# Energy-Efficient Subthreshold Processor Design

Bo Zhai, Sanjay Pant, Leyla Nazhandali, Scott Hanson, Javin Olson, Anna Reeves, Michael Minuth, Ryan Helfand, Todd Austin, *Member, IEEE*, Dennis Sylvester, *Senior Member, IEEE*, and David Blaauw

***Abstract*—Subthreshold circuits have drawn a strong interest in recent ultralow power research. In this paper, we present a highly efficient subthreshold microprocessor targeting sensor application. It is optimized across different design stages including ISA definition, microarchitecture evaluation and circuit and implementation optimization. Our investigation concludes that microarchitectural decisions in the subthreshold regime differ significantly from that in conventional superthreshold mode. We propose a new general-purpose sensor processor architecture, which we call the Subliminal Processor. On the circuit side, subthreshold operation is known to exhibit an optimal energy point ($V_{\min}$). However, propagation delay also becomes more sensitive to process variation and can reduce the energy scaling gain. We conduct thorough analysis on how supply voltage and operating frequency impact energy efficiency in a statistical context. With careful library cell selection and robust static RAM design, the Subliminal Processor operates correctly down to 200 mV in a 0.13-$\mu$m technology, which is sufficiently low to operate at $V_{\min}$. Silicon measurements of the Subliminal Processor show a maximum energy efficiency of 2.6 pJ/instruction at 360 mV supply voltage and 833 kHz operating frequency. Finally, we examine the variation in frequency and $V_{\min}$ across die to verify our analysis of adaptive tuning of the clock frequency and $V_{\min}$ for optimal energy efficiency.**

***Index Terms*—Sensor networks, subthreshold design, $V_{\min}$, ultra low power design.**

## I. Introduction

RAPID advances in digital circuit design has enabled a number of applications requiring complex sensor networks. This application space ranges widely from environmental sensing [1] [2] to structural monitoring [3] to supply chain management [4]. Highly integrated sensor network platforms [5] would combine MEMS sensing capabilities with digital processing and storage hardware, a low power radio, and an on-chip battery in a volume on the order of 1 mm$^3$. The design of energy-efficient data processing and storage elements is therefore paramount.

Voltage scaling into the subthreshold regime ($V_{\rm dd} < V_{\rm th}$) has recently been shown to be an extremely effective technique for achieving minimum energy. In previous work [10], we demonstrated the existence of a minimum energy voltage ($V_{\min}$), where CMOS logic reaches maximum energy efficiency per operation. This occurs when leakage energy and dynamic energy are comparable [11]. Fig. 1 shows the simulated energy consumption of a chain of 50 inverters as a function of supply voltage in 0.13-$\mu$m technology. A single transition is used as a stimulus and energy is measured over the time period necessary to propagate the transition through the chain. The dynamic energy component $E_{\rm act}$ reduces quadratically while the leakage energy $E_{\rm leak}$ increases with voltage scaling. This effect creates a minimum energy point (referred to as $V_{\min}$) that lies at 200 mV for the simulated inverter chain. Scaling the supply voltage below $V_{\min}$ ceases to reduce energy per operation due to the exponential increase of circuit delay with $V_{\rm dd}$, which causes leakage to dominate total energy consumption. Operating in the subthreshold regime clearly has its benefits, but there has been very little work to investigate the design of general-purpose processors in this region. In this study, we study the architecture- and circuit-level implications of subthreshold design.

We begin by exploring architecture-level energy optimization for low- to mid-performance sensor network processing applications. We examine 21 different microarchitectures with varied datapath widths, degrees of pipelining, prefetching capability, and with different register and memory architectures. Interestingly, we find that many of the area- and performance-optimal designs at subthreshold voltages are not ideal at superthreshold voltages. To further explore energy efficiency and performance at subthreshold voltages, we implemented the most energy-efficient sensor platform (which we call the Subliminal Processor) [12] in a 0.13-$\mu$m technology. In the subthreshold region, variability becomes a serious concern, so we dedicate much of this study to discussing the implications of variability and discuss how circuit design must accommodate this increased variation. Measurements of the Subliminal Processor demonstrate that our implementation attains a maximum energy efficiency of 2.6 pJ/instruction at 360 mV, with an operating frequency of 833 kHz. We use both simulated and measured data to examine the implications of process variation. We find that *dynamic $V_{\rm dd}$ scaling* to fight variability is less important than *dynamic frequency scaling* in subthreshold circuits. We use both simulations and silicon measurements to show that dynamic frequency scaling at a fixed supply voltage set to the nominal value of $V_{\min}$ should be used to minimize energy variability. Several subthreshold circuits [13]–[16], [32] have been presented recently. However, this paper presents a general-purpose sensor processor specifically optimized for energy-efficient subthreshold operation. With our optimization, the minimum energy voltage is achieved at 360 mV (compared to 500 mV in [32]).

The remainder of this paper is organized as follows. Section II introduces our sensor networking applications, representative data streams, and then makes a case for why sensor network

Manuscript received December 23, 2007; revised June 06, 2008. First published April 28, 2009; current version published July 22, 2009.

The authors are with the University of Michigan, Ann Arbor, MI 48128 USA (e-mail: bzhai@umich.edu; spant@umich.edu; lnazhand@umich.edu; hansons@umich.edu; javino@umich.edu; austin@umich.edu; dennis@umich.edu; blaauw@eecs.umich.edu).

Digital Object Identifier 10.1109/TVLSI.2008.2007564

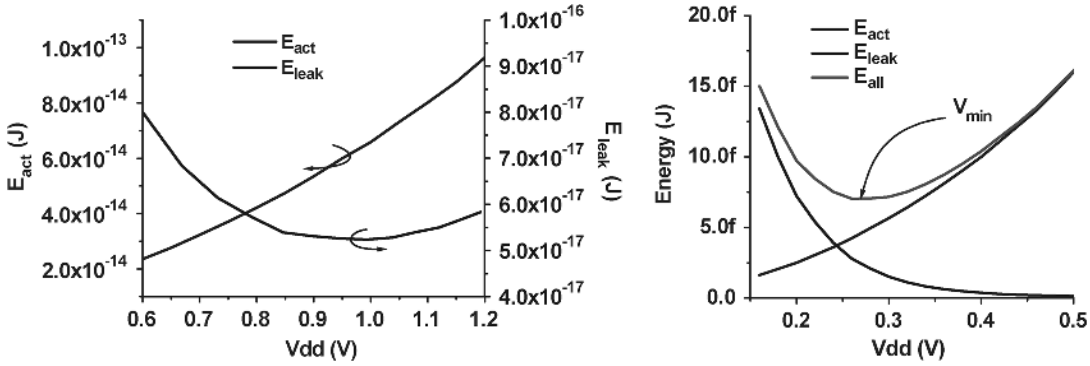



Fig. 1. Energy as a function of supply voltage (HSPICE simulation).

processors should employ subthreshold-voltage circuit implementations. Section III highlights the architecture-level optimizations at ultralow voltages. Section IV discusses the implications of variability and describes the circuit-level implementation, which is aimed at energy-efficient subthreshold operation. Measurement results of the implemented prototype are presented in Section V. Finally, Section VI draws conclusions and gives insights for future sensor network processor designs. Preliminary findings related to this study were first presented in [6], [12], including some of the figures.

## II. Sensor Network Processing

To effectively gauge the processing and energy demands of sensor network processors, we must first assemble a sensor network processing benchmark collection and examine each processor's performance under a variety of sensor processing data streams. Table I [9], [17], [18] lists the sensor network processing benchmarks we examine in this study. The applications are divided into three categories: communication algorithms, computational processing, and sensing algorithms. These programs represent a broad slice of the types of applications one could expect to see on an ultralow energy sensor network processor platform. Note that the last column numbers are static code size in terms of nibbles. Most of these applications contain loops and their dynamic instruction count is much higher.

Sensor network platforms evaluate environmental information in real time, by reading, processing, compressing, storing, and eventually transmitting the information to interested parties. To better understand the computational demands of a real-time sensor network platform, we collected the data processing rates of a variety of phenomena, which encompass a wide range of associated sample rates (in Hertz, samples per second) [23]. We categorize these applications into low-, mid-, and high-bandwidth rates, which reflect sample rates of less than 100 Hz, 100 Hz–1 kHz, and greater than 1 kHz, respectively. Fig. 2 illustrates the performance of four commercial embedded processors, in addition to one energy-efficient sensor network processor design proposed in this paper at three different voltages. Each of the processors are implemented in a 0.13-$\mu$m process. For each processor, we show the xRT rating, which is computed via simulation by determining how many times faster than real time the processor can handle the worst-case data stream rate on

TABLE I
Sensor Network Processing Algorithms

| Application | Description | Code Size nibble |
|---|---|---|
| *Communication Algorithms* | | |
| adRout | Ad-hoc router control algorithm | 42 |
| compRLE | Run-length encoded compressor | 73 |
| TEA | TEA encryption algorithm | 85 |
| crc8 | Cyclic redundancy code generator | 99 |
| *Computational Processing* | | |
| divide | Unsigned integer division | 80 |
| multiply | Unsigned multiplication | 48 |
| inSort | In-place insertion sort | 78 |
| binSearch | Binary search | 90 |
| *Sensing Algorithms* | | |
| intAVG | Signed integer average | 113 |
| intFilt | 4-tap signed FIR filter | 106 |
| tHold | Digital threshold detector | 45 |

the most computationally intensive sensor benchmark. For example, the ARM720T at 1.2 V with a 100-MHz clock is able to process worst-case mid-bandwidth data 2965 times faster than real-time data rates. A few of the high-bandwidth sensor applications can be served by the commercial ARM processors, while the highest bandwidth A/D sample rate greatly exceeds the computation capability of even the most competent embedded processors. Consequently, we restrict our studies in this paper to the lesser demands of the low- and mid-bandwidth sensor network applications. It is clear from Fig. 2 that the low- and mid-bandwidth sensor processing applications have computational demands that are well below those delivered by the commercial ARM processors. The same is true for the energy-efficient proposed design at full voltage (1.2 V) and 114 MHz. This design services the mid-bandwidth applications at more than 2253 times the required worst-case processing requirement.

We can reduce the energy demands of these applications by reducing the frequency of the processor, which in turn accommodates reductions in the voltage. As voltage is lowered, energy demands will decrease quadratically. However, even the lowest superthreshold voltages still deliver too much performance. The energy-efficient proposed design is shown in Fig. 2 at 0.5 V and runs with a 9-MHz clock. Even this low-voltage design is capable of delivering 180 times the performance required

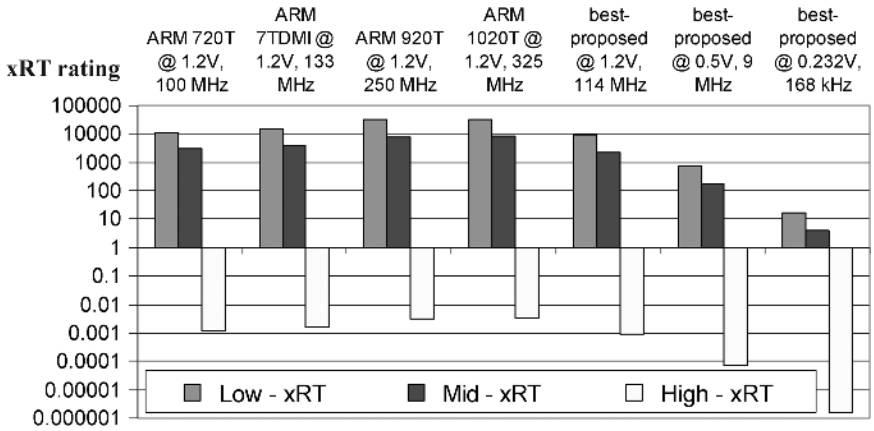



Fig. 2. Performance (relative to worst-case data stream rate) of sensor network processor applications on embedded targets.

by the low- and mid-range sensor processing applications. To further reduce energy requirements, we must consider running our sensor network processors at subthreshold voltages. The energy-efficient subthreshold design in Fig. 2 delivers more than four times the desired performance for mid-bandwidth applications at 232 mV with a 168-kHz clock.

It is noteworthy to mention that even increasing the sleep time of the processors is not helpful in reducing the energy *per instruction*. The run-and-sleep technique, in which the processor runs to execute a job and goes to sleep when the job is finished, reduces the overall energy consumption of a processor because it saves the energy consumed in idle state. However, in our analysis we are considering energy per instruction; hence, not including the idle energy consumption. In other words, we are making a comparison between the energy consumption of the processors during their service time, and assume they all employ some technique to save energy in idle periods.

The next section performs a detailed tradeoff study to determine which ISA and microarchitectural features are the best for reducing energy at subthreshold voltages.

## III. Architecture-Level Energy Optimization

Subthreshold circuit design differs from superthreshold design in that even circuits with low switching activity have a high impact on energy efficiency due to their leakage current. At subthreshold, the optimal operating voltage is determined by the balance between active and leakage energy. Higher activity rate reduces the wasteful leakage percentage per useful switching and therefore allows us to further scale down the operating voltage. However, it is essential that each switching activity contributes to useful computation and not just spurious switching, which would unnecessarily increase dynamic energy.

Hence, processors with simple control complexity are advantageous since they typically result in compact circuits with high activity rate and a low leakage/dynamic current ratio which in turn yields a low $V_{\min}$ and low overall energy consumption. At the same time, however, the required code size must be minimized to reduce leakage in the memory array. We examined this tradeoff between instruction set expressiveness (which leads to compact code size) and control logic complexity and found that in general, a decrease in code size outweighs the increase in control logic complexity in terms of energy efficiency in subthreshold operation. Therefore, we choose a CISC ISA as the focus of our study for higher code density and smaller memory requirement.

TABLE II
Sensor Network Processor ISA Summary

| Mnemonic | Operation | Length nibbles |
|---|---|---|
| ADD | Performs addition | 2 or 3 |
| SUB | Performs subtraction | 2 or 3 |
| AND | Performs logical and | 2 or 3 |
| OR | Performs logical or | 2 or 3 |
| XOR | Performs logical exclusive or | 2 or 3 |
| SHFT | Shifts the accumulator | 2 or 3 |
| LOAD | Loads the accumulator | 2 or 3 |
| STOR | Stores the accumulator | 2 or 3 |
| DW_BK | Sets BLCK and DW specifiers | 2 |
| PTR_INC | Increments pointer register | 2 |
| PTR_DEC | Decrements pointer register | 2 |
| PTR_LOAD | Loads acc. with pointer reg. | 2 |
| PTR_STOR | Stores acc. into pointer reg. | 2 |
| CALL | Calls a function | 3 |
| RET | Returns from a function | 1 |
| JUMP | Conditionally jumps to target | 4 |
| NOP | No operation | 1 |

Table II summarizes our sensor network processor instruction set. The table lists the instruction mnemonic, a short description of the instruction, and its size in nibbles. Our instruction set is a simple 32/16/8-bit single-operand ISA. The instruction set contains two register banks: a 4-entry 32-bit integer register file and a 4-entry 16-bit pointer register file. The pointer registers hold memory addresses, so the architecture can address up to 64 kB of storage. All computational instructions are of the form

$$(\mathrm{Acc}) \leftarrow (\mathrm{Acc}) \otimes \mathrm{operand}$$

where operand is either: 1) a general-purpose register operand; 2) a pointer register which specifies a value in memory; 3) a direct 6-bit memory address; or 4) a 2-bit signed immediate value.

Fig. 3 illustrates the tradeoff between ISA expressiveness (which results in a smaller code size) and increased control logic complexity. The PTR instructions provide efficient memory addressing by providing a compact means, in the

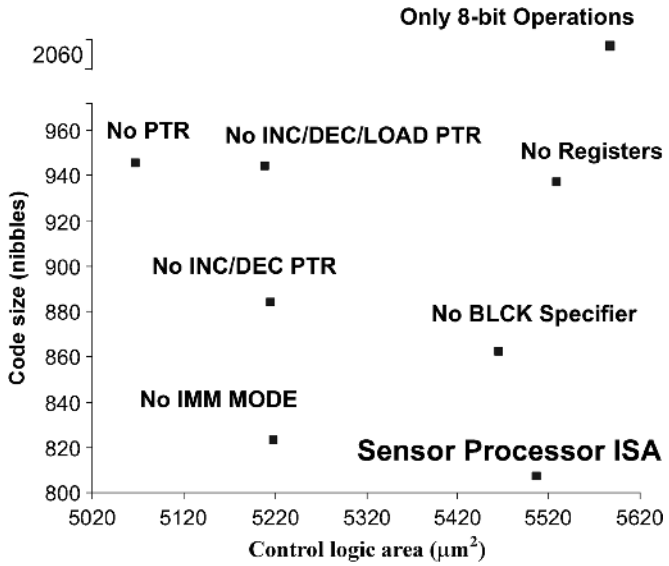



Fig. 3. Impact of ISA optimization on code size and logic complexity.

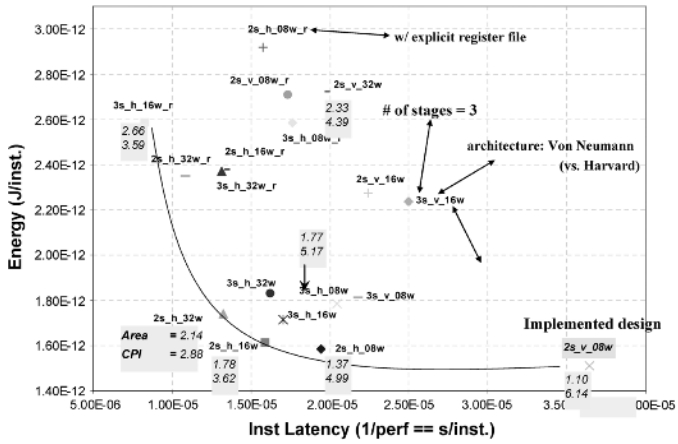



Fig. 4. Pareto analysis for 18 processors.

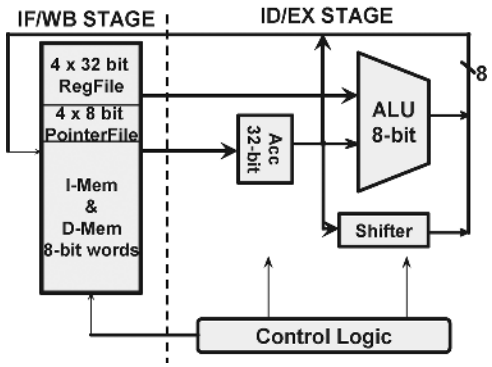



Fig. 5. Proposed architecture.

form of pointer registers, to express addresses and efficiently implement strided accesses. Eliminating the pointer registers, while reducing control complexity, has a significant impact on code size, increasing overall code size by 16%. Eliminating the general-purpose registers has a similar effect on code size, with little benefit to control complexity. The DW BK instruction sets both BLCK and DW specifiers. The BLCK specifier is used to take advantage of locality in absence of caches, where one can choose the working block in memory and therefore reduce the number of address bits in order to shorten the instruction. Eliminating the block specifier increases code size about 6% with a slight increase in control complexity. Finally, eliminating the ability to process 16- and 32-bit data types (implemented via the DW specifier, which determines the virtual width of the datapath) bloats code size by nearly 2.5×. This increase is due to the many additional instructions required to implement 16- and 32-bit operations (e.g., a 16-bit operation requires an 8-bit add, plus an 8-bit add-with-carry.) Removing support for multiple data widths provides little benefit to control complexity.

We investigated 18 different implementations of the CISC ISA, considering different combinations of the number of stages, the ALU width, explicit or implicit register files, and Von Neumann versus Harvard memory architectures. The implementations are shown in the Pareto plot in Fig. 4. The designs are labeled to indicate: 1) the number of pipeline stages (1 s, 2 s, or 3 s); 2) the number of memories (v—one memory, h-I, and D memory); 3) datapath width (8 w, 16 w, or 32 w); and 4) with (_r) or without explicit registers (designs without explicit registers store register values in the memory). Designs on the curve are pareto-optimal and the designs closer to the origin are faster and more energy efficient than designs farther away. The energy numbers in Fig. 4 were based on the netlists sythesized using a low-voltage library characterized at 250 mV. We selected the minimum energy implementation, which is labeled "2 s_v_08 w" in Fig. 4. The microarchitecture of the selected implementation, as shown in Fig. 5, consists of two pipeline stages, a unified memory for register file, pointer file, instruction memory and data memory, an 8-bit wide ALU and a 32-bit accumulator which is the only place where instruction results are stored. The implementation and test of this processor, which we call Subliminal, will be described in the next two sections.

## IV. Circuit Implementation for Optimal Energy Efficiency

In this section, we discuss the circuit implementation of the Subliminal microarchitecture described in the previous section. We begin with a focus on variability. We find, in particular, that dynamic frequency adaptivity is more important than dynamic voltage adaptivity when minimizing energy in subthreshold circuits subject to variability. We follow this discussion with a detailed description of our implementation of the Subliminal Processor.

### A. Addressing Variability

Process parameter variation has become a critical concern in nanometer technologies. The impact of process variation is further exacerbated at lower operating voltages [19]–[21]. In general, process variability can be broken into two categories: random variations and systematic variations. We focus briefly on both types of variation and discuss their implications on the design of the Subliminal Processor.

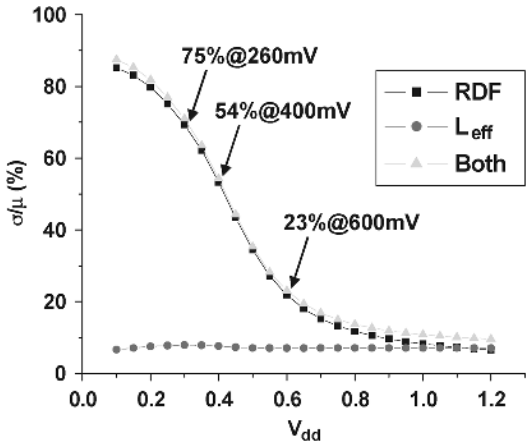



Fig. 6. 0.3-$\mu$m-wide NMOS ON-current variation of different sources with supply voltage in terms of $\sigma/\mu$ (from HSPICE simulation).

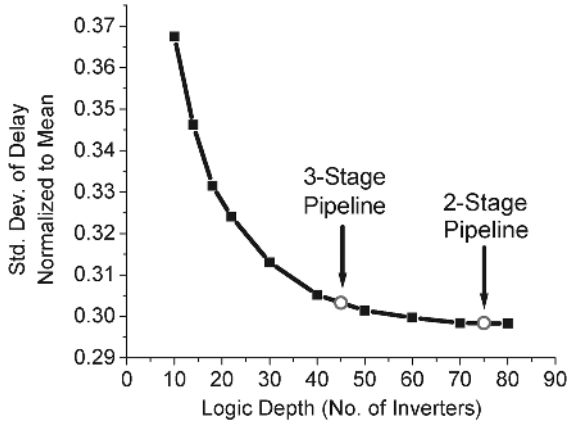



Fig. 7. Simulated delay variation with logic depth.

Simulated ON-current variation due to random process variations is shown in Fig. 6 for a 0.3-$\mu$m-wide n-type MOS (NMOS) in a 0.13-$\mu$m technology. We model $V_{\rm th}$ and $L_{\rm eff}$ variations since these are most important in a subthreshold device. Note that ON-current variation increases from $\sigma/\mu = 23\%$ at $V_{\rm dd} = 600$ mV to 75% at $V_{\rm dd} = 260$ mV for the simulated device. Also note that $V_{\rm th}$ variation, which is largely caused by random dopant fluctuations (RDFs), becomes the dominant source of variability at low voltage [22]. Due to its uncorrelated nature, RDF averages out over the length of a path making shallow pipelines with a large number of gates per stage advantageous, as shown in Fig. 7 for inverter chains of different lengths. Hence, a two-Stage pipeline implementation is attractive for the processor, which shows a 19% reduction in $\sigma/\mu$ of delay variation compared to a design with 10 gate delays per pipeline stage. The "2-Stage" and "3-Stage" corresponds to the "2 s" and "3 s" designs in the Fig. 4.

The long datapath is the result of both compact code size (more complex control) and subthreshold operation. In order to fully utilize the subthreshold energy, the memory array needs to be designed differently, as shown in Fig. 10. Traditional 6-T static RAM (SRAM) would not work near the threshold voltage. We could pipeline the design more heavily, but the memory speed would not be sufficient, leaving the faster core waiting for memory data.

While averaging helps minimize the effects of random variation, systematic variation in both $V_{\rm th}$ and $L_{\rm eff}$ remains a significant challenge. Due to the exponential dependence of ON-current on $V_{\rm th}$, even small fluctuations in $V_{\rm th}$ necessitate enormous design margins to meet delay and energy yields. Dynamic frequency and $V_{\rm dd}$ adaptivity have been proposed as solutions (usually as a single solution) to systematic process and runtime variations [15], but both techniques require significant hardware overhead. The determination of $V_{\rm min}$, for example, has been shown to require special energy measurement circuits and additional design complexity [29], [30].

To evaluate the effectiveness of dynamic $V_{\rm dd}$ and frequency adaptivity in subthreshold circuits, we consider a nominal system operating at the energy optimal $V_{\rm dd}$, $V_{\rm min}$, with the clock period, $t_p$, set to the minimum possible value, $t_{\rm min}$. Due to process variations, each particular die will have values for $V_{\rm min}$ and $t_{\rm min}$ that are different from this nominal case. We are interested in determining whether it is useful to select unique values for $V_{\rm dd}$ and $t_p$ for each die using dynamic correction or to simply use a single set of values with sufficient margin to guarantee correct operation with reasonable energy consumption across all dies.

Before quantifying the sensitivity of energy consumption to fluctuations in $V_{\rm dd}$ and $t_p$, we run Monte Carlo simulations (1000 trials) on a chain of 30 inverters with switching activity $\alpha = 0.2$. Fig. 8(a) shows that $V_{\rm min}$ for the inverter chain is tightly distributed, with $\sigma/\mu = 3.8\%$. Fig. 8(b) shows the delay distribution for the same inverter chain with the supply voltage fixed at 265 mV, which is the mean of the $V_{\rm min}$ distribution. The delay distribution is much wider, with $\sigma/\mu = 28\%$. The wide distribution of $t_{\rm min}$ is not surprising given the exponential dependence of delay on $V_{\rm th}$. The delay distribution of subthreshold circuit has a longer tail, which can be modeled with a lognormal *probability density function* (pdf) [22].

Even though the raw sensitivity of energy to $V_{\rm dd}$ is, in general, greater than the sensitivity to $t_p$, the data in Fig. 8 suggest that $t_{\rm min}$ variations are actually a much greater concern than $V_{\rm min}$ variations in subthreshold circuits. For example, the energy consumption in the inverter chain increases by only 13% when $V_{\rm dd}$ is increased from 265 to 290 mV [the 99% confidence point in Fig. 8(a)]. Increasing the delay from 393 to 718 ns [the 99% confidence point observed in Fig. 8(b)] results in a much larger energy increase of 29%. These sensitivities suggest that it is more important to control delay than supply voltage when minimizing energy in subthreshold circuits.

We investigate this observation further by performing Monte Carlo simulations for four cases: 1) $V_{\rm dd} = V_{\rm min}$ and $t_p = t_{\rm min}$ for each die; 2) $t_p = t_{\rm min}$ for each die, but for all die $V_{\rm dd}$ is fixed to 265 mV, the mean of the $V_{\rm min}$ distribution; 3) $V_{\rm dd}$ is again fixed to the mean of the $V_{\rm min}$ distribution for all die but $t_{\rm min}$ is also fixed to the maximum value, which we choose to be the 99% confidence point of the delay distribution across all dies; and 4) $V_{\rm dd} = V_{\rm min}$ and $t_{\rm min}$ is again set to the 99% confidence point. The distribution for case 2 is nearly identical to the distribution for case 1. However, the mean energies observed

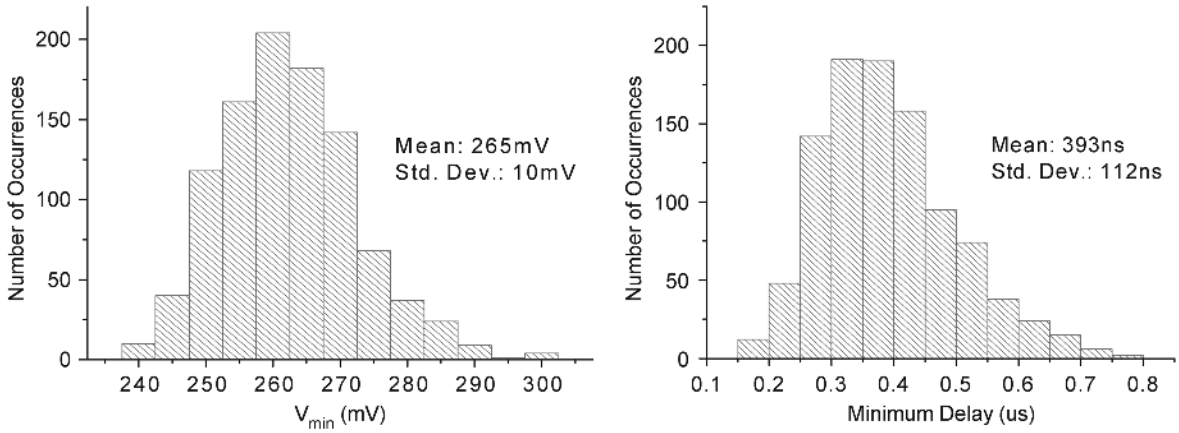
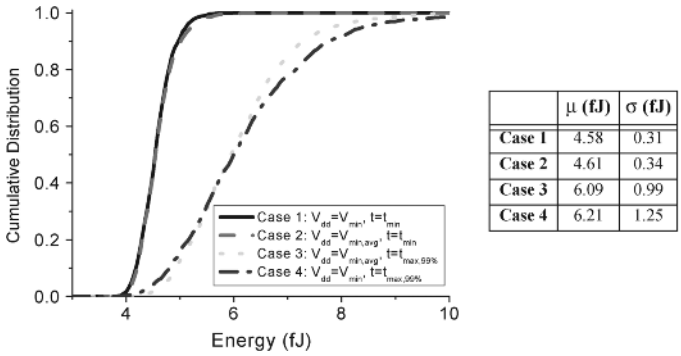



Fig. 8. (a) Simulated distribution of $V_{\min}$ for a chain of 30 inverters subject to gate length and $V_{\rm th}$ variations. (b) Distribution of minimum delay, $t_{\min}$, for the same inverter chain with $V_{\rm dd}$ fixed at 265 mV.




| | μ (fJ) | σ (fJ) |
|---|---|---|
| Case 1 | 4.58 | 0.31 |
| Case 2 | 4.61 | 0.34 |
| Case 3 | 6.09 | 0.99 |
| Case 4 | 6.21 | 1.25 |

Fig. 9. Simulated cumulative distribution function of energy for a chain of 30 inverters subject to variability (HSPICE simulation).

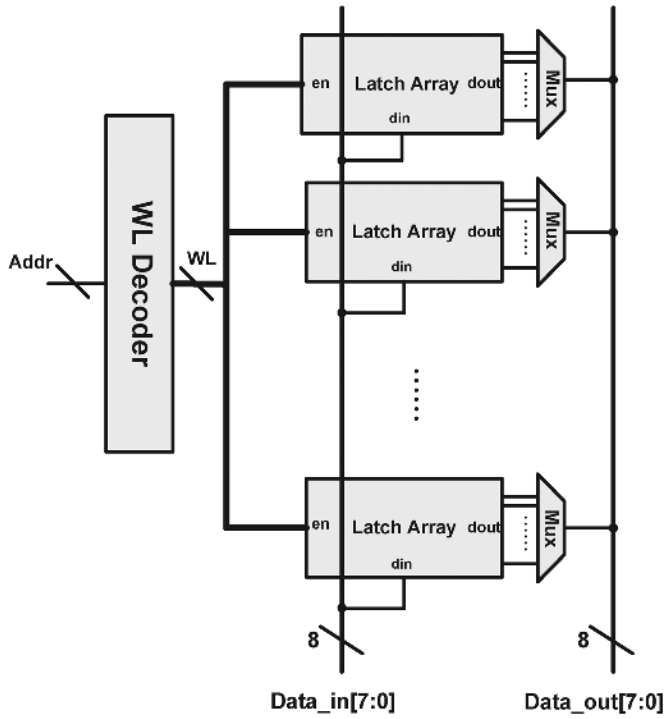



Fig. 10. Memory design for subthreshold operations.

in cases 3 and 4 are more than 30% larger than the mean energy for case 2. It is clear from Fig. 9 that the individual tuning of delay (frequency) is much more effective for minimizing energy than individual tuning of $V_{\rm dd}$. While this observation is not surprising when we consider the very large range in delay observed in Fig. 8(b), it has important implications on system design. Rather than focusing on finding the optimal value for $V_{\rm dd}$ for all dies, subthreshold circuit designers should focus on adaptive frequency scaling. In the subsequent sections, we use hardware measurements to confirm the conclusions made in this section.

### B. Implementation Details

In addition to variability, subthreshold design is complicated by several other factors that merit careful attention. We touch on these issues and describe the relevant design implementation details in this section.

General logic for the 8-bit Subliminal Processor was synthesized using a traditional standard cell-based design flow. For maximum robustness, all gates with more than two fan-ins as well as all pass-transistor logic gates were eliminated from the library, and the library was recharacterized at subthreshold voltages using a custom characterization tool. Simulation shows that a processor synthesized with this dedicated subthreshold library is ∼9% faster at subthreshold voltage than one with a typical commercial standard cell library, although both have the same performance at full $V_{\rm dd}$. This is caused by the different scaling of cell delays with $V_{\rm dd}$. More specifically, a 20% change in the $\beta$ ratio between 1.2 V and 250 mV caused an 18% change in the NAND/NOR cell delay ratio.

The 2 kb memory was implemented using a custom mux-based array structure [14], as shown in Fig. 10. Register file, instruction/data memory are physically one unified SRAM, where the implicit register file is mapped using special address. While this memory structure is area inefficient, it is extremely robust. Measurements show that the memory is functional with $V_{\rm dd}$ as low as 200 mV, which is much lower than $V_{\rm min}$ for the entire processor. Hence, reducing the minimum functional voltage further is unnecessary.

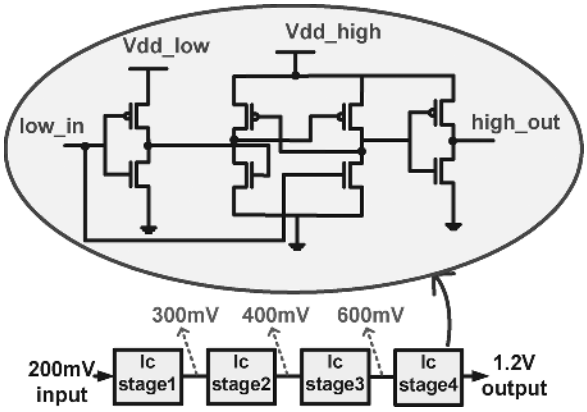



Fig. 11. Level converter design.

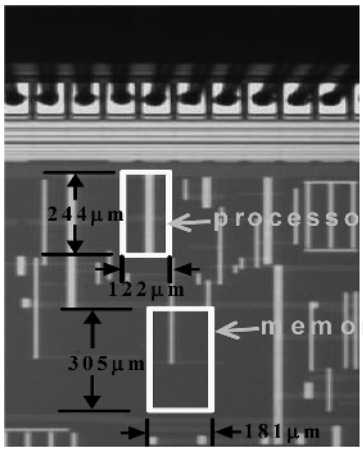



Fig. 12. Die photograph of core–memory combination.

The test harness supports a scan interface to all processor states including the memory and the registers. The scan interface at low voltage is controlled by a robust high-voltage conventional memory with level shifters in between. A dedicated testing environment has been written to load the instruction memory and register as well as to read out the data memory. A special level converter was implemented to convert the 200 mV signals to 1.2 V using four differential subconverter stages as shown in Fig. 11. The subconverter stages convert to 300, 400, 600 mV, and 1.2 V, respectively. In order to suppress process variability and improve robustness, the first two subconverter stages were increased in size and had body bias control to compensate for global $\beta$-ratio shift, if needed.

Fig. 12 shows the die photograph of the core and the memory in the test chip. The test chip was fabricated in an industrial 0.13-$\mu$m CMOS process with eight layers of metal. The area of the processor core is 29817 $\mu$m$^2$ and the area of the memory is 55205 $\mu$m$^2$. The next section presents measurements of the test chip.

## V. Measurement Results and Discussion

In this section, we present the silicon measurements including operating frequency, optimal energy voltage and energy consumption. The statistical energy measurements confirm our analysis and observation in Section IV-A. Finally, we illustrate how different applications affect the processor energy efficiency as well as temperature impact on speed.

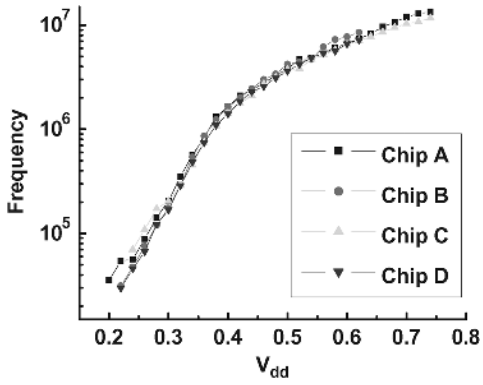



Fig. 13. Measured frequency with $V_{\rm dd}$ for four processors.

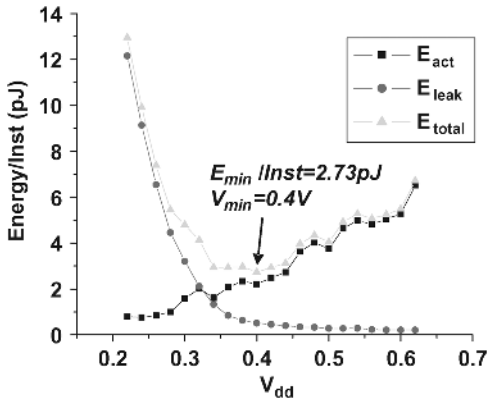



Fig. 14. Dynamic, static, and total energy for the processor as a function of $V_{\rm dd}$.

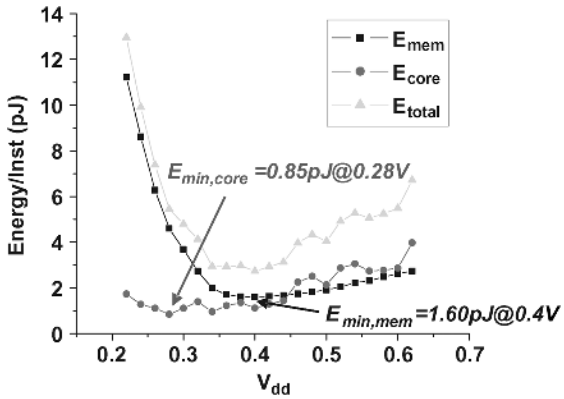



Fig. 15. Core and memory energy consumption as a function of $V_{\rm dd}$.

Fig. 13 shows the maximum operating frequency as a function of supply voltage measured across four chips. As expected, we observe an exponential relationship between $V_{\rm dd}$ and operating frequency. The operating frequency drops rapidly in the subthreshold region where $V_{\rm dd}$ becomes less than the threshold voltage ($\sim$400 mV).

In Fig. 14, we plot the measured change in energy consumption per instruction with supply voltage for one measured die. Note that the $V_{\rm min}$ is still determined by total energy consumption while the processor is executing instructions although total

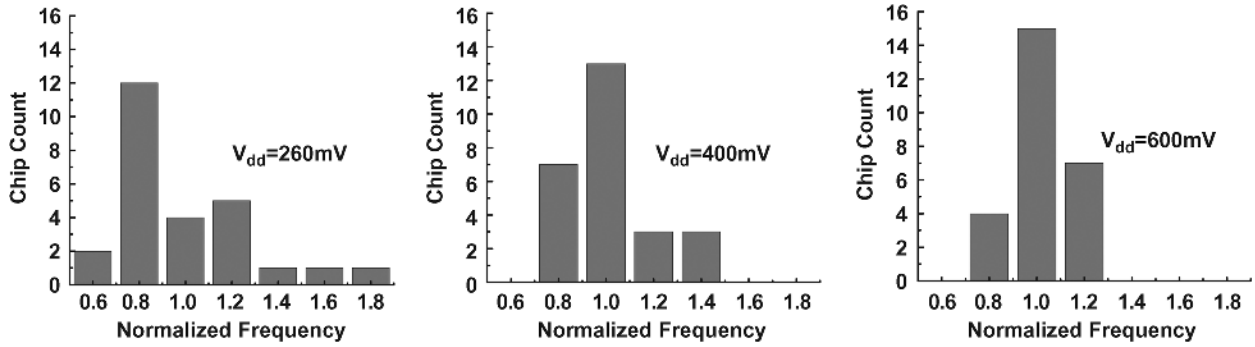



Fig. 16. Process variation across chips as a function of supply voltage.

energy is dominated by leakage at very low voltage and speed. The leakage energy increases rapidly as the operating voltage drops below the threshold voltage of ∼400 mV. The minimum energy $(E_{\min})$ occurs at ∼360 mV, where active energy (including short circuit current) and leakage have equal and opposite sensitivity to supply voltage, and leakage energy is ∼33% of the total energy. The nonmonotonic results comes from the operating frequency measurement, which is not perfectly exponential with $V_{\rm dd}$ in the subthreshold region.

In order to understand the relative contribution of different components, we have broken down the energy consumption between the core and the memory in Fig. 15. We still use energy/inst as our metric to be consistent with Fig. 14. Minimum energy operating voltage, $V_{\min}$ for the core is found to be 280 mV while that for the memory is much higher at 400 mV. This is attributable to the fact that the switching activity in the memory is considerably lower as compared to that of the core, thereby increasing the percentage of leakage energy to the total energy in the memory. On the other hand, a much higher switching activity in the core shifts its $V_{\min}$ to a lower value. It is also important to note that the minimum energy for the memory, $E_{\min,\rm mem}$ is almost twice that of the core, $E_{\min,\rm core}$. This shows that the core design is energy efficient but the overall system is limited by memory design. Recent work in the design of robust, energy-efficient subthreshold memories is promising for use in the Subliminal Processor [24]–[27]. Additionally, since the core and the memory have different optimal operating points $(V_{\min})$, it may be beneficial to design a system with separate supply and threshold voltages for the core and the memory [28]. Separate supply and threshold voltages would allow the core and the memory to operate at their respective most energy-efficient points, thereby resulting in additional energy savings.

In Fig. 16, we show the measured operating frequency distribution of 26 chips at three voltages: 260, 400, and 600 mV. Table III shows the corresponding $3\sigma/\mu$ values which range from 29.6% to 85.5%. This variation is ∼2.63× lower compared to the variation of individual devices, as discussed earlier in Fig. 6, and is due in part to the high logic depth in the Subliminal Processor.

Figs. 17 and 18 show the $V_{\min}$ and $E_{\min}$ distributions of the Subliminal Processor over 26 measured chips. The $V_{\min}$ ranges from 340 to 420 mV, with a mean and standard deviation of 378 and 21.4 mV, respectively ($3\sigma/\mu$ is 22.8%). The $E_{\min}$ per instruction ranges from 2.6 pJ/instruction to 3.4 pJ

TABLE III
MEASURED FREQUENCY DISTRIBUTION OF 26 CHIPS AT DIFFERENT SUPPLY VOLTAGES

| Vdd | μ (Freq) | 3σ/μ (Freq) |
|---|---|---|
| 0.260 | 84.66 kHz | 85.5% |
| 0.400 | 1.529 MHz | 49.8% |
| 0.600 | 6.759 MHz | 29.6% |

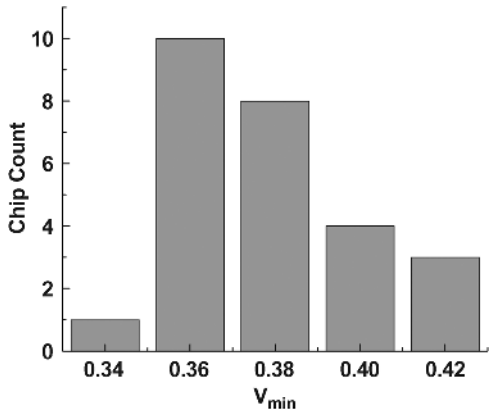



Fig. 17. Minimum energy voltage $(V_{\min})$ distribution.

with a mean of 3.0 pJ and standard deviation of 0.170 pJ ($3\sigma/\mu$ is 16.99%). However, to obtain this minimum energy operation, each die must operate at its individual $V_{\min}$ and operation frequency which requires adaptive *frequency* and *voltage tuning* of each die, as discussed in Section IV-A. Recall from Fig. 9 that the energy distributions remains nearly optimal when $V_{\rm dd}$ is fixed across all dies while clock period is selected individually for each die. This is confirmed in Fig. 18, which shows the energy distribution when all dies operate at a the minimum delay and a fixed $V_{\rm dd}$ equal to $\mu(V_{\min}) = 378$ mV. The resulting mean energy $\mu(E_{\min}) = 3.19$ pJ (a 6% increase) and standard deviation $\sigma(E_{\min}) = 0.21$ pJ (a 23% increase) are nearly the same as the original distribution. Fig. 18 also shows the energy distribution when all dies are operated at a fixed, worst-case $(\mu + 3\sigma)$ frequency as well as a fixed $V_{\min} = 378$ mV. In this case, $\mu(E_{\min}) = 3.72$ pJ (a 24% increase) and $\sigma(E_{\min}) = 0.283$ pJ (a 66% increase), a much more significant increase as compared to the original distribution. This confirms our earlier observation that adaptive voltage tuning is only marginally beneficial for maximum energy efficiency in subthreshold

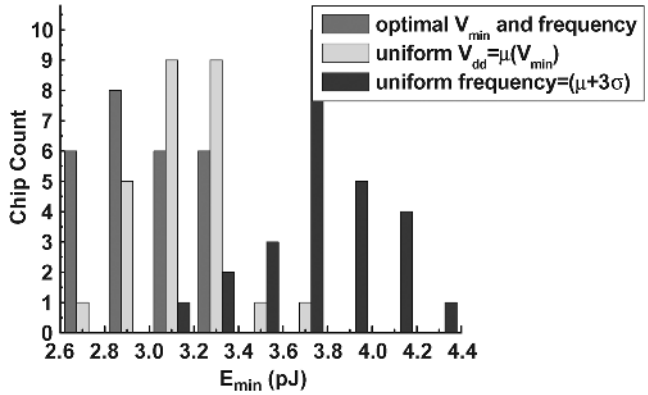



Fig. 18. Minimum energy consumption distribution.

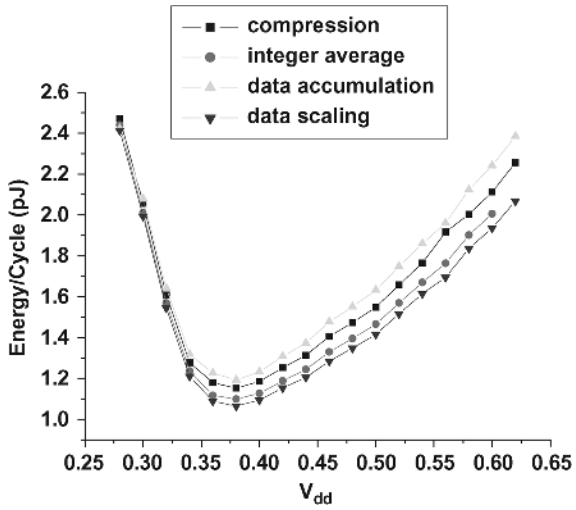



Fig. 19. Energy efficiency with $V_{\rm dd}$ for four sensor applications.

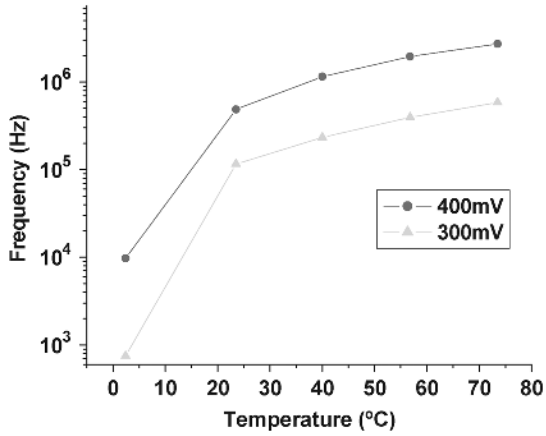



Fig. 20. Frequency variation with temperature at different supply voltages.

operation. Rather, more significant energy savings are obtained by applying adaptive frequency tuning in subthreshold design.

The energy consumption of the Subliminal Processor for four different sensor application programs is shown in Fig. 19. The variation in their individual energy demands was reduced in subthreshold operation due to the increased contribution of application-independent leakage current at lower operating voltages. Furthermore, the applications showed nearly identical $V_{\rm min}$ for all the applications, reinforcing the earlier finding that dynamic adjustment of $V_{\rm min}$ from die-to-die or during operation is only marginally useful.

Fig. 20 shows the frequency–temperature plot for two different supply voltages. As expected, the sensitivity of frequency to temperature is appreciable in subthreshold region [31]. Measured sensitivity was found to be $\sim 185\%/10$ °C at an operating voltage of 300 mV.

The Subliminal Processor was validated to be fully functional in the range of 1.2 V to 200 mV. The processor consumes 0.85 pJ/instruction at 0.04 MIPS and 1.2 pJ at 0.5 MIPS.

## VI. Conclusion

In this paper, we examined the landscape of energy optimization for sensor processors. We demonstrated that subthreshold-voltage circuit design is a compelling technique for energy-efficient sensor network processing. Based on the architecture- and circuit-level optimizations, we proposed the Subliminal Processor, a general-purpose sensor processor optimized for energy-efficient operation in subthreshold regimes. The Subliminal Processor is fully functional from a nominal supply voltage of 1.2 V down to 200 mV. Silicon measurements demonstrate that the processor attains the maximum energy efficiency of 2.6 pJ/instruction at 360 mV, operating at a frequency of 833 kHz. We also analyzed the variation in frequency and optimal voltage across different chips and found that the tuning of operating frequency is far more important in subthreshold voltage than is the tuning of supply voltage.

## Acknowledgment

This study presents a detailed description of the preliminary works published in the Proceedings of the International Symposium on Computer Architecture 2005, the Proceedings of the 41st IEEE/ACM Design Automation Conference 2004, and the Proceedings of the IEEE Symposium on VLSI Circuits 2006. The submitted study presents a more detailed literature review of subthreshold design. The study provides a detailed discussion on the circuit implementation, variability-related issues in subthreshold design and measurement results. The study also includes more measurement results on the implemented prototype that were not previously published.

## References

[1] J. L. Hill, "System architecture for wireless sensor networks," Ph.D. dissertation, Comput. Sci. Dept., Univ. California, Berkeley, 2003.

[2] A. Mainwaring, J. Pilastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proc. 1st ACM Int. Workshop Sensor Netw. Appl.*, 2002, pp. 88–97.

[3] N. Xu, S. Rangwala, K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, "A wireless sensor network for structural monitoring," in *Proc. 2nd Int. Conf. Embedded Netw. Sens. Syst.*, 2002, pp. 13–24.

[4] J. Rabaey, J. Ammer, T. Karalar, B. O. S. Li, M. Sheets, and T. Tuan, "Picoradios for wireless sensor networks: The next challenge in ultra-low-power design," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2002, pp. 200–201.

[5] B. A. Warneke and K. S. Pister, "An ultra-low energy microcontroller for smart dust wireless sensor networks," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2004, pp. 316–317.

[6] L. Nazhandali, J. Olson, A. Reeves, M. Minuth, B. Zhai, R. Helfand, S. Pant, T. Austin, and D. Blaauw, "Energy optimization of subthreshold-voltage sensor network processors," in *Proc. Int. Symp. Comput. Archit.*, 2005, pp. 197–207.
[7] V. Ekanayake, C. Kelly, and R. Manohar, "An ultra low-power processor for sensor networks," in *Proc. 11th Int. Conf. Archit. Support Program. Languages Oper. Syst.*, 2004, pp. 27–36.
[8] F. Koushanfar, V. Prabhu, M. Potkonjak, and J. Rabaey, "Processors for mobile applications," in *Proc. IEEE Int. Conf. Comput. Des.*, 2000, pp. 603–608.
[9] C. Schurgers and M. B. Srivastava, "Energy efficient routing in wireless sensor networks," in *Proc. Military Commun. Conf.*, Oct. 2001, pp. 357–361.
[10] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," in *Proc. 41st IEEE/ACM Des. Autom. Conf.*, 2004, pp. 868–873.
[11] S. Hanson, B. Zhai, K. Bernstein, D. Blaauw, A. Bryant, L. Chang, K. Das, W. Haensch, E. Nowak, and D. Sylvester, "Ultra-low voltage minimum energy CMOS," *IBM J. Res. Dev.*, vol. 50, no. 4/5, pp. 469–490, Jun. 2006.
[12] B. Zhai, L. Nazhandali, J. Olson, A. Reeves, M. Minuth, R. Helfand, S. Pant, D. Blaauw, and T. Austin, "A 2.60 pJ/Inst subthreshold sensor processor for optimal energy efficiency," in *Proc. IEEE Symp. VLSI Circuits*, 2006, pp. 154–155.
[13] C. H.-I. Kim, H. Soeleman, and K. Roy, "Ultra-low-power DLMS adaptive filter for hearing aid applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 11, no. 6, pp. 1058–1067, Dec. 2003.
[14] A. Wang and A. Chandrakasan, "A 180 mV FFT processor using subthreshold circuit techniques," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2004, pp. 292–C293.
[15] B. H. Calhoun and A. P. Chandrakasan, "Ultra-dynamic voltage scaling (UDVS) using subthreshold operation and voltage dithering," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 238–245, Jan. 2006.
[16] J. T. Kao, M. Miyazaki, and A. P. Chandrakasan, "A 175-mV multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1545–1554, Nov. 2002.
[17] "Online resource for information on data compression," 2004. [Online]. Available: http://www.data-compression.info/Algorithms/RLE,
[18] D. J. Wheeler and R. M. Needham, "TEA, a tiny encryption algorithm," *Lecture Notes Comput. Sci.*, vol. 1008, pp. 363–366, 1995.
[19] J. Kwong and A. P. Chandrakasan, "Variation-driven device sizing for minimum energy subthreshold circuits," in *Proc. IEEE Int. Symp. Low Power Electron. Des.*, 2006, pp. 8–13.
[20] B. H. Calhoun, A. Wang, and A. P. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *IEEE J. Solid-State Circuits*, vol. 40, no. 9, pp. 1778–1786, Sep. 2005.
[21] A. Raychowdhury, B. C. Paul, S. Bhunia, and K. Roy, "Computing with subthreshold leakage: Device/circuit/architecture co-design for ultralow-power subthreshold operation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 13, no. 11, pp. 1213–1224, Nov. 2005.
[22] B. Zhai, S. Hanson, D. Blaauw, and D. Sylvester, "Analysis and mitigation of variability in subthreshold design," in *Proc. IEEE Int. Symp. Low Power Electron. Des.*, 2005, pp. 20–25.
[23] C.-Y. Chong, "Sensor networks: Evolution, opportunities and challenges," *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.
[24] B. H. Calhoun and A. Chandrakasan, "A 256 kb subthreshold SRAM in 65 nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2006, pp. 628–629.
[25] B. Zhai, D. Blaauw, D. Sylvester, and S. Hanson, "A sub-200 mV 6T SRAM in 130 nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2007, pp. 332–333.
[26] T. Kim, J. Liu, J. Keane, and C. H. Kim, "A high-density subthreshold SRAM with data-independent bitline leakage and virtual ground replica scheme," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2007, pp. 329–330.
[27] N. Verma and A. P. Chandrakasan, "A 65 nm 8T sub-V, SRAM employing sense-amplifier redundancy," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2007, pp. 327–328.
[28] B. Zhai, R. Dreslinski, D. Blaauw, T. Mudge, and D. Sylvester, "Energy efficient near-threshold chip multi-processing," in *Proc. IEEE Int. Symp. Low Power Electron. Des.*, 2007, pp. 32–37.
[29] Y. Ramadass and A. Chandrakasan, "Minimum energy tracking loop with embedded DC-DC converter delivering voltages down to 250 mV in 65 nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2007, pp. 64–65.
[30] Y. Ikenaga, M. Nomura, Y. Nakazawa, and Y. Nagihara, "An optimal supply voltage determiner circuit for minimum energy operations," in *Proc. IEEE Symp. VLSI Circuits*, 2007, pp. 156–157.
[31] H. Soeleman, K. Roy, and B. Paul, "Robust subthreshold logic for ultra-low power operation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 9, no. 1, pp. 90–100, Feb. 2001.
[32] J. Kwong, Y. Ramadass, N. Verma, M. Koesler, K. Huber, H. Moormann, and A. Chandrakasan, "A 65 nm sub-Vt microcontroller with integrated SRAM and switched-capacitor DC-DC converter," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2008, pp. 318–319.

**Bo Zhai** received the B.S. degree in microelectronics from Peking University, China, in 2002, and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, in 2004 and 2007.

He is a currently a Senior Design Engineer with Advanced Micro Devices, Austin, TX. His research focuses on low power VLSI design.

**Sanjay Pant** (M'08) received the B.Tech degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 2001, and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, in 2004 and 2007.

Currently, he is a Senior Design Engineer with the Advanced Power Technology Group, Advanced Micro Devices, Fort Collins, CO. His research interests include low power VLSI design and signal integrity issues in power distribution networks.

**Leyla Nazhandali** received the B.S. degree (honors) in electrical engineering from Sharif University of Technology, Iran, in 2000, and the M.S. and Ph.D. degrees in computer engineering from the Advanced Computer Architecture Laboratory (ACAL), University of Michigan, Ann Arbor, in 2002 and 2006, respectively.

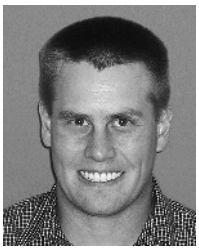
She is currently an Assistant Professor with the Bradley Department of Electrical and Computer Engineering, Virginia Institute of Technology, Blacksburg.

Dr. Nazhandali was a recipient of the prestigious National Science Foundation CAREER Award in 2008 for her proposed work entitled, "Overcoming Power Challenges in Embedded System Design with Subthreshold-Voltage Technology." She is also the winner of IEEE Real World Engineering Projects Contest, for her project "smart vehicles," where she has developed a hands-on project for freshman students in order to introduce them to the benefits of computer engineering, especially embedded systems, for the society. Among her other awards, she received a Riethmiller Fellowship Award for 2005-2006 to conduct research with applications in biomedicine. In 2005, she won the 1st place in Computer Science and Engineering Honors competition at the University of Michigan. In 1996, she was ranked 44th in Iran's National College Entrance Exam in a field of more than 150,000 applicants. Her research interests are in low-power energy-constrained embedded system design, subthreshold-voltage architectures, secure embedded hardware design and engineering education focusing on attraction and retention of underrepresented groups in computer engineering.

**Scott Hanson** received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, in 2004, 2006, and 2009, respectively.

He is currently a research fellow in electrical engineering with the University of Michigan. His research interests include low voltage circuit design for ultra-low energy applications, variation tolerant circuit design, and energy efficient high performance circuit design.

Dr. Hanson was a recipient of an SRC fellowship.



**Javin Olson** received the B.S. degree in computer engineering from Northwestern University, Evanston, IL, in 1999, and the M.S. degree in electrical engineering from the University of Michigan, Ann Arbor, in 2005.

He is currently employed in the Microprocessor Design Group, Advanced Micro Devices, Boston, MA. While attending the University of Michigan, he was a Research Assistant with the Advanced Computer Architecture Lab under the direction of Prof. T. Austin.

**Anna Reeves,** photograph and biography not available at time of publication.

**Michael Minuth,** photograph and biography not available at time of publication.

**Ryan Helfand,** photograph and biography not available at time of publication.

**Todd Austin** (M'88) received the Ph.D. degree in computer science from University of Wisconsin, Madison, in 1996.

He is an Associate Professor with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. Prior to joining academia, he was a Senior Computer Architect with Intel's Microcomputer Research Labs, a product-oriented research laboratory in Hillsboro, OR. His research interests include computer architecture, compilers, computer system verification, and performance analysis tools and techniques.

**Dennis Sylvester** (S'95–M'00–SM'04) received the B.S. degree in electrical engineering (*summa cum laude*) from the University of Michigan, Ann Arbor, in 1995, the M.S. and Ph.D. degrees in electrical engineering from University of California, Berkeley, in 1997 and 1999, respectively.

Currently, he is an Associate Professor with the Department of Electrical Engineering, University of Michigan, Ann Arbor. He previously held research staff positions with the Advanced Technology Group, Synopsys, Mountain View, CA, and with Hewlett-Packard Laboratories, Palo Alto, CA. He has published numerous articles along with one book and several book chapters in his field of research, which includes low-power circuit design and design automation techniques, design-for-manufacturability, and on-chip interconnect modeling. He also serves as a consultant and technical advisory board member for several electronic design automation firms in these areas.

Dr. Sylvester was a recipient of an NSF CAREER Award, the 2000 Beatrice Winner Award at ISSCC, a 2004 IBM Faculty Award, several Best Paper Awards and nominations, the ACM SIGDA Outstanding New Faculty Award, the 1938E Award from the College of Engineering Award for teaching and mentoring, and the Henry Russel Award, which is the highest award given to faculty at the University of Michigan. He has served on the technical program committee of numerous design automation and circuit design conferences and was general chair of the 2003 ACM/IEEE System-Level Interconnect Prediction (SLIP) Workshop and 2005 ACM/IEEE Workshop on Timing Issues in the Synthesis and Specification of Digital Systems (TAU). He is currently an Associate Editor for the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS. He also helped define the circuit and physical design roadmap as a member of the International Technology Roadmap for Semiconductors (ITRS) U.S. Design Technology Working Group from 2001 to 2003. He is a member of ACM, American Society of Engineering Education, and Eta Kappa Nu. His dissertation research was recognized with the 2000 David J. Sakrison Memorial Prize as the most outstanding research in the UC-Berkeley EECS Department.

**David Blaauw** received the B.S. degree in physics and computer science from Duke University, Durham, NC, in 1986, and the Ph.D. degree in computer science from the University of Illinois, Urbana, in 1991.

Until August 2001, he worked for Motorola, Inc., Austin, TX, where he was the manager of the High Performance Design Technology Group. Since August 2001, he has been on the faculty of the University of Michigan, Ann Arbor, where he is a Professor. His work has focussed on VLSI design with particular emphasis on ultra low power and high performance design.

Dr. Blaauw was the Technical Program Chair and General Chair for the International Symposium on Low Power Electronic and Design and was the Technical Program Co-Chair and member of the Executive Committee the ACM/IEEE Design Automation Conference. He is currently a member of the ISSCC Technical Program Committee.