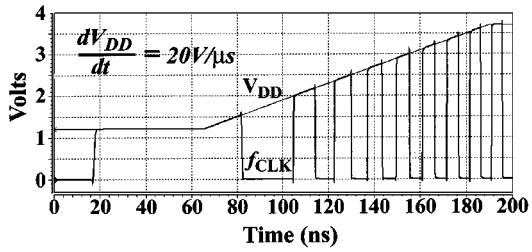

Fig. 11. Ring oscillator adapting to varying V_{DD} (simulated).

switching transient. The drawback to this approach is that interrupt latency increases and potentially useful processor cycles are discarded. Since static CMOS gates are quite tolerable of a varying voltage supply, there is no fundamental need to halt operation during the transient. When the gate's output is low, it will remain low independent of V_{DD}. However, when the output is high, it will track V_{DD} via the pMOS device(s). Simulation demonstrated that for a minimum-sized pMOS device in our 0.6-μm process, the *RC* time constant of the pMOS drain-source resistance and the load capacitance is a maximum of 5 ns, at low voltage. Thus, static CMOS gates track quite well for a dV_{DD}/dt in excess of 100 V/μs. Because all logic high nodes will track V_{DD} very closely, the circuit delay will instantaneously adapt to the varying supply voltage. Since the processor clock is derived from a ring oscillator also powered by V_{DD}, its output frequency will dynamically adapt as well, as demonstrated in Fig. 11.

Yet, there are constraints when using a design style other than static CMOS as well as limits on allowable dV_{DD}/dt. The processor system design contains a variety of different styles, including not only static CMOS logic, but dynamic logic, CMOS pass-gate logic, memory cells, sense-amps, bus drivers, and I/O drivers. As will be shown, the maximum dV_{DD}/dt that the circuits in this design can tolerate is approximately 5 V/μs. The converter loop has a maximum dV_{DD}/dt of only 0.2 V/μs, providing sufficient design margin. These design constraints sacrifice a small amount of energy-efficiency in the circuit design, but return much larger gains at the system level via DVS.

A. Pass Gate Logic

NMOS pass gates are often used in low-power design due to their small area and input capacitance. However, they are limited by not being able to pass a voltage greater than $V_{DD}-V_{Tn}$, such that a minimum V_{DD} of $2 \cdot V_T$ is required for proper operation. Since throughput and energy consumption vary approximately by 4x over the voltage range V_T to $2 \cdot V_T$, using nMOS pass gates restricts the range of operation by a significant amount, and are not worth the moderate improvement in energy efficiency. Instead, CMOS pass gates, or an alternate logic style, should be utilized to maximize the voltage range for DVS.

B. Dynamic Logic

Dynamic logic styles are often preferable over static CMOS as they are more efficient for implementing complex logic functions. They can be used with a varying supply voltage, but require some additional design considerations. One failure mode can occur while the circuit is in the evaluation state and the gate inputs are low such that the output node is undriven at a value V_{DD}. If V_{DD} ramps down by more than a diode drop by the end of the evaluation state, the drain-well diode will become forward biased. Current may be injected into the parasitic p-n-p transistor of the pMOS device and induce latchup [18]. This condition occurs when

$$\frac{dV_{DD}}{dt} \leq \frac{-V_{BE}}{t_{\mathrm{CLK|AVE}}/2} \tag{1}$$

where $t_{\mathrm{CLK|AVE}}$ is the average clock period as V_{DD} varies by a diode voltage drop V_{BE}. Since the clock period is longest at lowest voltage, this is evaluated as V_{DD} ranges from $V_{\mathrm{MIN}} + V_{BE}$ to V_{MIN}, where $V_{\mathrm{MIN}} = V_T + 100$ mV. For our 0.6-μm process, the limit is -20 V/μs. Another failure mode occurs if V_{DD} ramps up by more than V_{Tp} by the end of the evaluation state, and the output drives a pMOS device resulting in a false logic low, giving a functional error. This condition occurs when

$$\frac{dV_{DD}}{dt} \geq \frac{V_{Tp}}{t_{\mathrm{CLK|AVE}}/2} \tag{2}$$

and $t_{\mathrm{CLK|AVE}}$ is evaluated as V_{DD} varies from V_{MIN} to $V_{\mathrm{MIN}} + V_{Tp}$, since this condition is also most severe at low voltage. For our 0.6-μm process, the limit is 24 V/μs.

These limits assume that the circuit is in the evaluation state for no longer than half the clock period. If the clock is gated, leaving the circuit in the evaluation state for consecutive cycles, these limits drop proportionally. Hence, the clock should only be gated when the circuit is in the precharge state. These limits may be increased to that of static CMOS logic using a small bleeder pMOS device to hold the output at V_{DD} while it remains undriven. The bleeder device also removes the constraint on gating the clock, and since the bleeder device can be made quite small, there can be insignificant degradation of circuit delay due to the pMOS bleeder fighting the nMOS pull-down devices. The charge-redistribution problem of dynamic logic will be magnified by a varying supply voltage such that the internal nodes of nMOS stacks should be properly precharged [18].

C. Tri-State Busses

Tri-state busses that are not constantly driven for any given cycle suffer from the same two failure modes as seen in dynamic logic circuits due to their floating capacitance. The resulting dV_{DD}/dt can be much lower if the number of consecutive undriven cycles is unbounded. Tri-state busses can only be used if one of two design methods are followed.

The first method is to ensure by design that the bus will always be driven. While this is done easily on a tri-state bus with only two drivers, this may become expensive to ensure by design for a large number of drivers N, which requires routing N, or $\log(N)$, enable signals.

The second method is to use weak, cross-coupled inverters which continually drive the bus. This is preferable to just a bleeder pMOS as it will also maintain a low voltage on the floating bus. Otherwise, leakage current may drive the bus high while it is floating for an indefinite number of cycles. The size of these inverters can be quite small, even for a large bus. For our 0.6-μm process, the inverters could be designed to tolerate a

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

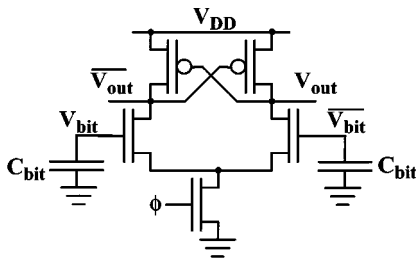

Fig. 12. Basic sense amplifier topology.

dV_{DD}/dt in excess of 75 V/μs with negligible increase in delay, while increasing by only 10% the energy consumed driving the bus.

D. Sense Amps

The basic sense-amplifier topology, shown in Fig. 12, responds to the varying V_{DD} in a desirable manner. When V_{DD} increases, the cell current drive pulling down $V_{\rm bit}$ increases because the cell's internal voltage increases, and the trip point of the sense amplifier shifts up. Likewise, when V_{DD} decreases, the cell current drive decreases, and the trip-point shifts down. The net affect is that the decrease/increase in response time of the sense amplifier with dV_{DD}/dt is relatively similar to the decrease/increase in clock period. Thus, the basic sense amplifier is very suitable for DVS, though second-order delay variation limits dV_{DD}/dt on the order of 5 V/μs, which ultimately determines the maximum slew rate allowed on the supply voltage.

E. Circuit Delay Variation

While circuit delays track well over voltage, subtle delay variations do exist which impact circuit timing. To demonstrate this, three chains of inverters were simulated whose loads were dominated by gate, interconnect, and diffusion capacitance respectively. To model paths dominated by stacked devices, a fourth chain was simulated consisting of four pMOS and four nMOS transistors in series. The relative delay variation of these circuits is shown in Fig. 13 for which the baseline reference is an inverter chain with a balanced load capacitance similar to the ring oscillator.

The relative delay of all four circuits is a maximum at only the lowest or highest operating voltages. This is true even including the effect of the interconnect's *RC* delay. Since the gate dominant curve is convex, combining it with one or more of the other effects' curves may lead to a relative delay maxima somewhere between the two voltage extremes. However, all the other curves are concave and roughly mirror the gate dominant curve such that this maxima will be at most a few percent higher than at either the lowest or highest voltage, and therefore insignificant. Thus, timing analysis is only required at the two voltage extremes, and not at all the intermediate voltage values.

As demonstrated by the series dominant curve, the relative delay of four stacked devices rapidly increases at low voltage, and larger stacks will further increase the relative delay [17]. Thus, to improve the tracking of circuit delay over voltage, a general design guideline is to limit the number of stacked devices, except for circuits whose alternative design would be significantly more expensive in area and/or power (e.g., memory address decoder).

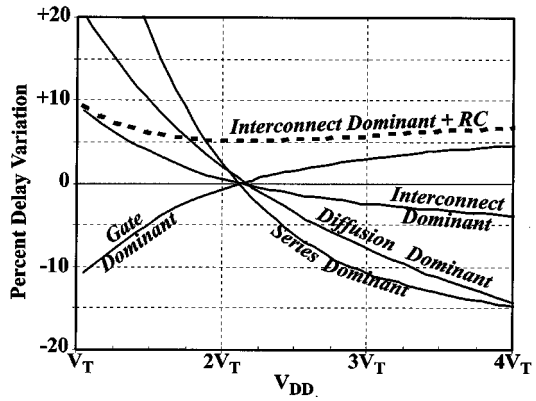

Fig. 13. Relative CMOS circuit delay variation (simulated).

VII. Architectural Enhancements for DVS

A. Desired Frequency Register

The primary architectural support for DVS is the addition of the desired frequency register, which has been added to the system coprocessor. Writes to this register send a new frequency request to the regulator, and reads report the current measured clock frequency. This allows the operating system to actively monitor the operating frequency. To reduce the pin count on the CPU-regulator interface, the 7-bit frequency value is serialized by the CPU and transmitted to the regulator upon writing to the register. The regulator then converts the serial data back to a 7-bit word. The interface requires just three pins to transmit the new frequency value, and one pin to transmit the clock signal from the ring oscillator.

B. Dynamic Performance Monitors

The system coprocessor also contains several read-only registers that monitor system run-time performance. Four registers track processor performance by counting the number of cycles the processor spends in each of its states: active, idle, sleep, and stalled. A separate register counts the number of instructions executed. Another four registers track cache system performance by counting the cache hits, misses, cache-line write-backs, and uncached accesses. These nine registers provide dynamic feedback to the operating system on processor utilization, which can be used to vary the processor speed accordingly.

C. Ring Oscillator

To accommodate process variation over the die as well as simulation error, the oscillator was designed to be programmable from 50% to 150% of nominal frequency with 5 bits of control. The frequency control is designed to be glitch-free so that it can be programmed via software through another register in the system coprocessor.

The basic oscillator architecture, shown in Fig. 14, consists of five binary weighted delay blocks, plus a return path to close the loop. Each of the delay blocks has both a slow and fast path which is selected by the ***ctrl*** signal. A new value for this

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

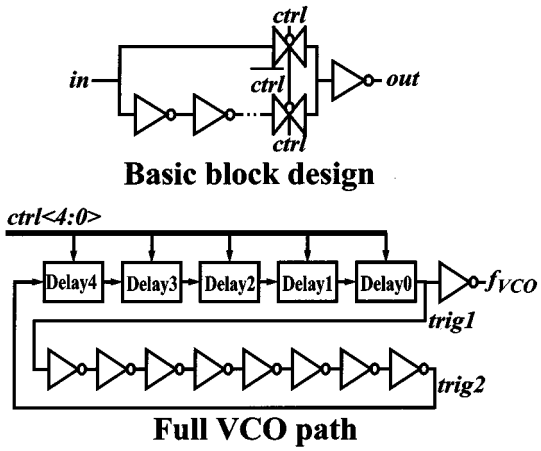

Fig. 14. VCO architecture.

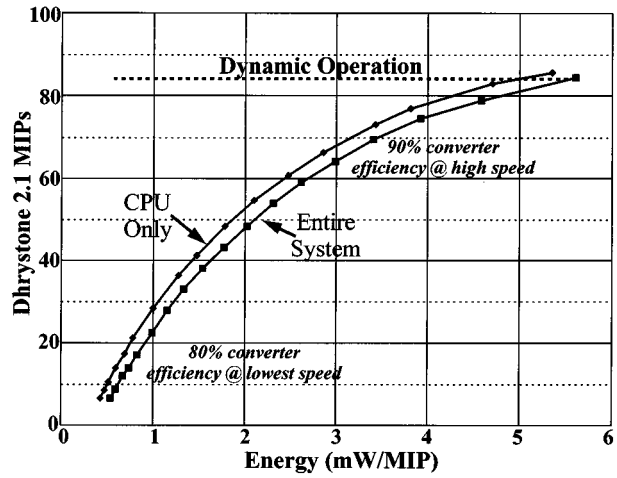

Fig. 15. Measured throughput versus energy consumption.

signal can be loaded in when the ***trig1*** signal transitions low-to-high. By ensuring that the pass gates in the basic block have switched by the time ***trig2*** transitions low-to-high, the oscillator will change frequency glitch-free.

The hardware was stepped from 5 to 80 MHz in 5-MHz increments, and at each step, the ring oscillator's control bits were decreased until processor failure. Decreasing the control bits had the effect of decreasing supply voltage since the converter loop maintains constant clock frequency. The minimum control setting was exactly the setting for nominal frequency at all frequency values, with the exception at 5 MHz, at which speed the control could be decreased by 1 LSB from nominal. This demonstrates that the critical paths of a CMOS processor do track extremely well over a wide range of voltage.

VIII. Measured Results

A. Range of Operation

A plot of throughput versus energy consumption is shown in Fig. 15. The upper curve is for the processor system running off of a fixed voltage source while the lower curve is for the entire system with the regulator powered by a battery voltage. The curves are generated by running the system at constant frequency and supply voltage to demonstrate the full operating range of the system. The throughput ranges from 6–85 Dhrystone 2.1 MIPs, and the total system energy consumption ranges from 0.54–5.6 mW/MIP. The efficiency of the dc-dc converter, which is the ratio of the regulated power (measured at fixed voltage) to the power drawn from the battery, ranges from 90% at high voltage to 80% at low voltage.

With DVS, peak throughput can be delivered on demand. Thus, the true operating point for the system lies somewhere along the dotted line because 85 MIPS can always be delivered when required. In the optimum case when only a small fraction of the computation requires peak throughput, the microprocessor system can deliver 85 MIPs while consuming on average as little as 0.54 mW/MIP.

A common energy-efficiency metric is MIPS/W. The equivalent for this system would be the ratio of peak MIPS to average power dissipation because the throughput and power dissipation can be dynamically varied. In the optimal case when peak throughput is required only a small fraction of the time, the system's average power dissipation can be as low as 3.24 mW, yielding 26 200 MIPS/W. When the system is operated at constant voltage, the energy-efficiency is a maximum of 1850 MIPS/W at 1.2 V.

B. Idle Energy Consumption

Because a microprocessor in portable systems idles a significant amount of time, a sleep mode has been implemented to minimize idle energy consumption. When the halt instruction is executed, which was implemented via a write to a system control register, the processor stops the clock to the processor core. This effectively stops all activity by clock gating the rest of the components in the system. The bus interface continues clocking a small finite-state machine to grant any DMA request that may come in while the processor is in sleep mode.

If the processor speed is set to 5 MHz before entering sleep, the entire system will dissipate only 800 μW of power, with a one cycle start-up from sleep. This is possible because the VCO and regulation loop are continually operating, albeit at their lowest-energy operating points. To achieve low-power sleep modes, other processors require powering down the voltage supply and/or PLL [1]. A high-speed frequency change can be immediately initiated upon detection of an interrupt to minimize interrupt latency via a separate interrupt-frequency register. The latency to ramp back to full speed is set by the regulation loop to be 70 μs, but the processor can continue operating during this ramp up period and begin immediate execution of the interrupt handler.

C. Benchmark Programs

To evaluate DVS, benchmarks were chosen that represented software applications that are typically run on notebook computers or PDA devices. Existing benchmarks (e.g., SPEC, MIPS, etc.) are not useful because they were constructed to only measure the peak throughput of a processor. New benchmarks were selected which combine computational requirements with realistic latency constraints. The three benchmarks that were executed on the system are:

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

TABLE I
MEASURED BENCHMARK ENERGY CONSUMPTION (NORMALIZED)

	Benchmark Programs		
Algorithm	MPEG	UI	AUDIO
Maximum Performance	100%	100%	100%
Optimal	67%	25%	16%
Voltage Scheduler	89%	30%	22%

1) MPEG: MPEG-2 decoding of an 80-frame 192 × 144 video at 5 frames/s, requiring an average 50-MHz clock rate in a single-task environment.
2) AUDIO: IDEA decryption of a 10-s 11-kHz mono audio stream, divided into 1-kB frames with a 93-ms deadline, requiring an average 17-MHz clock rate.
3) UI: A simple address-book user interface allowing simple searching, selection, and database selection. 432 frames are processed, each defined as a user triggered event, such as pen-down, which ends when the corresponding action has been completed. Most frames require less than a 10-MHz clock rate, while some frames are very compute intensive.

The key parameter to measure the energy-efficiency improvement of DVS is the system energy consumption. Energy consumption was measured by charging up a large (3.5 F) capacitor to the battery voltage, and measuring the voltage drop on it over the duration of the benchmark. The benchmarks were first run at constant maximum throughput to measure the baseline energy consumption. They were then run with the voltage scheduler and the energy consumption was measured again.

Table I shows the measured system energy consumption for the three benchmarks, and is normalized to when the system is running at maximum throughput, which is the typical operating mode of a processor system that operates from a fixed supply voltage. The row labeled Optimal is the energy reduction when all the computational requirements are known *a priori*, and is an estimated value derived from simulation. The optimal values represent the maximum achievable energy reduction for these benchmarks. The last row is the measured energy consumption with the voltage scheduler enabled. As expected, the compute-intensive MPEG benchmark has only a 11% energy reduction from DVS. However, DVS demonstrates significant improvement for the less compute-intensive AUDIO and UI benchmarks, which have a 4.5× and 3.5× energy reduction, respectively. Comparing the DVS results against the optimal results demonstrates that while the voltage scheduler's heuristic algorithm has a difficult time optimizing for compute-intensive code, it performs extremely well on non-speed-critical applications.

Table II shows the average power dissipation of the three benchmarks with the voltage scheduler operating. The effective MIPS/W is calculated as the ratio of peak throughput (85 MIPS) to average power dissipation, and demonstrates the achievable increase in energy efficiency when the system is running real programs. Both the UI and AUDIO benchmarks have an average power dissipation on the order of 10mW, yielding an energy efficiency on the order of 10 000 MIPS/W.

TABLE II
MEASURED POWER DISSIPATION WITH THE VOLTAGE SCHEDULER

	Benchmark Programs		
Voltage Scheduler	MPEG	UI	AUDIO
Average Power (mW)	145	11.75	8
Effective MIPS/W	600	7,200	10,600

IX. CONCLUSION

The prototype processor system demonstrates that DVS can improve the energy efficiency of battery-powered processor systems by up to a factor of 10x without sacrificing peak throughput. DVS is amenable to standard digital CMOS processes, with a few additional circuit design constraints. Existing operating systems can be retrofitted to support DVS, with little modification, as the voltage scheduler can be added to the operating system in a modular fashion. Finally, the prototype system demonstrated that when running real programs, typical of those run on notebook computers and PDAs, DVS provides a significant reduction in measured system energy consumption, thus significantly extending battery life.

ACKNOWLEDGMENT

The authors would like to thank P. Laramie, O. Rowhani, C. Chang, R. Davis, and J. C. Rudell for their contributions.

REFERENCES

[1] J. Montanaro, *et al.*, "A 160–MHz, 32–b 0.5W CMOS RISC processor," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1703–1714, Nov. 1996.
[2] E. Vittoz, "Micropower IC," in *Proc. IEEE ESSCC*, Sept. 1980, pp. 174–189.
[3] A. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 27, pp. 473–484, Apr. 1992.
[4] B. Davari, R. Dennard, and G. Shahidi, "CMOS scaling for high performance and low power—The next ten years," *Proc. IEEE*, pp. 595–606, Apr. 1995.
[5] *Advanced Configuration and Power Interface Specification, Revision 1.0b*, Intel/Microsoft/Toshiba, Feb. 1999, pp. 67–69.
[6] J. Rabaey, *Digital Integrated Circuits—A Design Perspective*. Englewood Cliffs, NJ: Prentice Hall, 1996.
[7] V. von Kaenel, P. Macken, and M. Degrauwe, "A voltage reduction technique for battery-operated systems," *IEEE J. Solid-State Circuits*, vol. 25, pp. 1136–1140, Oct. 1990.
[8] T. Kuroda, *et al.*, "Variable supply-voltage scheme for low-power high-speed CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 33, pp. 454–462, Mar. 1998.
[9] L. Nielsen, C. Niessen, J. Sparso, and K. van Berkel, "Low-power operation using self-timed circuits and adaptive scaling of the supply voltage," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 391–397, Dec. 1994.
[10] A. Chandrakasan, V. Gutnik, and T. Xanthopoulos, "Data driven signal processing: An approach for energy efficient computing," in *IEEE ISLPED Dig. Tech. Papers*, Aug. 1996, pp. 347–352.
[11] G. Wei, J. Kim, D. Liu, S. Sidiropoulos, and M. Horowitz, "A variable-frequency parallel I/O interface with adaptive power supply regulation," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2000, pp. 298–299.
[12] "ARM 8 Data-Sheet," ARM Ltd., Doc. No. ARM-DDI-0080C, July 1996.
[13] T. Pering, T. Burd, and R. W. Brodersen, "Voltage scheduling in the 1pARM microprocessor system," in *IEEE ISLPED Dig. Tech. Papers*, July 2000, pp. 96–101.
[14] T. Pering, "Energy-efficient operating system techniques," Ph.D. dissertation, Univ. California, Berkeley, CA, 2000.
[15] A. Stratakos, "High-efficiency, low-voltage dc-dc conversion for portable applications," Ph.D. dissertation, Univ. California, Berkeley, CA, 1999.

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.

[16] A. Stratakos, R. W. Brodersen, and S. Sanders, "High-efficiency low-voltage dc-dc conversion for portable applications," in *Int. Workshop Low-Power Design*, Apr. 1994, pp. 619–626.
[17] T. Burd and R. W. Brodersen, "Design issues for dynamic voltage scaling," in *IEEE ISLPED Dig. Tech. Papers*, July 2000, pp. 9–14.
[18] N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*. Reading, MA: Addison Wesley, 1993.

Thomas D. Burd (S'94) received the B.S and M.S. degrees in electrical engineering from the University of California, Berkeley, in 1992 and 1994, respectively, where he is currently working toward the Ph.D. degree in the area of energy-efficient processor system design.

For the InfoPad research project at Berkeley, he developed a low-power CMOS cell library for custom DSP ASICs, which was used in the design of several custom chips. He has since worked on energy-efficient system, architecture, and circuit design for general-purpose processors, energy-efficient low-swing bus transceivers, CAD methodology to automate IC verification, and DVS converter loop architecture design.

Mr. Burd is the recipient of the 1998 Analog Devices Outstanding Student Award for recognition in IC design. He is a member of Tau Beta Pi and Eta Kappa Nu.

Trevor Pering received the B.S. degree in computer science and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 1993 and 2000, respectively. In September, 1999, he joined the Microprocessor Research Lab, Intel Corporation, near Portland, OR.

At Berkeley, he worked on the InfoPad project, where he was responsible for the design and implementation of hardware-based wireless transmission protocols, as well as system-level integration and debugging of the InfoPad portable terminal. His Ph.D. work focused on energy-efficient software techniques for portable computers, including the design of a real-time voltage-scaling operating system. With Intel, he is currently engaged in user interfaces and system-level design issues for ubiquitous computing environments.

Anthony J. Stratakos received the B.S. and M.S. degrees in electrical engineering from The Johns Hopkins University, Baltimore, MD, in 1992, and the Ph. D. degree in electrical engineering from the University of California, Berkeley, in 1998.

While at Berkeley, he worked on the InfoPad research project in the area of low-power mixed-signal design, with particular emphasis on low-voltage dc-dc conversion. In 1996, he co-founded Volterra, where he presently serves as Vice President of Advanced Research and Development and Chief Technical Officer. At Volterra, he has worked on low-voltage dc-dc converters for low-power portable/wireless and high-performance microprocessor applications.

Robert W. Brodersen (M'76–SM'81–F'82) received the Bachelor of Science degrees in Electrical Engineering and Mathematics from California State Polytechnic University, Pomona, CA in 1966; the Engineering and Master of Science degrees from the Massachusetts Institute of Technology (MIT), Cambridge in 1968; and the Ph.D. degree in Engineering from MIT in 1972. On May 28, 1999, Professor Brodersen was formally declared Technologiae Doctor Honoris Causa (Honorary Doctor of Technology) by the University of Lund, Sweden.

He is a Professor in the Department of Electrical Engineering and Computer Sciences (EECS) at the University of California, Berkeley. He joined the EECS Department faculty in 1976. From 1972–1976, he was a member of the Technical Staff, Central Research Laboratory at Texas Instruments. In addition to teaching, his present research focus is the application of integrated circuits as applied to personal communication systems with emphasis on wireless communications and low power design. He was appointed the first holder of the John R. Whinnery Chair in the Department of Electrical Engineering and Compputer Science, University of California, Berkeley, in September, 1995. He was the National Chair of Information Science and Technology (ISAT) Study Group sponsored by the Institute for Defense Analysis, Washington, D.C., from 1992 to 1994. He currently serves on several committees associated with the National Academy of Sciences, Washington, D.C. He is the author or co-author of over 60 journal publications and 120 published conference papers, and author, co-author, editor, or contributor to 14 books, including *Anatomy of a Silicon Compiler* (Norwood, MA: Kluwer, 1992), and *Low Power Digital CMOS Design* (Norwood, MA: Kluwer, 1995). He is the holder of three patents. He has served on the editorial board or as reviewer for numerous scholarly journals and publications, including the IEEE JOURNAL OF SOLID-STATE CIRCUITS, IEEE TRANSACTIONS ON VLSI SYSTEMS, IEEE PERSONAL COMMUNICATIONS MAGAZINE, and *Wireless Personal Communications* (Kluwer Press).

He won conference best paper awards at Eascon in 1973, the International Solid-States Circuits Conference in 1975, and the European Solid-States Circuits Conference in 1978. In 1979, he received the W.G. Baker Award for the outstanding paper in the IEEE Journals and Transactions. He was co-recipient of the Morris Libermann Award of the IEEE in 1983 for "outstanding contributions to an emerging technology." He received the best paper award in the Transactions on CAD in 1985 and the best tutorial paper of the IEEE Communications Society in 1992. In 1997, he received the distinguished IEEE Solid-States Circuits Award "for contributions to the design of integrated circuits for signal processing systems." He is a member of the National Academy of Engineering.

Authorized licensed use limited to: Russel Baker. Downloaded on August 01,2025 at 23:58:12 UTC from IEEE Xplore. Restrictions apply.