Low Voltage Circuit Design Techniques for Cubic-Millimeter Computing

by

Scott McLean Hanson

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Electrical Engineering) in The University of Michigan 2009

Doctoral Committee:

Associate Professor Dennis M. Sylvester, Chair
Professor David Blaauw
Professor Kensall D. Wise
Assistant Professor David D. Wentzloff
Kerry Bernstein, International Business Machines Corporation

© Scott McLean Hanson 2009

To my family

Acknowledgements

Graduate school has been fascinating, exciting, often exhausting, and ultimately an incredible learning experience. Throughout graduate school, I have been supported by a strong network both inside and outside the University of Michigan.

Inside the University, I have worked with an intelligent, creative group of professors and students. Many of these people have directly contributed to this work. The low voltage overview described in Chapter 2 is drawn from work done originally by Bo Zhai. Mingoo Seok offered valuable feedback about the work covered in Chapter 3. The Subliminal Processor, which is discussed in Chapter 4, was a joint effort by a long list of students including: Bo Zhai, Leyla Nazhandli, Brian Cline, Meghna Singhal, Javin Olson, Michael Minuth, Kevin Zhou, and Mingoo Seok as well as several faculty members: Dennis Sylvester, David Blaauw, and Todd Austin. Mingoo Seok, Yu-Shiang Lin, and I worked closely together to develop the Phoenix Processor (Chapter 5) under the guidance of David Blaauw and Dennis Sylvester. Fellow students Yoonmyung Lee, Rach Liu, Zhiyoong Foo, and Daeyeon Kim also played important roles in the design and test of the Phoenix Processor. Finally, Zhiyoong Foo helped design on-chip test infrastructure for the low voltage image sensor test-chip. I would also like to thank the Semiconductor Research Corporation for funding my fellowship and Kerry Bernstein for serving as my mentor.

Outside the University, my family, fiancée and friends have offered unwavering support. I am fortunate to have such a strong support group, and I thank them all.

Table of Contents

Dedication ii
Acknowledgements iii
List of Figures vi
List of Tables xi
Abstract xii
Chapter 1: Introduction 1
Chapter 2: Motivating Low Voltage Operation 12
Chapter 3: Device Scaling in Low Voltage Circuits 21
Chapter 4: The Subliminal Processor 48
Chapter 5: The Phoenix Processor 74
Chapter 6: An Ultra-Low Voltage CMOS Image Sensor 101
Chapter 7: Conclusion 116
References 119

List of Figures

Figure 1.1: An implantable intra-ocular pressure sensor (courtesy of Y-S. Lin) 3

Figure 1.2: The canonical $1mm^3$ computer..4

Figure 1.3: NFET drain current as a function of gate-source voltage 5

Figure 2.1: Simulated NFET drain current as a function of gate voltage in a 0.13µm technology..14

Figure 2.2: (a) Power consumed by an inverter chain as a function of supply voltage (V_{dd}) (b) Energy consumed per switching operation by the same inverter chain as a function of V_{dd} 15

Figure 2.3.: Delay variability (σ/µ) as a function of supply voltage (65 nm technology) 18

Figure 2.4: Variation in worst-case (µ+3σ) V_{min} and E_{min} for an inverter chain of length n gates. The relative increases in $V_{min}$and E_{min} are less severe at large n (0.13µm technology) 18

Figure 3.1: (a) A device cross-section showing scaling parameters (b) Doping profile for a 90nm NFET..24

Figure 3.2: NFET inverse sub-V_{th} slope and on-current to off-current ratio.....................27

Figure 3.3: NFET on-current 28

Figure 3.4: (a) Definition of SNM (b) Simulated SNM for a scaled inverter....................29

Figure 3.5: Simulated delay for a scaled inverter 31

Figure 3.6: Simulated energy/cycle and V_{min} for a chain of 30 inverters with α=0.1 33

Figure 3.7: S_S as a function of gate length for a 45nm device 34

Figure 3.8: Energy and delay factors for a 45nm device 34

Figure 3.9: NFET L_{poly} and S_S for sub-V_{th} and super-V_{th} scaling strategies 37

Figure 3.10: Simulated SNM for an inverter under super-V_{th} and sub-V_{th} scaling 39

Figure 3.11: Simulated delay for an inverter at V_{dd}=250mV under super-V_{th} and sub-V_{th} scaling 39

Figure 3.12: Simulated energy and V_{min} under super-V_{th} scaling and sub-V_{th} scaling......41

Figure 3.13: (a) RDF V_{th} variability model in a 65nm device closely matches gate area dependence in Eq. 9. (b) V_{th} is approximately linear with dopant count (c) SRAM test circuit for measuring SNM and I_{read}/I_{leak}. Node voltages during hold and read conditions are also shown...42

Figure 3.14: Simulated SNM in a 6T SRAM cell at V_{dd}=250mV under three different device optimization strategies: (1) sub-V_{th} optimized device, (2) unoptimized super-V_{th} device with minimum length, and (3) unoptimized super-V_{th} device with the same length as case (1) 43

Figure 3.15: Read failure probability for a single SRAM cell under different device optimization strategies at V_{dd}=250mV. Failure is defined as the point where the read SNM drops below 6% of V_{dd} (15mV 45

Figure 3.16: Ratio of read-current to pass-transistor leakage in a 6T SRAM at V_{dd}=250mV under super-V_{th} scaling and sub-V_{th} scaling. I_{read}/I_{leak} is proportional to the maximum number of bits per bitline and is therefore closely tied to SRAM area 46

Figure 4.1: (a) System-level diagram of the 8-bit subthreshold processor (b) CPU implementation details...50

Figure 4.2: Effective NFET resistance as a function of V_{dd}. The resistances of wires of several alternative materials are included for reference (with 100µm length and widths from inset) 52

Figure 4.3: The 8-bit subthreshold processor was fabricated in a 0.13µm technology. Three CPU variants are shown 53

Figure 4.4: Frequency and energy measurements for a typical die as functions of V_{dd}.....54

Figure 4.5: (a) The simulated ratio of NFET on-current ($I_{on,NFET}$) to PFET on-current ($I_{on,PFET}$) at two voltages (b) The simulated high static noise margins (SNM) at two voltages for an inverter with $W_{PFET}=2\cdot W_{NFET}$...57

Figure 4.6: Simulated NFET V_{th} and I_{on} as functions of V_{off}...59

Figure 4.7: (a) Simulated energy consumption for a chain of 30 inverters at V_{dd}=300mV as a function of V_{diff} (b) Simulated energy and delay for the same inverter chain as functions of V_{off} 61

Figure 4.8: (a) Energy and $V_{dd,limit}$ as functions of V_{diff} for a typical die (b) $V_{dd,limit}$ distribution for 20 dies with and without body biasing. The mean $V_{dd,limit}$ reduces from 221mV to 168mV, a 24% improvement 62

Figure 4.9: Energy and frequency as functions of body bias offset for a typical die 64

Figure 4.10: Energy and frequency distributions for 20 dies measured at V_{dd}=300mV 64

Figure 4.11: Temperature sensitivity of energy and frequency for a typical die at V_{dd}=300mV 65

Figure 4.12: Simulated energy and frequency for an inverter chain subjected to voltage scaling and body biasing. Data is plotted for switching activities of 1, 0.2, and 0.05 68

Figure 4.13: A comparison of energy and frequency measurements for variable body bias and variable V_{dd} systems 69

Figure 4.14: Simulated on-current for an NFET as a function of total device capacitance. The trade-off is shown for both gate width and gate length sizing 71

Figure 4.15: Energy and frequency for three sizing strategies for V_{dd}=280-400mV 72

Figure 5.1: The Phoenix Processor 75

Figure 5.2: A typical power gating switch 79

Figure 5.3: Footer allocation in the Phoenix Processor 80

Figure 5.4: CPU diagram 81

Figure 5.5: Distribution of temperature in Muskegon, MI in 2006 [75] represented as the difference between temporally adjacent measurements 84

Figure 5.6: Hardware support for compression 86

Figure 5.7: Memory support for compression 88

Figure 5.8: Proposed ultra-low standby power SRAM cell 89

Figure 5.9: Effectiveness of (a) stack forcing and (b) gate length biasing for leakage reduction 89

Figure 5.10: Memory column diagram showing completion detection 90

Figure 5.11: Phoenix Processor die photo 92

Figure 5.12: Measured frequency and energy consumption 94

Figure 5.13: Measured frequency distribution for 13 dies at V_{dd}=0.5V 95

Figure 5.14: Measured active mode power distribution at 60 kHz for 13 dies at V_{dd}=0.5V 95

Figure 5.15: Measured standby mode power distribution for 13 dies at V_{dd}=0.5V 95

Figure 5.16: Measured (a) frequency and (b) standby leakage as functions of CPU footer width 97

Figure 5.17: Total energy consumption assuming 1000 instructions are executed every 10 minutes 97

Figure 5.18: Measured (a) frequency and (b) power as functions of temperature 98

Figure 5.19: Computed time profiles of (a) energy and (b) memory size for a temperature measurement routine 99

Figure 6.1: A conventional 3T active pixel sensor design 103

Figure 6.2: Column architecture and timing diagram for a read operation 104

Figure 6.3: Low voltage pixel architecture and comparator voltage transfer characteristic 105

Figure 6.4: Die photo 106

Figure 6.5: Test setup 107

Figure 6.6: Mean responsivity over 100 frames 109

Figure 6.7: Mean pixel value as a function of integration time over 100 frames 109

Figure 6.8: Signal-to-noise ratio as a function of incident light 111

Figure 6.9: Peak signal-to-noise ratio as a function of V_{dd} 111

Figure 6.10: Power at V_{dd}=0.6V as a function of incident light 113

Figure 6.11: Power at maximum SNR as a function of V_{dd} 113

Figure 6.12: Image capture test setup 114

Figure 6.13: (a) Actual 256x256 8-bit image (b) Image downsampled to 128x128 (c) Image captured by CMOS image sensor 115

List of Tables

Table 3.1: NFET parameters under super-V_{th} scaling 25

Table 3.2: NFET parameters under sub-V_{th} scaling 36

Table 5.1: Instruction set architecture overview...82

Abstract

Cubic-millimeter computers complete with microprocessors, memories, sensors, radios and power sources are becomingly increasingly viable. Power consumption is one of the last remaining barriers to cubic-millimeter computing and is the subject of this work. In particular, this work focuses on minimizing power consumption in digital circuits using low voltage operation.

Chapter 2 includes a general discussion of low voltage circuit behavior, specifically that at subthreshold voltages. In Chapter 3, the implications of transistor scaling on subthreshold circuits are considered. It is shown that the slow scaling of gate oxide relative to the device channel length leads to a 60% reduction in I_{on}/I_{off} between the 90nm and 32nm nodes, which results in sub-optimal static noise margins, delay, and power consumption. It is also shown that simple modifications to gate length and doping can alleviate some of these problems.

Three low voltage test-chips are discussed for the remainder of this work. The first test-chip implements the Subliminal Processor (Chapter 4), a sub-200mV 8-bit microprocessor fabricated in a 0.13µm technology. Measurements first show that the Subliminal Processor consumes only 3.5pJ/instruction at V_{dd}=350mV. Measurements of 20 dies then reveal that proper body biasing can eliminate performance variations and reduce mean energy substantially at low voltage. Finally, measurements are used to

explore the effectiveness of body biasing, voltage scaling, and various gate sizing techniques for improving speed.

The second test-chip implements the Phoenix Processor (Chapter 5), a low voltage 8-bit microprocessor optimized for minimum power operation in standby mode. The Phoenix Processor was fabricated in a 0.18µm technology in an area of only 915x915µm^2. The aggressive standby mode strategy used in the Phoenix Processor is discussed thoroughly. Measurements at V_{dd}=0.5V show that the test-chip consumes 226nW in active mode and only 35.4pW in standby mode, making an on-chip battery a viable option.

Finally, the third test-chip implements a low voltage image sensor (Chapter 6). A 128x128 image sensor array was fabricated in a 0.13µm technology. Test-chip measurements reveal that operation below 0.6V is possible with power consumption of only 1.9µW at 0.6V. Extensive characterization is presented with a specific emphasis on noise characteristics and power consumption.

Chapter 1

Introduction

The miniaturization of electronics has launched a wireless sensing revolution that is gaining traction in both industry and academia. Complex wireless systems can be packaged in volumes on the order of several cubic centimeters, enabling pervasive sensing. Companies are marketing wireless soil moisture monitors for crop irrigation [1], compact wireless tire pressure sensors [2], and even tiny pressure sensors for monitoring the integrity of stent grafts [3]. Academic researchers have made more ambitious demonstrations ranging from wireless neural monitoring and stimulation [4][5] to complex gas analysis in ultra-small form factors [6].

Even as new applications emerge for wireless sensing, the miniaturization of electronics continues. Innovations in microprocessors, radios, sensors, actuators, packaging, and power sources will soon take us from the cubic-centimeter domain to the cubic-millimeter ($1mm^3$) domain. In $1mm^3$ computing, computers may be embedded virtually anywhere. They may be woven into clothing, implanted in the body, and set in construction materials. In light of such flexibility, it becomes critically important to ask: who needs a $1mm^3$ computer and why?

1.1 The Need for the 1mm^3 Computer

The promise of 1mm^3 computing is perhaps most apparent in medicine. Advances in the semiconductor industry have historically been catalysts for tremendous progress in medicine, enabling devices ranging from pacemakers to neurostimulators to continuous blood glucose monitors. Despite this past success, there is a continuing drive for smaller devices to address increasingly delicate problems.

For example, the diagnosis and treatment of glaucoma (open angle glaucoma and angle closure glaucoma are expected to affect 60.5 million people worldwide by 2010 [7]) requires periodic measurements of pressure in the eye (intra-ocular pressure). Intra-ocular pressure is currently monitored directly by a doctor, requiring frequent trips to the doctor's office to ensure sufficient temporal resolution [14]. An intra-ocular pressure sensor (Figure 1.1) with a MEMS pressure sensor, microprocessor, memory, radio and power source implanted in the eye would reduce both cost and time investment and would increase the temporal resolution of pressure measurements. Size is of the utmost importance in this example since implant damage to the patient must be minimized. Miniaturized computers are similarly useful for a wide range of medical monitoring devices including intra-cardiac pressure sensors for patients with congestive heart failure, intra-cranial pressure sensors for patients with hydrocephalus, and stent graft structural monitors for patients affected by abdominal aortic aneurysms.

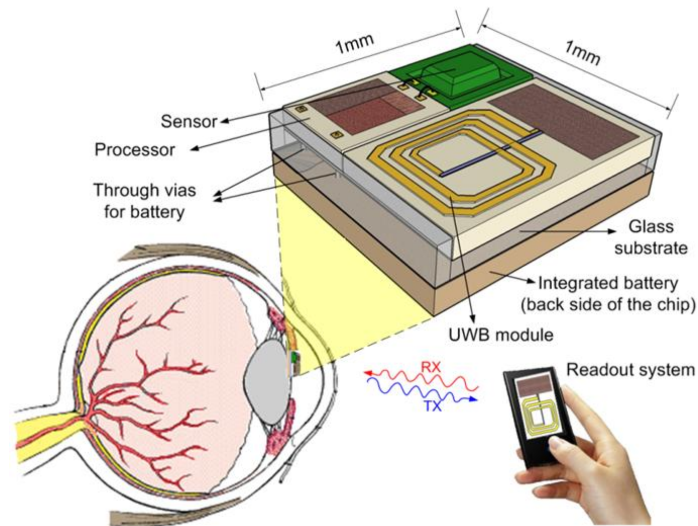


Figure 1.1: An implantable intra-ocular pressure sensor (courtesy of Y-S. Lin)

The broad field of wireless sensor networks will also be among the primary beneficiaries of 1mm^3 computing. Tiny sensors could be used for widespread battlefield surveillance [8][9], forest fire and flood detection [9], and energy management in buildings and homes [9]. While less volume-constrained than medical applications, these and other wireless sensor network applications will benefit from the low costs associated with small volume and high integration.

1.2 Challenges in 1mm^3 Computing

To meet the requirements of the aforementioned applications, 1mm^3 computers will include a microprocessor, a memory, sensors, actuators, wireless communication, and a power source in a compact package (Figure 1.2). Thanks to consistent reductions in transistor size, circuit components for computing and communication easily meet the size requirements set by 1mm^3 computing. Similarly, innovations in MEMS have enabled the design of tiny sensors and actuators well suited to 1mm^3 computing. Despite recent advances in the design of compact batteries [10] and energy scavenging devices [11],

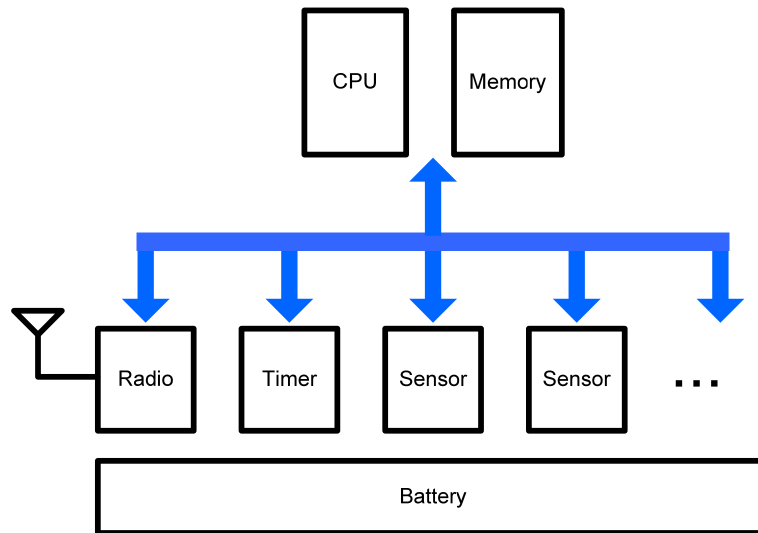


Figure 1.2: The canonical 1mm^3 computer

power sources cannot be easily miniaturized while also serving the power demands of circuit and MEMS components. Consider, for example, a thin film zinc/silver oxide battery with a capacity of 100µAh/cm^2 and output voltage of 1.55V [15]. Assuming that battery size is restricted to 1mm^2 and that energy density remains constant at small battery sizes, the average current draw of circuit and MEMS components must be 114pA (for power consumption of 177pW) to guarantee one year of battery life. The power consumption of the most energy efficient commercial microcontrollers (for example, the Texas Instruments MSP430 [12]) exceeds these limits by several orders of magnitude even in standby mode.

In light of power source limitations, power minimization is one of the clear challenges of 1mm^3 computing. Other notable challenges include compact antenna design and compact (and potentially biocompatible) packaging, but the focus of this work is power minimization. Particular emphasis is placed on minimizing power consumption in digital circuits. In the remainder of this chapter, the topic of energy minimization in

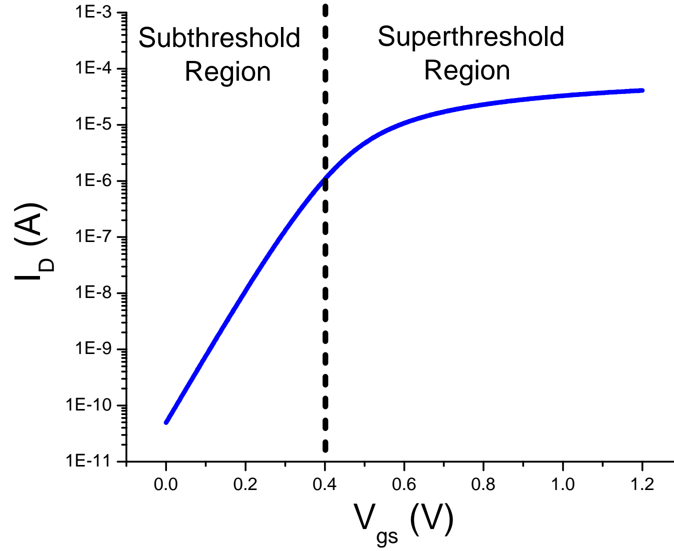


Figure 1.3: NFET drain current as a function of gate-source voltage

digital circuits is explored and the contributions of our research to the growing field of low voltage circuit design are discussed.

1.3 Addressing the Power Problem

The energy consumed by a digital circuit is typically broken into contributions from switching energy (E_{switch}) and leakage energy (E_{leak}), as shown in Equation 1.1, where C_S is the switched capacitance, V_{dd} is the supply voltage, α is the switching activity, I_{leak} is the total leakage current of the circuit, and t_p is the maximum delay across the circuit.

$$E_{total} = E_{switch} + E_{leak} = C_S \cdot V_{dd}^2 \cdot \alpha + I_{leak} \cdot V_{dd} \cdot t_p \qquad \text{EQ 1.1}$$

In a typical circuit operating at the nominal supply voltage, E_{switch} far exceeds E_{leak} (though E_{leak} generally grows with subthreshold leakage and gate leakage as technology scales [16]). Given the quadratic dependence of E_{switch} on V_{dd}, the most effective technique for reducing energy in a circuit is to reduce V_{dd}. Though it is clear that reducing V_{dd} will yield energy reductions, it is not obvious how far V_{dd} can be scaled. In general,

circuit designers assume that transistors turn off when the gate-source voltage drops below the threshold voltage (V_{th}) of the device. This is, of course, a very simplistic assumption since subthreshold leakage is non-negligible, as shown in Figure 1.3. With careful design, subthreshold leakage can be used to charge and discharge nodes in a digital circuit, suggesting that it is possible to use $V_{dd} < V_{th}$. Circuits operating in this region are called subthreshold circuits and are the focus of this work.

Aggressive voltage scaling into the subthreshold region poses a number of daunting challenges to circuit designers. The first of these issues is reduced performance (i.e., reduced operating frequency). As shown in Figure 1.3, subthreshold currents are much smaller than the strong inversion currents used in typical superthreshold ($V_{dd} > V_{th}$) circuits, so switching delays are increased by several orders of magnitude [17]. In many $1mm^3$ computing applications, this performance penalty is tolerable, but many applications, such as those requiring streaming media processing, may have more stringent deadlines for computation.

The reduced noise margins that come with subthreshold operation are also potentially problematic. Though coupling noise may reduce with voltage [17], noise from external components (for example, coupling noise due to a reference oscillator) may not scale with voltage and will create serious robustness problems.

The most significant problem facing subthreshold designers is an increased sensitivity to variability. Due to the exponential dependence of subthreshold current on V_{th}, V_{dd} and temperature, small process and environmental variations can lead to enormous noise margin, delay and energy fluctuations [18]. This is a particular challenge

for memory designers since dense SRAM arrays require extremely high yield and use small variation-prone devices aggressively [19].

1.4 Previous Work in Low Voltage Design

The limits of voltage scaling were first explored in 1972 [20]. Further research in digital subthreshold operation was limited until 1999, when the authors of [21] began to explore logic family selection for subthreshold circuits. Subsequently, a number of research groups have begun to focus their efforts on addressing the challenges of subthreshold operation.

The existence of an energy-optimal voltage (which typically lies in the subthreshold regime) was first noted in [23] and later in [24]. This was an important discovery because it showed 1) that subthreshold operation is typically energy optimal and 2) that scaling to the minimum functional voltage can actually increase energy consumption. Early hardware demonstrations of subthreshold circuits were presented in [25] and later in [26], where operation down to 180mV was achieved. The simulation-based conclusions regarding the energy-optimal voltage [23][24] were confirmed by hardware presented in [27]. The first subthreshold general microcontroller was presented in [28], and was shown to consume only 2.6pJ per instruction. These early research efforts proved that energy efficient subthreshold operation was possible but did not address the more important topics of variability and robust memory design.

Recently, the problems posed by robust memory design have begun to receive more attention. In [29] a simulation-based study explored the key sensitivities of SRAM noise margins in the subthreshold regime. The authors of [19][30] showed that a

modified SRAM cell using 8 transistors could achieve robust operation from nominal superthreshold voltages down to 0.41V. Several alternative SRAM cells have also been proposed with a specific focus on subthreshold operation [31][32][33][71][72].

The recent burst of activity in subthreshold circuits research has not yet addressed several key needs. Variability has still not been explored adequately to permit widespread commercial use. The importance of standby power in low voltage $1mm^3$ systems has also not been addressed. Additionally, low voltage research has focused largely on conventional circuits (e.g., microprocessors, memories, etc.) and has not been expanded to include sensors, critical components in any $1mm^3$ computer.

1.5 Contributions of this Work

This work explores aggressive voltage scaling for robust energy efficient operation in $1mm^3$ computing systems. The study begins in Chapter 2 with a review of basic digital subthreshold circuit concepts. This review gives a detailed overview of the most important prior work in subthreshold circuit design with a particular emphasis on device characteristics, voltage selection, and variability. This work is drawn from several papers that I authored or co-authored, including [17] and [34], though most results are derived from results originally presented by Bo Zhai in [18] and [23].

Chapter 3 discusses the topic of device scaling in subthreshold circuits. MEDICI simulations are used to investigate the implications of transistor scaling on subthreshold circuits. In particular, it is shown that conventional scaling trends lead to sub-optimal noise margins, performance, and energy consumption and that simple changes to channel doping and gate length provide dramatic improvements to these parameters. The work in

Chapter 3 is derived from work originally presented in [77]. I developed each of the experiments described in that chapter, though Mingoo Seok offered valuable feedback throughout.

Given the foundations laid during Chapters 2 and 3, Chapter 4 is used to explore the design and test of the Subliminal Processor, an 8-bit subthreshold processor that was designed with the needs of $1mm^3$ computing in mind. Architectural and physical design decisions affecting energy efficiency are discussed in addition to test-chip measurements which demonstrate active energy of only 3.5pJ/instruction. Measurements of Subliminal offer new insights into variability control for subthreshold circuits. In particular, it is shown that control of the body bias can be used to effectively eliminate performance variation and can be used to reduce energy variations. Test chip measurements are also used to evaluate body biasing, voltage scaling, and various gate sizing techniques for improving performance (i.e., speed) in subthreshold circuits. The work in Chapter 4 was derived from [22], a joint collaboration with students including Bo Zhai, Leyla Nazhandli, Brian Cline, Meghna Singhal, Javin Olson, Michael Minuth, Kevin Zhou, and Mingoo Seok. During the design phase of this project, I was directly responsible for memory design, physical design of the processor, and gate sizing strategies. I designed the experiments described in Chapter 4 though Mingoo Seok and Kevin Zhou contributed significant time to testing chips.

The processor described in Chapter 4 is an important demonstration of robust low voltage operation but does not consider several important topics in $1mm^3$ computing, most notably standby power management. In Chapter 5, the Phoenix Processor, an 8-bit sensor processor, is introduced. Unlike previous work, the design of Phoenix is focused

primarily on standby power minimization (as opposed to active power minimization or performance maximization). Phoenix leverages a comprehensive standby mode strategy to minimize the impact of idle periods, which can be >99% of the total lifetime of a sensor node. Device-, circuit-, and architecture-level considerations in the Phoenix Processor are discussed in addition to energy and performance measurements. At only 35.4pW, the standby mode power consumption in the Phoenix Processor is several orders of magnitude lower than that of previously reported microprocessors in both academia and industry. The work described in Chapter 5 was originally presented in [66] and was the result of collaboration between multiple students including Mingoo Seok, Yu-Shiang Lin, Zhiyoong Foo, Nurrachman Liu, Yoonmyung Lee, and Daeyeon Kim. I was directly responsible for the design of the CPU, compression blocks, the power management unit, and the system bus. I also helped in the initial planning of memory blocks and co-led testing efforts with Mingoo Seok.

In Chapter 6, the topic of low voltage microprocessors is left behind and focus is instead shifted to a low voltage CMOS image sensor. While the topic of Chapter 6 is a clear divergence from conventional microprocessor design, sensors are an important part of any $1mm^3$ computer and merit significant attention. This work is the first demonstration of a near-threshold image sensor and test-chip measurements reveal that extremely low power consumption is achieved. I led both design and test efforts in the image sensor test-chip though Zhiyoong Foo made valuable contributions in implementing on-chip test infrastructure.

The research efforts described in this work have offered new insights on subthreshold circuit design and on the greater topic of $1mm^3$ computing. However, a number of

challenges remain and are discussed briefly in Chapter 7. Topics covered include the ever-present topic of variability in subthreshold circuits, low power radio design, and $1mm^3$ system integration.

To summarize, this work makes the following new contributions:

- Explores the implications of device scaling on subthreshold circuits and proposes an alternative device scaling strategy to counteract problems encountered at advanced technology nodes
- Uses a subthreshold test-chip (the Subliminal Processor) to demonstrate the effectiveness of body biasing for reducing variability in subthreshold circuits and to compare body biasing, voltage scaling, and various gate sizing techniques for improving performance in subthreshold circuits
- Demonstrates the effectiveness of aggressive standby mode power minimization in a low voltage test-chip (the Phoenix Processor), giving unprecedented standby power of only 35.4pW
- Discusses a new image sensor architecture optimized for low voltage, which measurements reveal to be the first near-threshold image sensor test-chip

Chapter 2

Motivating Low Voltage Operation

Due to the quadratic dependence of switching energy on V_{dd}, low voltage operation is an extremely effective technique for reducing energy in digital circuits. Recently, aggressive voltage scaling into the subthreshold ($V_{dd}<V_{th}$) regime has been explored for applications like 1mm^3 computing, where energy minimization is one of the primary design requirements. This chapter explores the theory behind low voltage operation.

The discussion of low voltage operation begins in Section 2.1 with a look at the basic device sensitivities observed in subthreshold circuits. The exponential dependence of subthreshold current on V_{dd}, V_{th}, and subthreshold slope make variability a pressing concern in low voltage circuits. The on-current to off-current ratio also reduces exponentially in the subthreshold regime, giving reduced noise margins and exponentially slower switching speeds in subthreshold circuits.

In light of these concerns at low V_{dd}, it is important to quantify the expected energy benefits of subthreshold operation. Section 2.2 reviews recent explorations of the energy characteristics of subthreshold circuits. In particular, the existence of an energy-optimal voltage and its energy implications are discussed.

The problems posed by variability are considered in Section 2.3. Several important sources of variation are considered and it is shown that systematic and random V_{th} variations pose the most serious problems for subthreshold energy, delay, and robustness. In addition, it is shown that the implications of V_{th} variability on subthreshold logic can be quite serious.

2.1 Subthreshold Device Sensitivities

As V_{dd} is reduced to manage power consumption, FETs make the transition from drift-dominated superthreshold (super-V_{th}) operation to diffusion-dominated subthreshold (sub-V_{th}) operation. The current in the subthreshold region is described by Equation 2.1, where W is the gate width, L_{eff} is the effective gate length, μ_{eff} is the effective mobility, C_d is the depletion capacitance, v_T is the thermal voltage, V_{gs} is the gate-source voltage, m is the subthreshold slope factor, and V_{ds} is the drain-source voltage.

$$I_{sub} = \frac{W}{L_{eff}} \cdot \mu_{eff} \cdot C_d \cdot v_T^2 \cdot e^{\left(\frac{V_{gs} - V_{th}}{m \cdot v_T}\right)} \cdot \left(1 - e^{-\frac{V_{ds}}{v_T}}\right) \quad \text{Equation 2.1}$$

A close inspection of Equation 1 reveals a great deal about the challenges of sub-V_{th} operation. Most notably, subthreshold current is exponentially dependent on V_{th}, V_{gs} (and therefore V_{dd}), m, and temperature (through v_T). Figure 2.1 confirms the exponential dependences on V_{gs} and V_{th} for an NFET simulated in a 0.13µm process. The most important implication of this dependence is an increased sensitivity to process, voltage, and temperature variability in the subthreshold regime. For example, simulations show that a 100mV increase in V_{th} results in a ~14X change in current at the same V_{gs} and V_{ds}. A circuit-level exploration of variability in subthreshold circuits will be discussed later in this chapter. We also note in Equation 2.1 that subthreshold current has only linear

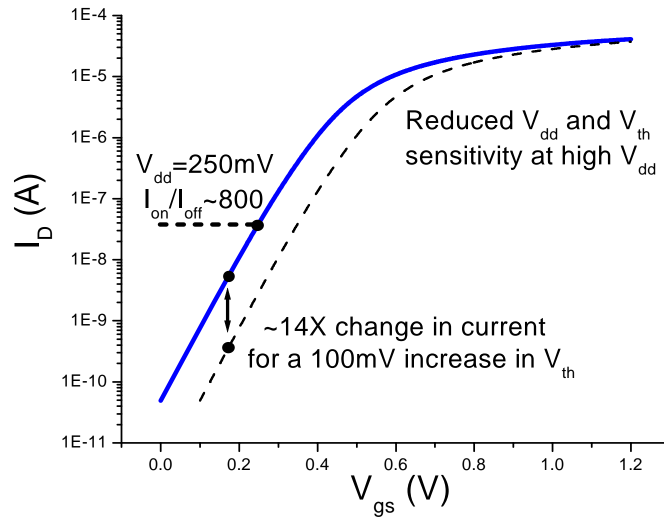


Figure 2.1: Simulated NFET drain current as a function of gate voltage in a 0.13µm technology

dependence on W, L_{eff}, and μ_{eff}. These quantities, which are very important in superthreshold device-level and gate-level design, are somewhat less important in subthreshold circuits.

In addition to increased sensitivities, Figure 2.1 shows that the on-current to off-current ratio (I_{on}/I_{off}) reduces dramatically from ~800,000 at V_{dd}=1.2V to ~800 at V_{dd}=250mV. The value of I_{on}/I_{off}, which reduces exponentially with V_{dd} in the subthreshold regime, is strongly correlated with delay, suggesting that subthreshold circuits are exponentially slower than superthreshold circuits. Additionally, a reduction in I_{on}/I_{off} tends to reduce noise margins, which is particularly problematic for SRAM.

2.2 Selecting V_{dd}

In [35], it was shown that CMOS gates composed of ideal transistors with a subthreshold swing of 60 mV/decade should function properly with a supply voltage as low as 36mV. Despite a non-ideal subthreshold swing, measurements of an inverter

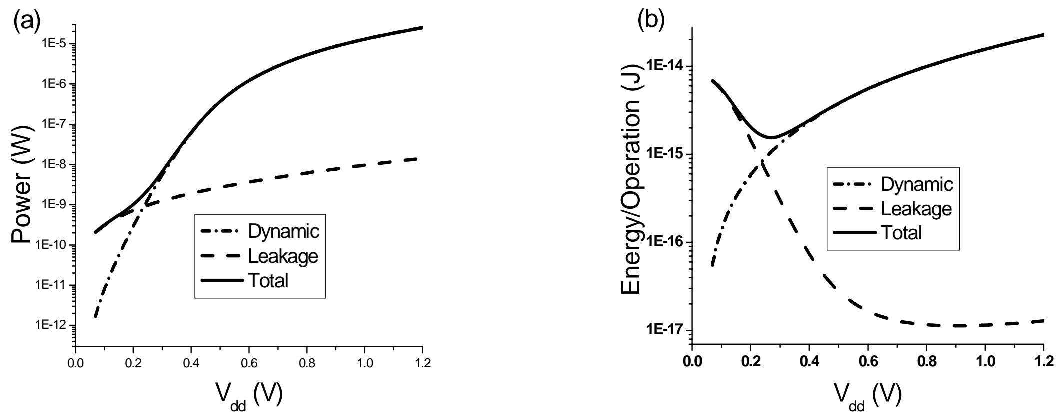


Figure 2.2: (a) Power consumed by an inverter chain as a function of supply voltage (V_{dd}) (b) Energy consumed per switching operation by the same inverter chain as a function of V_{dd}.

show that functionality can be achieved with a supply voltage of just 65mV [17]. It is clear that CMOS logic functions at extremely low voltages, but we must still consider the question of whether operation at these voltages is worthwhile. Figure 2.2(a) shows how the power consumed by an inverter chain scales with supply voltage. The total power consumption is broken into dynamic power (the power consumed by switching gates) and leakage power (the power consumed by idle gates). Minimum power is achieved by choosing the minimum functional supply voltage.

However, power is not always the most appropriate metric. For many applications, especially those in which battery life is the primary concern, energy per instruction may be a more sensible metric. There is a subtle but important difference between energy and power that is highlighted in Figure 2.2(b), which shows the energy consumed per switching event (which we call an operation) for the inverter chain from Figure 2.2(a).

Although Figure 2.2(a) shows that minimizing supply voltage will minimize power, the energy inflection point in Figure 2.2(b) shows that minimum energy is achieved at

some voltage that is greater than the minimum functional supply voltage. This energy minimum is due to a rapid increase in gate delay as the supply voltage scales below the threshold voltage. As gate delay increases, the amount of time that each gate spends leaking also increases. As a result, the total leakage energy (the product of leakage current, supply voltage, and total leakage time) increases quickly and creates the minimum apparent in Figure 2.2(b). The location of this minimum energy supply voltage (V_{min}) is a strong function of both switching activity and logic depth (the number of gates between an input and an output) and was derived in [23] as:

$$V_{\min} = \left[1.587 \cdot \ln\left(\eta \cdot \frac{n}{\alpha}\right) - 2.355\right] \cdot m \cdot v_T \qquad \text{Equation 2.2}$$

where α is the switching activity, n is the logic depth, m is the subthreshold slope factor, v_T is the thermal voltage, and η is a delay-related technology parameter.

2.3 Fighting Variability

The last sub-section showed that energy/operation may be, in theory, minimized by operating in the subthreshold regime. In practice, achieving energy optimality is not as simple as reducing V_{dd} to V_{min}. Process-induced variability leads to problems with both functionality and energy efficiency.

We can take a simplistic but accurate view of variability by assuming that there are four types of variation: systematic (global) V_{th} variation, random V_{th} variation, systematic (global) gate length variation, and random gate length variation. There is also some component of threshold and gate length variation that varies from region to region

on the chip, but this can be safely grouped with systematic (global) variation for this simple discussion.

Figure 2.3 shows how the delay variation of a chain of 10 inverter changes as V_{dd} scales in a 65nm technology. Since subthreshold current is exponentially dependent on V_{th}, variation in V_{th} becomes more problematic at subthreshold voltages. Conversely, subthreshold current is inversely proportional to gate length and has a relatively weak exponential dependence through DIBL-induced V_{th} variations. Threshold variations are therefore the most important concern for subthreshold designers.

The increased sensitivity to threshold voltage fluctuations in the subthreshold regime leads to dramatic variations in gate delay, which may result in both late-mode (i.e., setup time violations) and early-mode timing failures (i.e., hold time violations). Late-mode failures occur when a circuit path delay exceeds the clock period and may be fixed by increasing the clock period. Monte Carlo simulations show that the clock period for a 10-inverter chain in a 65nm technology must increase by 10% at V_{dd}=1V and an astonishing 230% at V_{dd}=300mV to eliminate late-mode errors introduced by variability. The performance and energy implications of addressing late-mode failures are clearly undesirable. Early-mode failures can occur when excessive clock skew allows data to be latched on clock cycle early at a receiving latch. Monte Carlo simulations suggest that clock skew (in terms of FO4 delay) can increase by more than 10X as voltage is scaled from 1V to 300mV. Early-mode failures must be fixed by adding delay elements to short paths or by designing variation-tolerant clock distribution networks.

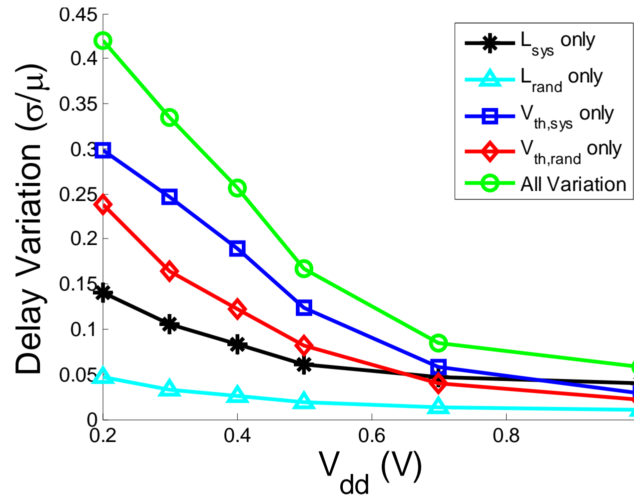


Figure 2.3.: Delay variability (σ/μ) as a function of supply voltage (65 nm technology)

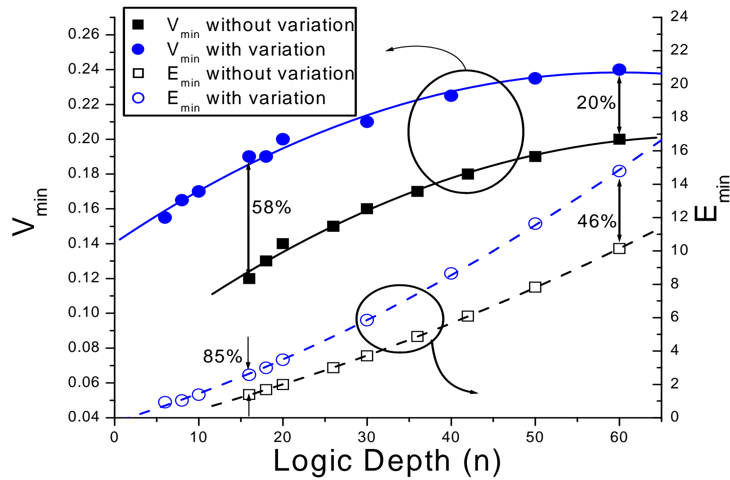


Figure 2.4: Variation in worst-case (μ+3σ) V_{min} and E_{min} for an inverter chain of length n gates. The relative increases in V_{min} and E_{min} are less severe at large n (0.13μm technology).

For most subthreshold designs, energy will be the most important metric. It is therefore very important to understand how V_{min}, identified in the previous sub-section, is affected by variability. While dynamic energy remains relatively constant with variability, worst-case delay and worst-case leakage energy increase dramatically [18]. Assuming a fixed frequency and supply voltage across all chips, process variations leads to a lower operating frequency and consequently, a dramatic increase in leakage. Figure 2.4 shows the variation in the worst-case V_{min} and energy for an inverter chain of length n in a 0.13µm technology.

Though variability is one of the primary problems in subthreshold regime, careful design will help alleviate its effects. Systematic variations can be addressed globally with techniques like dynamic voltage scaling (DVS) and adaptive body biasing (ABB), though both techniques incur area and complexity overheads. ABB, in particular, will be explored in Chapter 4 within the context of an 8-bit subthreshold processor.

Random variations are somewhat more difficult to address in a methodical fashion. The "averaging" of variability through longer path lengths and larger gate sizes is one of the few tools available to the designer. Consider a timing path composed of identical gates affected only by random delay variability (i.e., gates are identically and independently distributed with mean of μ_o and standard deviation of σ_o). The Central Limit Theorem predicts that the delay of a path with n gates will approach a distribution with mean of $\mu_n = n \cdot \mu_o$ and standard deviation of $\sigma_n = \sqrt{n} \cdot \sigma_o$. While the *absolute* variation (i.e., σ_n) of this delay distribution increases with n, the *relative* variation of delay (i.e., σ_n/μ_n) reduces by a factor of $\sqrt{n}$ with increasing n. Figure 2.4 confirms this behavior and shows that relative energy (E_{min}) and V_{min} variations reduce dramatically

with increasing path lengths. Designers will find the reduced relative variability at longer path lengths attractive, but this behavior may be problematic for clock distribution, where an increase in absolute variability results in increased clock skew and a greater incidence of timing errors. Recent research has also shown that timing and energy fluctuations due to random variation can be reduced by using larger gate sizes [18][36]. The Central Limit Theorem is again relevant, showing that variation reduces by a factor of $\sqrt{W \cdot L}$ with increasing gate area. Note in this case that the reduced variability comes at the price of increased energy, a factor that must be considered carefully by designers.

Successful subthreshold operation will undoubtedly require the development of error-tolerant architectures. Error correction codes (ECC) and memory redundancy have been studied extensively [37], but it will be important to create whole architectures than can dynamically detect and correct errors [88]. Without the development of such techniques, designers will be forced to incorporate large margins in design parameters to avoid timing errors and will need to resort to expensive statistical techniques like Monte Carlo simulation.

Chapter 3

Device Scaling in Low Voltage Circuits

To this point we have explored the energy benefits of low voltage operation and have examined variability, the most challenging problem for subthreshold designers. Most of these explorations have been carried out in a 0.13μm process. Device scaling has been one of the most important developments for performance, energy, and cost in super-V_{th} circuits [38], so it is prudent to investigate how scaling will affect subthreshold operation. In this chapter, the implications of device scaling are considered and simple modifications to the standard CMOS processing flow are suggested to improve noise margins, energy, and delay in scaled subthreshold circuits.

We first use realistic two-dimensional device models (in MEDICI) scaled from the 90nm technology node down to the 32nm technology node to quantify the device-level and gate-level implications of conventional super-V_{th} device scaling. We show that the slow scaling of gate oxide relative to the channel length leads to a 60% reduction in I_{on}/I_{off} between the 90nm and 32nm nodes, which results in static noise margin (SNM) degradation of more than 10% between the 90nm and 32nm nodes in a CMOS inverter. We propose a modified scaling strategy that uses increased channel lengths and reduced doping to improve inverse subthreshold slope. We develop new delay and energy metrics that effectively capture the important effects of device scaling, and we use those to drive

device optimization. We find that noise margins improve by 19% and energy improves by 23% in 32nm sub-V_{th} circuits when applying the modified device scaling strategy. The proposed approach is particularly attractive since it requires only simple modifications to existing device technologies. Following this initial analysis, we look at the problem of scaled sub-V_{th} SRAM, which will likely be the sub-V_{th} circuit most sensitive to device scaling. We use both nominal and corner based analysis to measure noise margins in memories targeted at sensor applications, where memory sizes on the order of several kilobits are sufficient. We find that our optimized sub-V_{th} device has a nominal read SNM that is 64% larger than that of the unoptimized super-V_{th} device at the 32nm generation.

The remainder of this chapter is organized as follows. In Section 3.1, the implications of performance-driven scaling in the sub-V_{th} regime are described. In Section 3.2, an alternative scaling strategy driven by the needs of sub-V_{th} circuits is proposed and compared to a super-V_{th} scaling strategy. Finally, in Section 3.3, scaled sub-V_{th} SRAM is studied in detail.

3.1 Modeling Super-V_{th} Scaling

In this section, we describe a simple but accurate bulk transistor model, illustrated in Figure 3.1(a), which captures the important effects of conventional super-V_{th} scaling. The text and figures will focus on the NFET device for the remainder of this paper, but an analogous methodology is also used to describe the PFET device. The device model has four key scaling parameters: physical gate length (L_{poly}), gate oxide thickness (T_{ox}), substrate doping (N_{sub}), and peak halo doping ($N_{p,halo}$). These parameters receive special

attention because they are most important when determining key device characteristics like V_{th}, on-current, off-current, and gate capacitance. In addition to these four parameters, we specify V_{dd} as an additional knob for adjusting performance. All physical dimensions other than T_{ox} (source/drain junction depth, lateral source/drain diffusion, halo dimensions, etc.) scale in proportion to L_{poly}.

Note that halo doping regions are located near the source and drain edges. Halo doping is used to control V_{th} roll-off observed at short channels and large drain biases, and has become indispensable for super-V_{th} devices. The V_{th} of a short channel device with halo doping may be represented as the sum of three components: intrinsic (long channel) threshold voltage (V_{th0}), roll-off due to short channel effects and DIBL ($\Delta V_{th,SCE}$), and roll-up due to halo doping ($\Delta V_{th,halo}$) [39]. In a well optimized device, the halo regions increase the effective channel doping at short channel lengths such that $-\Delta V_{th,SCE}=\Delta V_{th,halo}$, and V_{th} remains flat as a function of both L_{poly} and V_{ds}. The halo regions are modeled as a pair of two dimensional Gaussian distributions superimposed on a uniformly doped substrate similar to [40][41]. The doping contours of a representative 90nm device are shown for illustrative purposes in Figure 3.1(b). The net halo doping, N_{halo}, is the sum of N_{sub} and $N_{p,halo}$.

For our purposes, describing a device at a particular technology node only requires that the four key parameters and V_{dd} are specified. The iterative process described in [77] is used to optimize device parameters at a given technology node. L_{poly} and T_{ox} are first determined based upon published industry data. V_{dd} and V_{th} (through N_{sub} and $N_{p,halo}$) are then chosen to optimize delay under leakage constraints. The selection of each parameter is described in the remainder of this section.

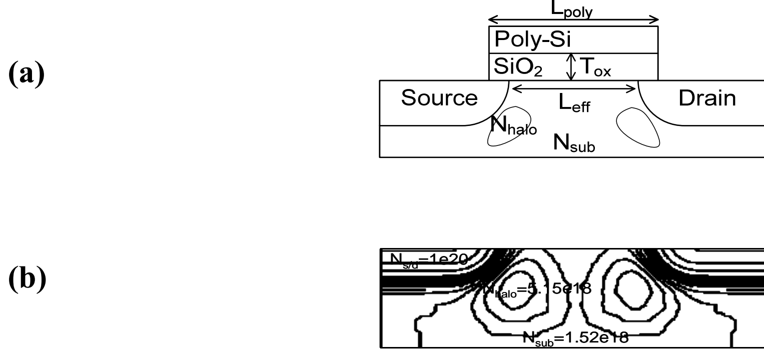


Figure 3.1: (a) A device cross-section showing scaling parameters (b) Doping profile for a 90nm NFET

The aggressive scaling of L_{poly} has been one of the primary drivers of performance improvement in MOSFETs. Note that L_{poly} represents the length of the bottom of the poly-Si gate after etching. For example, a gate with a designed length of 90nm might have L_{poly}=65nm after etching. Throughout this section, the minimum L_{poly} is assumed to reduce by 30% per generation, which agrees well with recent L_{poly} scaling trends.

Selecting a realistic value for T_{ox} plays a critical role in determining the sub-V_{th} characteristics of a device. A survey of recent industrial publications in [42] shows that T_{ox} has been reduced by ~10% per generation below the 130nm technology node, which is slower than other dimensions. In this paper, it is assumed that T_{ox} reduces by 10% per generation. Note that the oxide scaling problem may be even worse than the assumption of 10%. High-κ dielectrics may be the only solution since conventional gate stacks may be limited to a minimum of ~1nm thickness [43].

With L_{poly} and T_{ox} fixed for each generation, the remaining three parameters (N_{sub}, $N_{p,halo}$, V_{dd}) may be tuned to match delay and leakage requirements. As in [77], the optimization in this work uses delay (τ) as an objective and leakage ($I_{leak,max}$) as a

Node	90nm	65nm	45nm	32nm
L_{poly} (nm)	65	46	32	22
T_{ox} (nm)	2.10	1.89	1.70	1.53
N_{sub} (cm^{-3})	1.52e18	1.97e18	2.52e18	3.31e18
N_{halo} (cm^{-3})	3.63e18	5.17e18	7.83e18	12.0e18
$N_{ch,avg}$ (cm^{-3})	2.82e18	3.84e18	5.27e18	7.38e18
V_{dd}	1.2	1.1	1.0	0.9
$V_{th,sat}$ (mV)	403	420	438	461
I_{off} (pA/µm)	100	125	156	195
C_gV_{dd}/I_{on} (ps)	1.3	0.97	0.75	0.62

Table 3.1: NFET parameters under super-V_{th} scaling

constraint. Note that N_{sub} is treated as a function of the long channel device (where halo doping is largely unnecessary), and $N_{p,halo}$ is treated as a function of the short channel device. While the approach described in [77] may not converge on the optimal solution, it is a systematic, simple heuristic that produces realistic scaled devices.

The selection of $I_{leak,max}$ is a complex topic since every new technology provides a range of devices optimized for different power-delay points. For example, the 65nm technology described in [44] offers low power and high power devices, with each device having 3 different V_{th} variants. The International Technology Roadmap for Semiconductors (ITRS) [45], which maps out near-term and long-term goals for the semiconductor industry, describes three different devices with different power-delay trade-offs: high performance, low operating power (LOP), and low standby power (LSTP). The LOP and LSTP devices are optimized in a similar manner, though the LSTP device has more stringent leakage constraints. In this paper, a super-V_{th} scaling strategy similar to that of the LSTP device is used. The ITRS predictions rely on the introduction

of advanced technologies like high-κ gate stacks to meet stringent leakage constraints. Since we are studying the effects of current scaling trends (rather than projected scaling goals that require the introduction of advanced technologies), the leakage constraints may be relaxed slightly. A maximum leakage current of 100pA/µm is selected at the 90nm node and leakage is allowed to grow by 25% each generation. The supply voltage is reduced regularly at each generation to control dynamic energy, and the device is optimized for minimum delay under the leakage constraint. Table 3.1 shows values for the NFET model parameters generated for the 90nm through 32nm nodes using the scaling approach described in this section. Throughout this paper, the results in Table 3.1 are referred to as the "super-V_{th} scaling strategy."

The intrinsic delay of a device may be quantified as $\tau = C_g V_{dd} / I_{on}$ where C_g is the gate capacitance including gate/drain-source overlap and I_{on} is the drain current at $V_{gs} = V_{ds} = V_{dd}$. This metric, which has been shown to correlate well with CMOS gate delay [46], is shown for reference in Table 3.1.

3.2 Implications of Super-V_{th} Scaling

The device models from the previous section have been simulated in MEDICI, a two-dimensional device simulator. All simulations were conducted at a temperature of 300ºK. Temperature fluctuations are not considered in this work, though they do merit study since the expected lower operating temperature of sub-Vth circuits is tempered by an exponential temperature sensitivity.

We begin with a focus on device characteristics and then look at gate-level characteristics. The current in a sub-V_{th} circuit may be described by the well-known weak

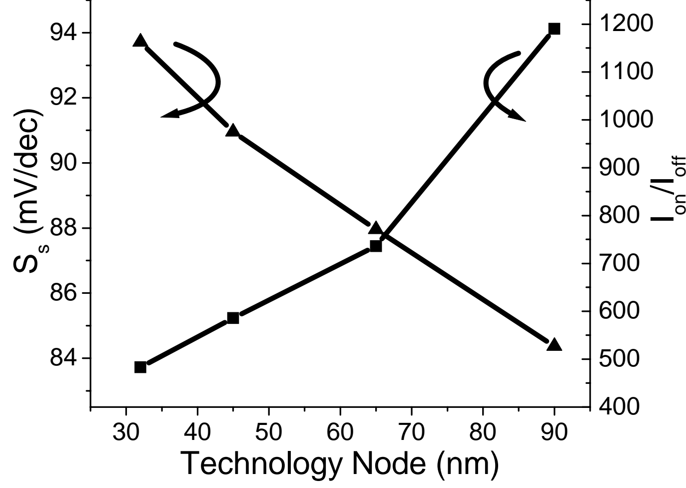


Figure 3.2: NFET inverse sub-V_{th} slope and on-current to off-current ratio

inversion current expression shown in Eq. 3.1 [47], where m is the subthreshold slope factor and C_{dep} is the depletion capacitance. Note the exponential dependence on m and V_{th}.

$$I_{sub} = \frac{W}{L_{eff}} \cdot \mu_{eff} \cdot C_d \cdot v_T^2 \cdot e^{\left(\frac{V_{gs}-V_{th}}{m \cdot v_T}\right)} \cdot \left(1 - e^{-\frac{V_{ds}}{v_T}}\right) \qquad (3.1)$$

The inverse subthreshold slope (S_S), an excellent measure of channel control, may be expressed for short channel MOSFETs as [47]:

$$S_S = 2.3 \cdot v_T \cdot m \qquad (3.2a)$$

$$S_S = 2.3 \cdot v_T \cdot \left(1 + \frac{3 \cdot T_{ox}}{W_{dep}}\right)\left(1 + \frac{11 T_{ox}}{W_{dep}} e^{-\frac{\pi \cdot L_{eff}}{2\left(W_{dep}+3T_{ox}\right)}}\right) \qquad (3.2b)$$

where $W_{dep} \propto 1/\sqrt{N_{eff}}$ is the depletion width with effective channel doping, N_{eff}. The value of S_S which is theoretically limited to values larger than ~60mV/dec at T=300K, should be as small as possible to ensure the steepest sub-V_{th} characteristic. As shown in Eq. 3.2(b), the final exponential term forces S_S to increase as L_{poly} (and consequently L_{eff}) reduces

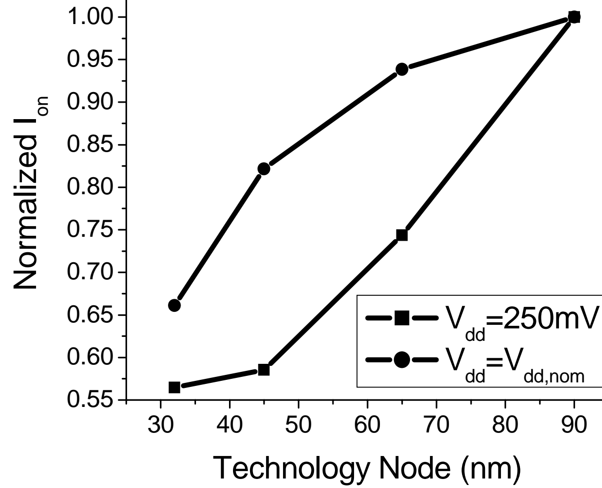


Figure 3.3: NFET on-current

relative to T_{ox} and W_{dep}. Figure 3.2 shows the simulated S_S for an NFET device at different technology nodes. Between the 90nm and 32nm nodes, S_S degrades by 11%, which corresponds to a 60% reduction in the on-current to off-current ratio (I_{on}/I_{off}) at V_{dd}=250mV. I_{on} is measured at $V_{gs}=V_{ds}=V_{dd}$. Note in Table 3.1 that all devices have V_{th}>400mV, so V_{dd}=250mV is well within the sub-V_{th} regime. We will show later in this section that the dramatic reduction in I_{on}/I_{off} leads to serious problems for noise margins and energy efficiency.

Figure 3.3 highlights the behavior of I_{on} at both nominal V_{dd} (with values taken from Table 3.1) and V_{dd}=250mV. Under our leakage constrained scaling scenario, I_{on} reduces between technology generations in the super-V_{th} region. Note that the choice of leakage constraint (100pA plus 25% per generation) affects this outcome. A more aggressive technology, especially one leveraging strain in the channel, would likely achieve increased drain current with scaling. However, in this study, we are concerned with low power devices. Note that the reduction in current is more dramatic for the

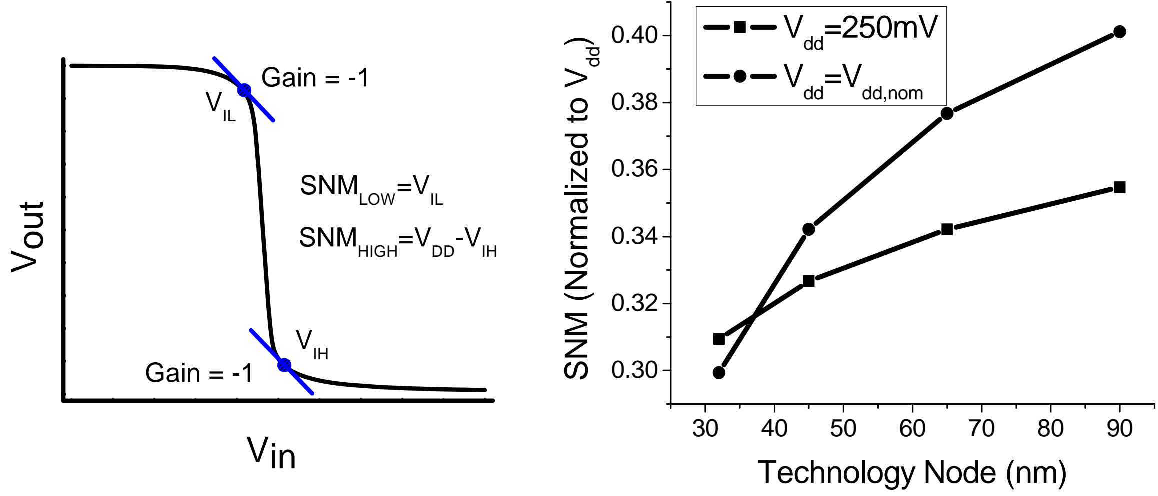


Figure 3.4: (a) Definition of SNM (b) Simulated SNM for a scaled inverter

device measured in the sub-V_{th} region. This loss of drain current has important delay implications that will be discussed later in this section.

Consider the static noise margins (SNM) of a CMOS inverter. The voltage transfer characteristic of a sub-V_{th} inverter is computed by equating drain current (Eq. 3.1) through NFET and PFET devices, as shown in Eq. 3.3(a). $I_{o,N}$ and $I_{o,P}$ are the NFET and PFET currents at $V_{gs}=V_{th}$ with $V_{ds}>>v_T$. V_{in} and V_{out} are the voltages at the input and output of the inverter. We can relate V_{in} and V_{out} using Eq. 3.3(b). We can further simplify the expression by assuming $I_{o,N}=I_{o,P}$, $V_{th,N}=V_{th,P}=V_{th}$ and $m_N=m_P=m$ (Eq. 3.3(c)).

$$I_{o,N} \cdot e^{\frac{V_{in}-V_{th}}{m \cdot v_T}} \left(1-e^{-\frac{V_{out}}{v_T}}\right) = I_{o,P} \cdot e^{\frac{V_{dd}-V_{in}-V_{th}}{m \cdot v_T}} \left(1-e^{-\frac{V_{dd}-V_{out}}{v_T}}\right) \quad (3.3a)$$

$$V_{in} = \frac{m_n\left(V_{dd}-V_{th,p}\right)+m_p V_{th,n}+m_n m_p v_T \ln\left(\frac{I_{o,P}}{I_{o,N}} \cdot \frac{1-e^{-\frac{V_{dd}-V_{out}}{v_T}}}{1-e^{-\frac{V_{out}}{v_T}}}\right)}{m_n+m_p} \quad (3.3b)$$

$$V_{in} = \frac{V_{dd}}{2} + \frac{m \cdot v_T}{2} \ln\left(\frac{1 - e^{-\frac{V_{dd} - V_{out}}{v_T}}}{1 - e^{-\frac{V_{out}}{v_T}}} \right) \quad (3.3c)$$

The important role of S_S (through m) in determining the voltage transfer characteristic (and consequently SNM) is obvious, particularly in Eq. 3.3(c). In simulation, we define SNM at the points where the gain in the voltage transfer characteristic equals negative one, as shown in Figure 3.4(a). We present SNM as the mean of SNM_{LOW} and SNM_{HIGH}. Figure 3.4(b) shows the simulated value of SNM for a CMOS inverter simulated at nominal V_{dd} (Table 3.1) and V_{dd}=250mV. The increase in S_S with scaling results in SNM degradation of more than 10% between the 90nm and 32nm nodes. This is a serious concern for sub-V_{th} designers since absolute noise margins are already dramatically reduced compared to high voltage operation. It is particularly concerning for SRAM, where noise margins are paramount and a small I_{on}/I_{off} in sub-V_{th} circuits already places tight limits on the maximum number of bits/line [72].

The delay of a CMOS gate may be expressed as:

$$t_p = \frac{k_d \cdot C_L \cdot V_{dd}}{I_{on}} \quad (3.4)$$

where C_L is the load capacitance and k_d is a fitting parameter. Note that C_L includes both gate capacitance (which is dependent on device dimensions) and junction capacitance (which is dependent on device dimensions and doping), both of which are captured by the simple device model presented in this work. The sub-V_{th} delay may be found by substituting Eq. 3.1 into Eq. 3.4:

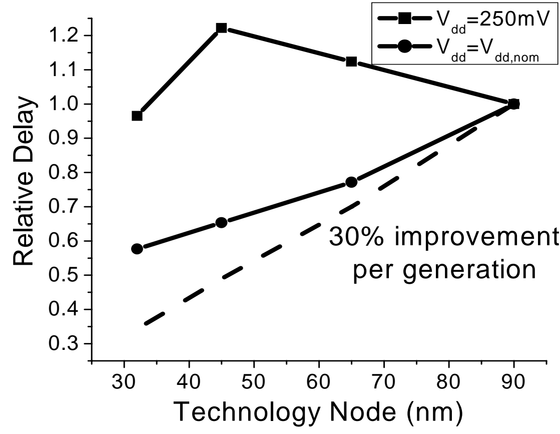


Figure 3.5: Simulated delay for a scaled inverter

$$t_p = \frac{k_d \cdot C_L \cdot V_{dd}}{I_{on}} = \frac{k_d \cdot C_L \cdot V_{dd}}{I_{o,N} \cdot e^{\frac{V_{dd}-V_{th}}{m \cdot v_T}}} \qquad (3.5)$$

The V_{ds} dependence of I_{on} (shown in Eq. 3.1) has been ignored since it is negligible for $V_{gs}=V_{dd} >> v_T$. The delay expression is clearly dominated by an exponential dependence on V_{dd}, V_{th}, and m.

The simulated delay of a CMOS inverter with FO1 loading is shown in Figure 3.5 at nominal V_{dd} (Table 3.1) and at 250mV. As expected, the delay at nominal V_{dd} improves with L_{poly}, though at a rate that is slower than the target of 30% per generation under generalized scaling (assuming $1/\alpha=0.7$). In contrast, the delay increases between the 90nm and 45nm nodes at V_{dd}=250mV. Due to the relaxed I_{off} constraint imposed at advanced technology nodes (a 25% increase is allowed per generation), one might expect that delay would decrease with scaling (since I_{on} increases with I_{off}). However, S_S degrades over the same region, dramatically reducing I_{on} and increasing delay. Between the 45nm and 32nm nodes, the increase in I_{off} begins to dominate any degradation in S_S and causes a reduction in delay. A more stringent leakage constraint during device

optimization would yield a monotonic delay increase. The important lesson is that sub-V_{th} delay is exponentially sensitive to V_{th} and S_S and only linearly sensitive to L_{poly}. Even small changes to a super-V_{th} device to control leakage and short channel effects may result in large fluctuations in sub-V_{th} delay. It is likely that V_{th} and S_S scaling, not L_{poly} scaling, will control the performance of future sub-V_{th} circuits. Strict attention to V_{th} selection and S_S control will be an important part of any technology optimized for sub-V_{th} use.

In sub-V_{th} applications, V_{dd} is typically set at the energy optimal value, V_{min}, so the scaling of delay at $V_{dd}=V_{min}$ is of interest. The value of V_{min} was found in [23][24] to be proportional to S_S. If we ignore the dependence of V_{min} on the slope of the input waveform, then we can set $V_{dd}=V_{min}=K_{Vmin}\cdot S_S$ where K_{Vmin} is a parameter that depends only on the structure of the circuit (and not on scaling parameters) [23]. Using this new relation and by recognizing that $S_S=V_{dd}/log(I_{on}/I_{off})$, we can express Eq. 3.4 and Eq. 3.5 in terms of only scaling dependent parameters (Eq. 3.6). The simple expression in Eq. 3.6 suggests that we can predict the scaling behavior of sub-V_{th} delay simply by understanding the scaling of C_L, S_S, and I_{off}. We develop a similar expression for energy in the next sub-section.

$$t_p = \frac{k_d \cdot C_L \cdot K_{V\min} \cdot S_S}{I_{off} \cdot 10^{\frac{K_{V\min} S_S}{S_S}}} \propto \frac{C_L \cdot S_S}{I_{off}} \quad (3.6)$$

The energy of a single inverter driving an identical inverter can nominally be separated into two components: dynamic (E_{dyn}) and leakage (E_{leak}).

$$E_{dyn} = C_L \cdot V_{dd}^2 \cdot \alpha \quad (3.7a)$$

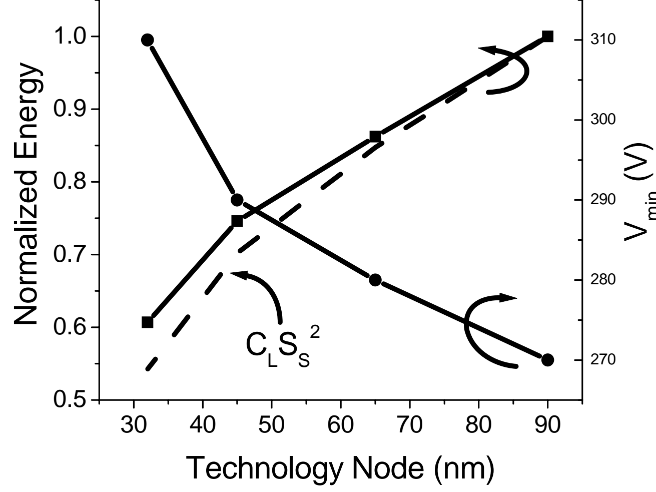


Figure 3.6: Simulated energy/cycle and V_{min} for a chain of 30 inverters with α=0.1

$$E_{leak} = I_{off} \cdot V_{dd} \cdot t_p = I_{off} \cdot V_{dd} \cdot \frac{k_d \cdot C_L \cdot V_{dd}}{I_{on}} = C_L \cdot V_{dd}^2 \cdot k_d \cdot \frac{I_{off}}{I_{on}} \qquad (3.7b)$$

The term α is the activity factor and all other terms are previously defined. If we again assume that operation only occurs at the energy optimal V_{dd}=V_{min}, then we can simplify Eq. 3.7(a) and Eq. 3.7(b) as follows:

$$E_{dyn} = C_L \cdot \left(K_{V\min} \cdot S_S\right)^2 \cdot \alpha \propto C_L \cdot S_S^2 \qquad (3.8a)$$

$$E_{leak} = C_L \cdot \left(K_{V\min} \cdot S_S\right)^2 \cdot k_d \cdot 10^{-K_{V\min}} \propto C_L \cdot S_S^2 \qquad (3.8b)$$

The only parameters that change as a result of device scaling are C_L and S_S. Equation 3.8 suggests the interesting result that dynamic energy and leakage energy in sub-V_{th} circuits have an identical dependence on scaling parameters and that the ratio E_{dyn}/E_{leak} is insensitive to scaling when operating at V_{dd}=V_{min}.

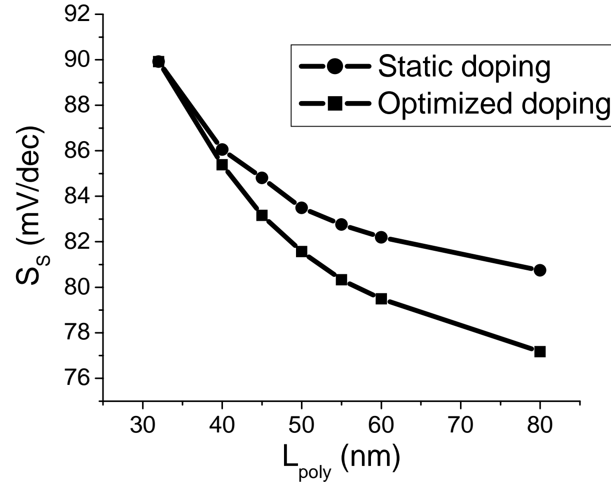


Figure 3.7: S_S as a function of gate length for a 45nm device

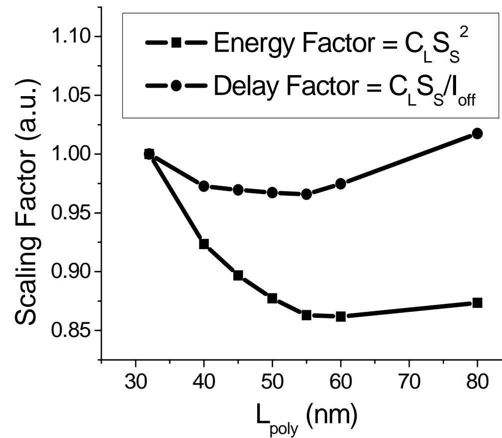


Figure 3.8: Energy and delay factors for a 45nm device

The simulated energy consumed per cycle by a chain of 30 inverters with α=0.1 and V_{dd}=V_{min} is plotted in Figure 3.6. There is a substantial energy reduction as devices are scaled from the 90nm to the 32nm node. However, note that V_{min} increases by 40mV for this simple circuit between the 90nm and 32nm nodes. Recall that V_{min} is proportional to S_S, so this trend is not surprising. It was shown in [17] that an increase in V_{min} is generally not beneficial for energy efficiency. An increase in V_{min} essentially equates to a dynamic energy ($C_L V_{dd}^2$) penalty. Ideally, a scaled sub-V_{th} device should experience a

reduction in capacitance while maintaining V_{min}. The factor $C_L{\cdot}S_S^2$, which is also plotted in Figure 3.6, matches very closely to the energy measurements, thus confirming the validity of Eq. 3.8.

3.3 Redirecting Scaling for Sub-V_{th} Circuits

It became clear in the last sub-section that the degradation of S_S with device scaling will be problematic for robust, energy efficient sub-V_{th} operation. Moreover, the scaling of L_{poly} to improve the delay characteristics of super-V_{th} devices is not relevant in sub-V_{th} circuits since delay is largely controlled by V_{th} and S_S. Ideally, we would like a sub-V_{th} transistor with a very small S_S to address noise margin and energy concerns. This device should be available in multiple well controlled thresholds in order to provide a wide range of performance points. In this section, the design of such a device is explored.

The degradation of S_S with scaling is driven by two related factors. The first factor has already been made clear: the ratio L_{eff}/T_{ox} reduces with each technology generation due to the slow scaling of T_{oz} and worsens the V_{th} roll-off problem. This suggests that longer channel lengths should be used to accommodate the gate oxide. The second factor causing S_S degradation, which was also covered in [41], is more subtle. To compensate for the V_{th} roll-off problem, the channel doping is effectively increased through aggressive use of halo doping. For long-channel devices, the halo doping is less critical and actually degrades S_S. Therefore, to fully optimize S_S with device scaling, it is not sufficient to simply lengthen L_{poly} without considering the doping. Instead, L_{poly} and doping must be optimized simultaneously. This notion is confirmed in Figure 3.7, which

Node	90nm	65nm	45nm	32nm
L_{poly} (nm)	95	75	60	45
T_{ox} (nm)	2.10	1.89	1.70	1.53
N_{sub} (cm^{-3})	1.61e18	1.99e18	2.53e18	3.19e18
N_{halo} (cm^{-3})	2.02e18	2.73e18	2.93e18	4.89e18
$N_{ch,avg}$ (cm^{-3})	2.01e18	2.45e18	2.93e18	3.55e18
$C_L \cdot S_S^2$ (a.u.)	1	0.80	0.65	0.51
$C_L \cdot S_S$ (a.u.)	1	0.80	0.65	0.50

Table 3.2: NFET parameters under sub-V_{th} scaling

shows S_S for a 45nm device with a fixed doping profile and for a 45nm device with a doping profile optimized for each value of L_{poly}.

Increasing L_{poly} and reducing doping improves S_S at the cost of increased gate capacitance. The cost of this optimization can be quantified in terms of energy and delay. Equation 3.6 shows that sub-V_{th} delay is proportional to $C_L \cdot S_S / I_{off}$ at $V_{dd}=V_{min}$. Similarly, Eq. 3.8(a) and Eq. 3.8(b) show that energy in a sub-V_{th} circuit is proportional to $C_L \cdot S_S^2$. These expressions are useful since they are simple functions of device parameters and offer a quick estimation of energy and delay in a prospective technology. Figure 3.8 plots these energy and delay factors as functions of L_{poly} for the optimized 45nm device originally highlighted in Figure 3.7. Both reach a minimum, suggesting that there is both a delay-optimal and energy-optimal L_{poly}. Both minima occur at similar values of L_{poly}. At the delay-optimal value of L_{poly} (55nm), energy is only 0.1% greater than its minimum value. At the energy-optimal value of L_{poly} (60nm), delay is only 0.9% greater than its minimum value. For simplicity, we select the energy minimal L_{poly} for a negligible delay penalty. Note that delay typically degrades as ~1/L_{poly}, but we are able to avoid this problem by also optimizing the doping.

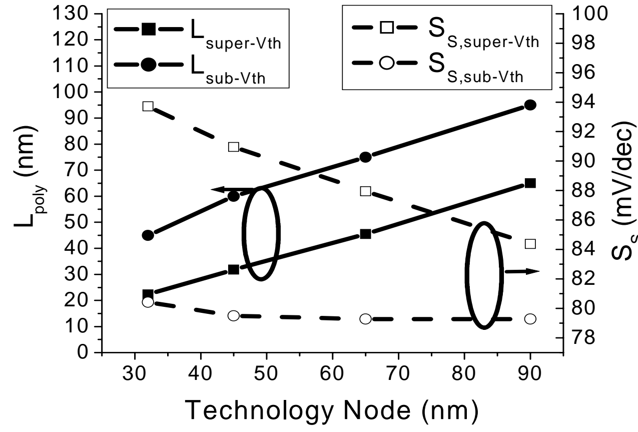


Figure 3.9: NFET L_{poly} and S_S for sub-V_{th} and super-V_{th} scaling strategies

Given the important role that S_S plays in determining energy efficiency, performance, and noise margins, we propose a scaling strategy that reduces S_S by targeting the energy optimal L_{poly} at each technology node. The proposed strategy uses longer channel lengths that scale more slowly than the rate of 30% assumed in Section 2. As we will see, one consequence of this strategy is that S_S remains approximately constant with device scaling. For this study, a constant I_{off} of 100pA/µm is maintained across all device generations. Fixing I_{off} yields a more predictable delay scaling characteristic and avoids the problems illustrated in Figure 3.5. Just as in super-V_{th} technologies, different performance levels can be targeted by offering multiple thresholds.

Using a baseline 90nm device identical to the super-V_{th} 90nm device, L_{poly} and doping have been optimized for minimum energy using Eq. 3.8(a) and Eq. 3.8(b). It is again assumed that T_{ox} reduces by 10% and all other physical dimensions, excluding L_{poly}, reduce by 30% each generation. The optimal L_{poly}, N_{sub}, and $N_{p,halo}$ at each generation are found as described in this section. The resulting NFET device parameters are listed in

Table 3.2. Energy (Eq. 3.8) and delay (Eq. 3.6) factors are also listed in Table 3.2. Note that the delay factor simplifies to $C_L \cdot S_S$ since I_{off} is constant with scaling. A similar set of values is derived for PFET devices. The energy optimal L_{poly} for the PFET device is almost identical to that of the NFET, so the L_{poly} values in Table 3.2 are used during PFET doping optimization. For the remainder of this section, the results in Table 3.2 will be called the "sub-V_{th} scaling strategy."

3.4 Evaluating the Modified Scaling Strategy

The primary purpose of the revised scaling strategy is to maintain strong channel control, even at very small dimensions. Figure 3.9 shows how L_{poly} and S_S scale under the proposed scaling strategy and under the original super-V_{th} scaling strategy. L_{poly} is larger than in the super-V_{th} scaling scheme and also scales at a slower rate (20-25% per generation) than the L_{poly} in the super-V_{th} scaling scheme (30%). Note that S_S stays very close to ~80mV/dec under our proposed strategy, varying by only 1.2mV/dec between the 90nm and 32nm nodes. As a result, SNM remains nearly constant as well (Figure 3.10). At the 32nm node, the optimized sub-V_{th} scaling strategy yields an SNM that is 19% larger than that observed under the super-V_{th} scaling strategy.

Normalized FO1 inverter delay is plotted in Figure 3.11 for both scaling scenarios. Delay reduces by ~18% per generation under the proposed strategy. Recall from the previous discussion that the delay characteristic for the super-V_{th} scaling strategy is not monotonic due to the scaling of V_{th} and I_{off}. It is therefore not fair to directly compare the delay scaling of the two strategies. However, it is clear that the sub-V_{th} scaling strategy

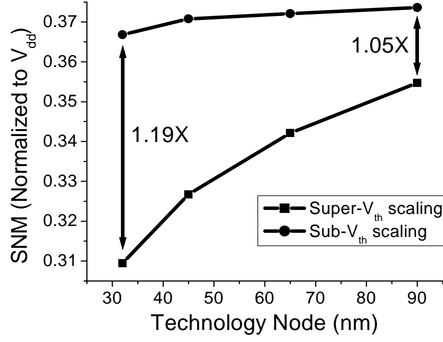


Figure 3.10: Simulated SNM for an inverter under super-V_{th} and sub-V_{th} scaling

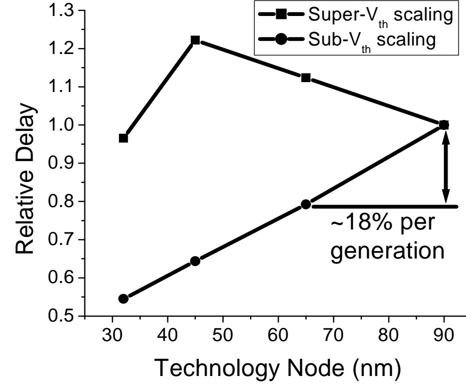


Figure 3.11: Simulated delay for an inverter at V_{dd}=250mV under super-V_{th} and sub-V_{th} scaling

exerts much tighter control over I_{off} and S_S than the super-V_{th} strategy so the delay characteristic scales much more gracefully.

Figure 3.12 shows the simulated energy and V_{min} for a chain of 30 inverters under the conventional super-V_{th} scaling scheme and the proposed scheme. The proposed strategy consumes ~23% less energy than the super-V_{th} scaling strategy at the 32nm node (measured at V_{min}), with V_{min} changing by only 10mV between the 130nm and 32nm nodes. The relatively low V_{min} (which previous work has shown to be a strong function of S_S and leakage energy [23][24]) is responsible for this energy reduction.

3.5 Stability of Scaled Sub-V_{th} SRAM

Robust memory design is the most challenging task facing low voltage designers. Recent work has demonstrated dramatic improvements in low voltage operation [19][31][32][33][71][72], but concerns about density and robustness in the face of variability still remain. In this section, we focus on memory design deep in the sub-V_{th} regime (in this case, V_{dd}=250mV). Such designs typically target sensor mote applications

where memory sizes on the order of several kilobits are sufficient. Though the minimum functional voltage of the traditional 6-transistor (6T) SRAM cell can be as high as 2/3 of the nominal supply voltage (0.8V in [48]), the 6T SRAM cell is used as a test vehicle in this work since it is the most widely adopted SRAM variant. Before looking at the scalability of sub-V_{th} SRAM, a simple variability model is described to supplement later observations.

Process-induced V_{th} variation makes sub-V_{th} SRAM design extremely challenging. Due to the exponential dependence of subthreshold current (Eq. 3.1) on V_{th}, even small variations lead to large strength mismatches between the pull-up (M1 in Figure 3.13c), pull-down (M2), and pass transistors (M3). Random V_{th} variability due to random dopant fluctuations (RDF) is largely responsible for within-cell mismatch. Past work has shown that RDF-induced variation is a formidable problem in super-V_{th} circuits [49], and it has also been shown that the importance of RDF grows relative to other random L_{poly} variation as V_{dd} reduces [18]. The V_{th} variations due to RDF may be modeled as [50]:

$$\sigma_{Vth,RDF} = 3.19e-8 \cdot \frac{T_{ox} N_A^{0.4}}{\sqrt{L \cdot W}} \qquad (3.9)$$

The inverse relationship to the square root of gate area suggests that random variation will worsen with device scaling. To model RDF in MEDICI, the expected number of dopants in the channel is first calculated by integrating the continuous doping profile in the box with corners at the source/drain top and bottom edges and multiplying by the width of the device. A Poisson distribution is then used to create perturbations in the dopant count [50]. To map the discrete dopant count back to a continuous distribution,

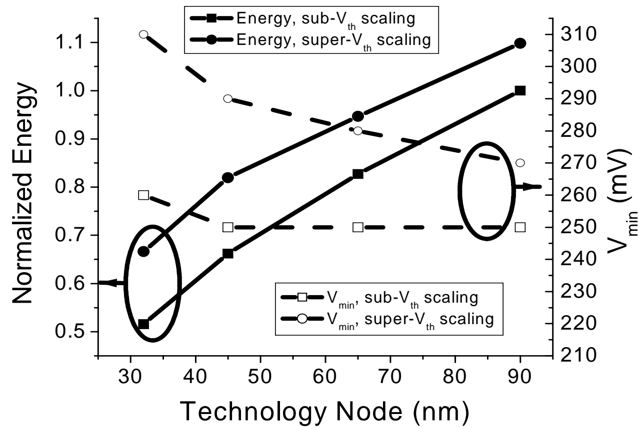
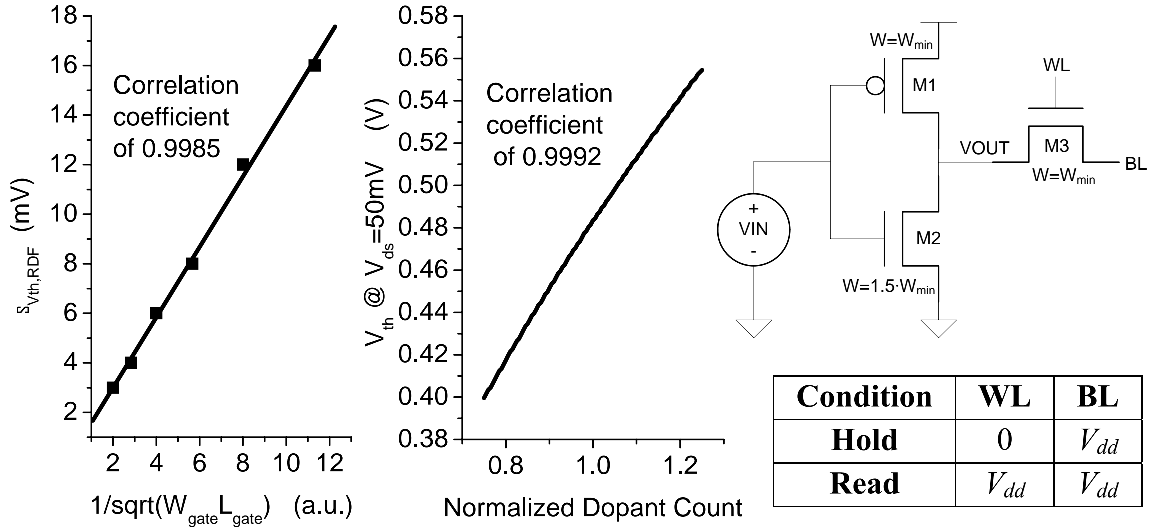


Figure 3.12: Simulated energy and V_{min} under super-V_{th} scaling and sub-V_{th} scaling

the entire doping profile is scaled by a constant. The model is shown in Figure 3.13(a) to agree well with the gate area dependence highlighted in Eq. 3.9.

Rather than run computationally intense Monte Carlo simulations, we can skew the SRAM cell to a worst case corner. The read becomes unstable when the pull-down device becomes weak (i.e. M2 has a high V_{th}) and the pass transistor becomes strong (i.e. M3 has a low V_{th}). We can skew each transistor to equally probable corners to achieve a desired joint probability. For example, we may skew the pull-down transistor to a point that is slower than 99.87% of all transistors (i.e., 3σ away from the mean, μ, on a normal distribution). At the same time we would skew the pass transistor to a point that is faster than 99.87% of all transistors. If we observe failure at this point, we conclude that failure may occur any time that the pull-down transistor is slower than the μ-3σ point or the pass transistor is faster than the μ+3σ point, giving a failure probability of 0.13% (3σ away from the mean on a normal distribution). Note that this is a conservative approach since we make the approximation that failure occurs any time either one of the devices is >3σ



Condition	WL	BL
Hold	0	V_{dd}
Read	V_{dd}	V_{dd}

Figure 3.13: (a) RDF V_{th} variability model in a 65nm device closely matches gate area dependence in Eq. 9. (b) V_{th} is approximately linear with dopant count (c) SRAM test circuit for measuring SNM and I_{read}/I_{leak}. Node voltages during hold and read conditions are also shown.

away from the mean, while the observed failure occurred when both devices were skewed to 3σ corners.

We can simplify our approach by making the following approximations: 1) that the dopant count distribution is approximately normal, and 2) that the dopant count maps linearly to a V_{th}. The first approximation allows us to use σ, 2σ, 3σ, etc. as meaningful metrics and the second approximation (which is shown to be reasonable in Figure 3.13b) allows us to say that the σ, 2σ, and 3σ points in the dopant distribution map directly to the σ, 2σ, and 3σ points in the V_{th} distribution. This model is used later in this section to approximately bound the characteristics of sub-V_{th} SRAM. A more accurate RDF model could be obtained by dividing the channel into cells with unique Poisson distributions [51] or by placing each dopant atom individually within the channel [52]. However, the simple model is useful since we only seek to identify the general characteristics that are

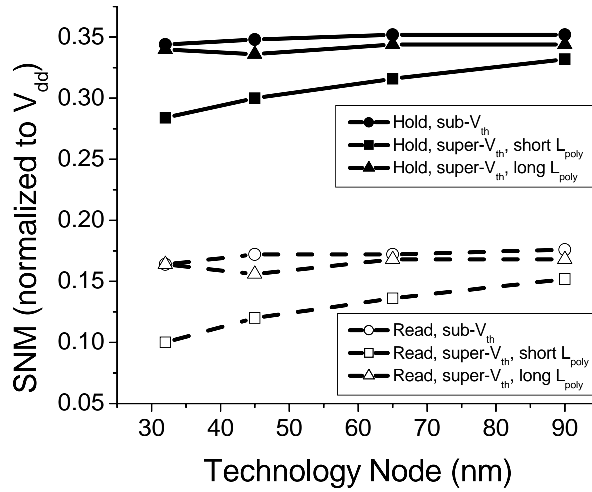


Figure 3.14: Simulated SNM in a 6T SRAM cell at V_{dd}=250mV under three different device optimization strategies: (1) sub-V_{th} optimized device, (2) unoptimized super-V_{th} device with minimum length, and (3) unoptimized super-V_{th} device with the same length as case (1).

favorable in scaled sub-V_{th} devices rather than predict accurate estimates of variation in future sub-V_{th} SRAM.

The nominal hold and read SNM of a 6T SRAM at V_{dd}=250mV are plotted in Figure 3.14 using the circuit and voltage settings shown in Figure 3.13(c). The voltage transfer characteristic is captured and reflected to construct a butterfly curve, which is then used to extract the SNM. For the unoptimized super-V_{th} device (called "super-V_{th}, short L_{poly}"), the nominal read SNM goes as low as 10% of V_{dd} at the 32nm node. Note that the read SNM at the 32nm node is 64% larger for the sub-V_{th} device (called "sub-V_{th}") than for the super-V_{th} device. Recall from Eq. 3.3 that noise margins have a strong dependence on S_s, which is dramatically improved in the device optimized for sub-V_{th} operation. It is interesting to note that the discrepancy in SNM can be nearly eliminated by increasing the lengths in the super-V_{th} devices to match those of the sub-V_{th} devices

(called "super-V_{th}, long L_{poly}" in Figure 3.14). However, the read/write delays will not be optimal since doping has not been re-optimized for the larger gate length.

Figure 3.15 shows the probability of a read failure for a single cell at each technology node for each device type. The butterfly curve, which is used to extract SNM, is again constructed by simulating and reflecting the voltage transfer characteristic of the circuit in Figure 3.13(c). A failing case has a read SNM below 6% of V_{dd} (15mV for V_{dd}=250mV). Failure is induced by progressively skewing the V_{th} of the pull-down transistor (M2) and the pass transistor (M3) such that the devices become weaker and stronger, respectively. The probability of the failing case is calculated based upon the individual probabilities of the pull-down and pass transistor V_{th} skew, as described in the previous discussion. Figure 3.15 also includes a 3σ failure probability line (~0.13% failure probability) for reference. We focus on the 3σ point since 99.87% yield should be sufficient for small SRAM arrays in sensor applications with several kilobits and column/row redundancy. The unoptimized device with short L_{poly} exceeds the 3σ failure probability at 90nm. The optimized device and the unoptimized device with long L_{poly} exceed the 3σ failure probability at 65nm, suggesting that simple device optimizations may extend the lifetime of sub-V_{th} 6T SRAM by one technology generation. It is interesting to again note that increasing gate length is responsible for most of the improvement in noise margins and could be used as a near-term fix for subthreshold memories using conventional devices.

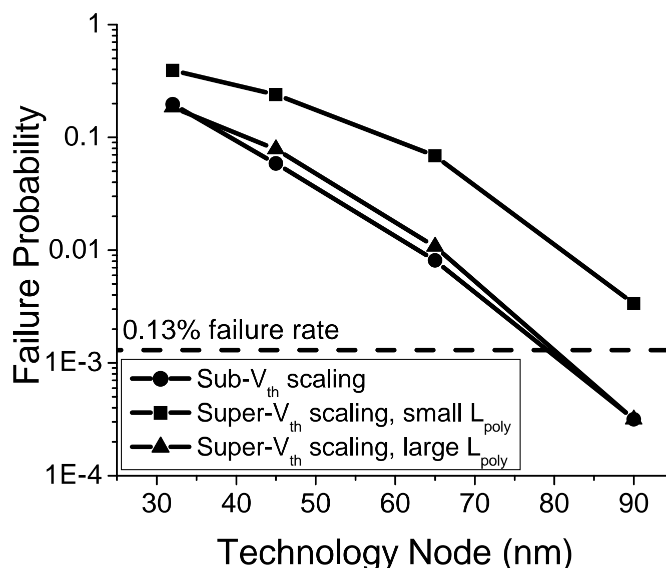


Figure 3.15: Read failure probability for a single SRAM cell under different device optimization strategies at V_{dd}=250mV. Failure is defined as the point where the read SNM drops below 6% of V_{dd} (15mV).

It is also interesting to consider the implications of I_{on}/I_{off} reductions observed in Figure 3.2. At 32nm, this ratio may drop below 500 for a device operating at 250mV. This is particularly challenging when reading data out of an SRAM cell since it becomes very difficult to distinguish between read current and bitline leakage. The ratio of the read-current to pass-gate leakage current in a 6T SRAM cell is approximately 238,000 at V_{dd}=1V in the 90nm technology node. Figure 3.16 shows this ratio is well below 1000 for all device types at V_{dd}=250mV. This ratio is extremely important since it is proportional to the number of bits allowed on a single bitline. At 90nm and 32nm, the sub-V_{th} device has a 1.7X and 3.7X larger current ratio, respectively, than the super-V_{th} device. These numbers can be reduced to 1.17X and 1.28X by increasing the lengths of the super-V_{th} devices to match those of the sub-V_{th} devices at the cost of increased read/write delay.

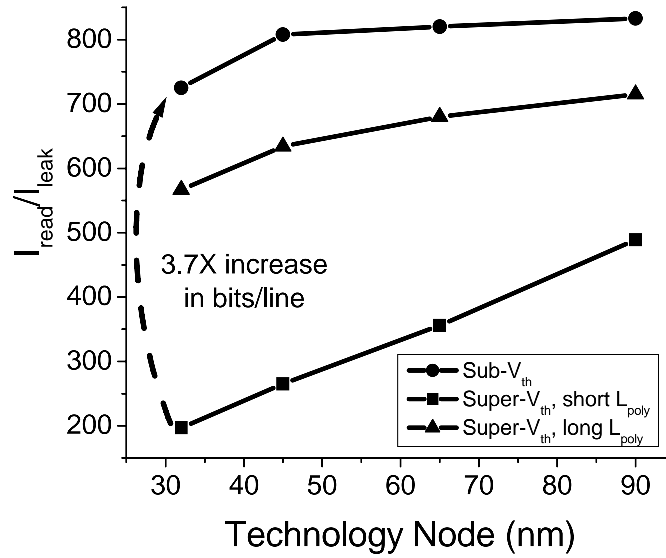


Figure 3.16: Ratio of read-current to pass-transistor leakage in a 6T SRAM at V_{dd}=250mV under super-V_{th} scaling and sub-V_{th} scaling. I_{read}/I_{leak} is proportional to the maximum number of bits per bitline and is therefore closely tied to SRAM area.

The data in this section suggest that the future is quite grim for scaled sub-V_{th} memories. Significant help can be offered at advanced technology nodes by the gate length and doping optimizations studied in this work. Radical device redesign may be required for large SRAM arrays. High-κ gate dielectrics will help improve channel control (and subsequently noise margins, delay, and energy). Multi-gate devices with lightly doped bodies will offer superior channel control and will eliminate RDF [17]. In the near term, however, designers need to focus on variability-aware circuit design techniques in combination with simple device modifications.

3.6 Conclusion

In this chapter the implications of device scaling on subthreshold operation were discussed in detail. In particular, this work demonstrated that the slow scaling of gate

oxide leads to 60% I_{on}/I_{off} degradation in the subthreshold regime. MEDICI simulations of simple circuits were used to illustrate the energy, performance, and robustness characteristics of scaled subthreshold devices. An alternative scaling strategy that uses larger gate lengths and reduced doping was proposed to achieve an improved inverse subthreshold slope. The proposed strategy maintains an S_S~80 mV/dec down to the 32nm node and offers a robust energy efficient alternative to conventional devices. A study of scaled subthreshold 6T SRAM suggested that read noise margins will be dangerously small due to variability at the 90nm node, but simple device modifications can push the problem out to the 65nm node. It is likely that new device geometries will be important for the aggressive scaling of subthreshold circuits, but the simple modifications described in this chapter may help subthreshold circuits reliably scale in the near term.

Chapter 4

The Subliminal Processor

A number of daunting challenges remain for subthreshold circuits. The most important concern is variability. Exponential sensitivities to V_{dd}, V_{th}, and temperature make even small variations problematic. Performance is also considerably degraded at low voltage since nodes are charged and discharged by weak inversion currents. The speeds of subthreshold digital circuits have typically been reported in the kHz and low MHz ranges [27][28]. To guarantee widespread adoption of subthreshold design, it will be necessary to address both of these issues.

In this chapter, the subthreshold design space is explored further with a particular emphasis on addressing the variability and performance problems at low voltage. Section 4.1 describes an 8-bit processor called Subliminal that has been fabricated in a 0.13µm technology [22]. The architecture is described in detail with emphasis placed on accommodations made for energy efficiency. Measurements show that the processor is functional below 200mV and that the total energy consumption is only 3.5pJ/instruction at V_{dd}=350mV. With the application of a reverse body bias, the power consumption goes as low as 11nW.

Section 4.2 describes a body biasing strategy that takes advantage of the unique sensitivities of subthreshold operation. The body bias sensitivities of subthreshold circuits

are contrasted with those of super-threshold circuits ($V_{dd}>V_{th}$). Measurements of the Subliminal Processor show that robustness at low voltages can be improved dramatically with the application of a body bias and that performance fluctuations induced by process and temperature variability can be eliminated with minimal energy penalties.

Finally, Section 4.3 explores techniques for improving performance in the Subliminal Processor. Body biasing and voltage scaling are first compared for improving performance globally. Sizing techniques for improving performance locally are then discussed in detail. At low voltages, gate length sizing can give an exponential increase in drain current due to reverse short channel effects (RSCE). Test-chip measurements show that gate length sizing is superior to gate width sizing for improving performance along timing critical paths.

4.1 Design Overview

In this section we describe the architecture of an ultra-low energy subthreshold processor designed for sensor applications. We discuss circuit and physical implementation details as well as energy and frequency measurements.

4.1.1 Architecture

While our energy efficiency improvements are primarily derived from aggressive voltage scaling, architectural decisions can have a dramatic impact on the energy efficiency of a system. We have accordingly adopted a simple processor architecture but have made a number of additions to enhance energy efficiency. A system-level diagram

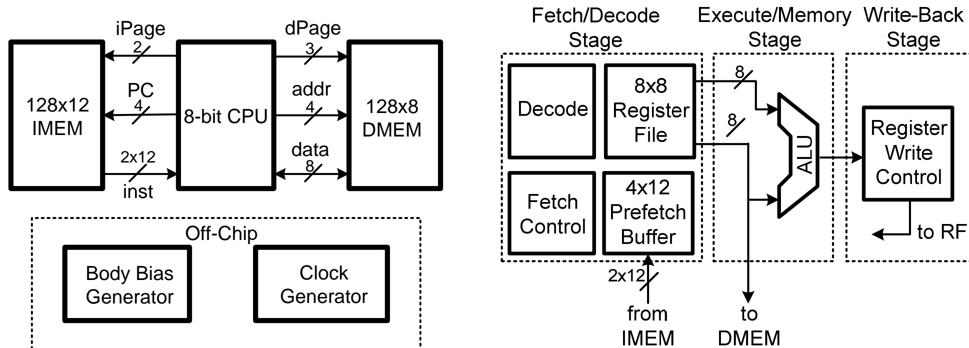


Figure 4.1: (a) System-level diagram of the 8-bit subthreshold processor (b) CPU implementation details

of the processor and CPU, which addresses a 1.5kbit instruction memory and a 1kb data memory, is shown in Figure 4.1(a) [22][53].

To minimize both decoding complexity and memory footprint, we choose a RISC-style architecture with an instruction width of only 12 bits. As we will see in subsequent sections, the memory energy demands can dominate the total energy consumption of the system, so these decisions are extremely important. To further reduce the energy consumption of the memories, we divide both data and instruction memories into pages of 16 words each. A special instruction pre-decodes the upper bits of the memory address (*iPage* and *dPage* in Figure 4.1(a)) and allows single cycle access to the contents of the specified page. Significant energy is saved when accessing multiple words within a page.

The CPU, shown in Figure 4.1(b), is a 3-stage pipeline with 8-bit data width. A highly pipelined design ensures that the majority of the logic is active throughout the clock cycle, thus minimizing time spent idly leaking. However, pipelining also requires additional sequential elements, which can be energy hungry. A 3-stage pipeline is attractive since it balances these competing trends [53]. We choose an 8-bit data width since the upper bits in a 16-bit or 32-bit processor would be idle for much of the computation in simple sensor applications, leading to an unnecessary leakage overhead.

Simple "taken" branch speculation has been implemented to reduce branch-related stalling in the CPU. A small 4-entry prefetch buffer helps facilitate this branch prediction.

4.1.2 Implementation

The 8-bit processor was implemented using a conventional digital synthesis and place-and-route design flow. All circuits were designed with the goal of maximizing robustness at low voltage. For example, the synthesis library included a limited subset of CMOS gates with a maximum fan-in of two. Gates with large fan-ins have been shown in previous work to have reduced noise margins at low voltage [17][54][65]. The instruction memory, data memory, and register file were implemented using a robust latch-based memory with a mux-based read-out structure [65]. While this memory structure is large and energy inefficient, it helps us to reliably explore the low voltage domain. Recently, several authors have proposed more compact low voltage memories that are promising for future subthreshold development [19][31][32][33][71][72].

The topic of physical design in low voltage circuits has been overlooked in previous work. However, it is important to observe that interconnect RC delay ($\sim 0.38 \cdot R_{wire} \cdot C_{wire}$) is only a function of materials and circuit geometry and does not scale with V_{dd}. Subthreshold current, on the other hand, is exponentially related to V_{dd}. Consequently, wire resistance (and wire RC delay) becomes insignificant compared to device resistance at low voltage. Figure 4.2 shows the effective resistance of an NFET device as a function of V_{dd}. The resistances of 100μm minimum width wires of various materials have been included for reference. At V_{dd}=300mV, the device resistance is >10,000 times greater than that of a 100μm wire in the first metal layer.

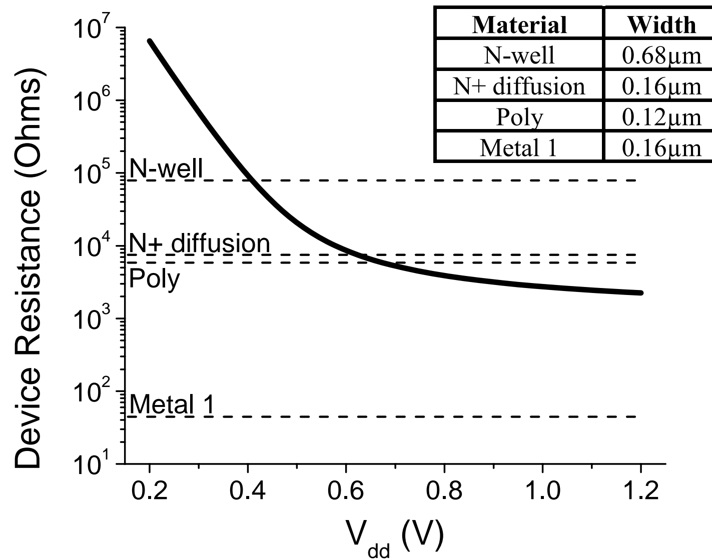


Figure 4.2: Effective NFET resistance as a function of V_{dd}. The resistances of wires of several alternative materials are included for reference (with 100µm length and widths from inset).

The reduced importance of RC delay has several important implications. Minimum width wires can be used for any interconnect with no penalty. We have leveraged this in our design by using minimum width metal for clock and power routing. This opens considerable routing area and reduces energy consumption in our clock distribution network. Interestingly, Figure 4.2 suggests that density could be further improved by shifting some of the routing to the poly and diffusion layers. In addition to thinner wire routes, the reduced importance of RC delay permits the use of a much simplified clock distribution network. The large capacitance of the clock network can be treated as a grid driven by a single level of clock drivers. This reduces design complexity and also minimizes skew induced by process, voltage and temperature variations, which can be severe in subthreshold circuits. In our design, we used a single clock buffer for all pipeline registers.

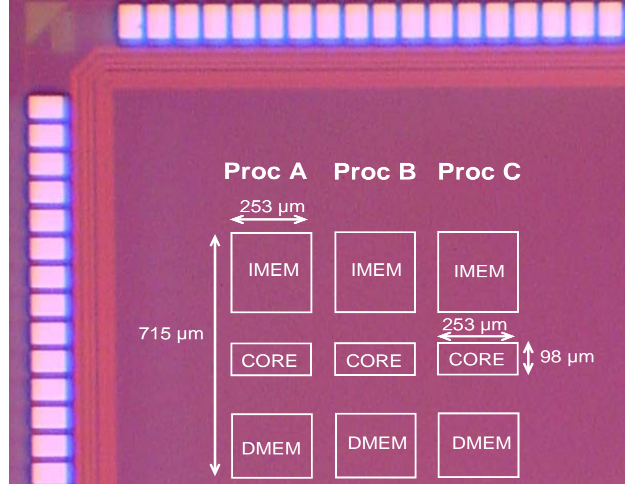


Figure 4.3: The 8-bit subthreshold processor was fabricated in a 0.13µm technology. Three CPU variants are shown.

We have fabricated the proposed processor in a 0.13µm technology with V_{th}~400mV at V_{ds}=50mV. The die photo for three processor variants is shown in Figure 4.3. Proc A uses minimum gate sizes while Proc B and Proc C use different sizing strategies that will be described in Section IV. Each processor has a footprint of 253µm x 715µm.

4.1.3. Energy and Frequency Measurements

Energy and maximum operating frequency measurements for the processor with minimum gate sizes (Proc A) are shown for a typical die in Figure 4.4. Both were measured for a simple arithmetic program that tests a wide range of instructions. The average current demand for the CPU and memories was measured over many program iterations using a high-precision electrometer. As predicted by [23][24], energy reaches a minimum due to increased leakage energy at low V_{dd}. The processor achieves a minimum of 3.5pJ/inst at V_{dd}=350mV with a frequency of 354kHz. The core (without memories, register file or prefetch buffer) reaches a minimum of 515fJ/inst at V_{dd}=290mV. The data

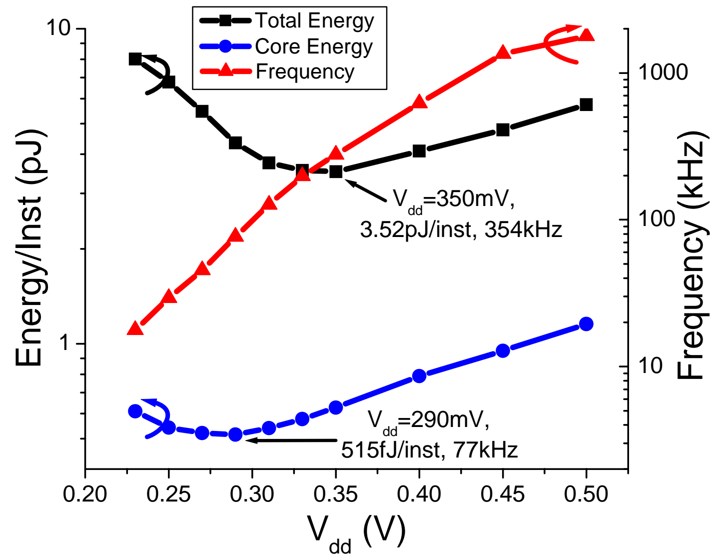


Figure 4.4: Frequency and energy measurements for a typical die as functions of V_{dd}

memory, instruction memory, prefetch buffer and register file consume >70% of the total energy, which is not surprising given that they are the most area intensive circuits. The processor remains functional down to ~210mV without a body bias but can function well below 200mV with the proper body bias (to be discussed in the next section).

In power-limited applications, such as those that scavenge ambient energy [55], a reverse body bias may be applied to minimize power consumption (as opposed to energy consumption). Under a reverse bias of 300mV, the processor draws 11nW at V_{dd}=160mV with a maximum frequency of 710 Hz. The core alone draws only 735pW. We focus on energy minimization for the remainder of this paper since it is more relevant for battery-powered applications.

4.2 Body Biasing for Variability Control

Process variability has the potential to be a crippling problem in subthreshold circuits. Even at the 0.13μm technology node, simulations show that 3σ/μ current variation for an NFET can be >200% [18]. In this section, we explore the use of body biasing in our subthreshold processor to combat global process and temperature variability.

4.2.1 Body biasing in subthreshold circuits

Variation is typically classified as random within-die, correlated within-die, or die-to-die. Random within-die variations in V_{th} and gate length are becoming particularly problematic in scaled devices [56]. Random V_{th} variations induced by random dopant fluctuations (RDF) were shown in [18] to be the dominant source of delay variability for subthreshold circuits simulated in a 0.13μm technology due to the exponential dependence of subthreshold current on V_{th}. Mismatch induced by RDF is also a significant threat to SRAM robustness [29]. This problem will only be exacerbated with device scaling to 65nm and beyond because of the gate area dependence of RDF [56]. The use of increased gate sizes has been shown to be an effective technique for reducing the effects of RDF for both logic [18] and SRAM [29]. Increasing the number of logic stages between sequential elements can also help reduce delay variations [18]. The focus of this work is addressing correlated within-die and die-to-die variations, but random within-die variations will require further attention in the near future.

Given the small size of our subthreshold processor, we group correlated within-die and die-to-die variations together under the name global variation. Global variations can

be either static process variations (e.g., V_{th}, gate length, gate oxide thickness) or temporal variations (e.g., temperature, V_{dd}) [56]. Subthreshold operation is dominated by an exponential dependence on V_{th}, so global variations in V_{th} due to doping fluctuations or those induced by gate length and gate oxide thickness variations are especially concerning. Temperature-induced V_{th} variations, which typically change over time, are similarly problematic. These variations lead to exponential variations in current, which lead to large fluctuations in both energy and delay. Global mismatch between $V_{th,NFET}$ and $V_{th,PFET}$ can lead to robustness problems as well. In the remainder of this section, we discuss the use of body biasing to address global V_{th} variations. Other sources of global variation (e.g., V_{dd} variation or S_S variation induced by gate oxide thickness variation) are important, especially as we scale to smaller device dimensions, but we focus in this work on first addressing V_{th} variations.

Body biasing, which effectively skews V_{th}, has been proposed for global V_{th} compensation in the past. The authors of [57] designed a multiply-accumulate unit that used adaptive supply voltage and body bias to minimize power in super-threshold circuits. Body biasing is a particularly effective technique in the subthreshold regime due to the exponential dependence of subthreshold current on body bias. The use of body biasing in subthreshold circuits was briefly explored in [25], but little attention was given to how the body bias should be selected and only limited measurements were presented. The authors of [58] showed that correct operation can be achieved with V_{dd} as low as 100mV by tuning body biases to match PFET and NFET leakages. In this work, we extend these early studies to develop a comprehensive body biasing strategy that accounts for the

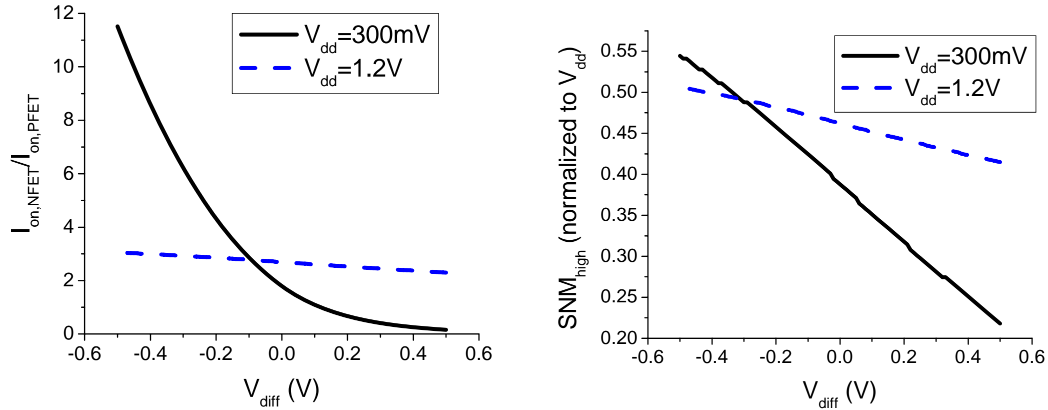


Figure 4.5: (a) The simulated ratio of NFET on-current ($I_{on,NFET}$) to PFET on-current ($I_{on,PFET}$) at two voltages (b) The simulated high static noise margins (SNM) at two voltages for an inverter with $W_{PFET}=2 \cdot W_{NFET}$.

unique sensitivities of subthreshold circuits. We also present detailed measurements of 20 measured dies to confirm the observed trends.

Throughout this section, we refer to two terms relevant to body biasing: *offset* and *differential*. The differential (V_{diff}) is the relative difference between the PFET and NFET body biases (i.e., $V_{diff}=(V_{dd}-V_{b,PFET})-V_{b,NFET}$), which may be tuned to skew the relative strengths of PFET and NFET devices, as shown in Figure 4.5(a) for two different supply voltages. We have chosen V_{dd}=300mV as a representative subthreshold voltage since it lies close to the minimum energy V_{dd} for our processor. As expected, the sensitivity to V_{diff} is particularly high at V_{dd}=300mV due to the exponential dependence of current on body bias. Balanced static noise margins (SNM) depend on matching between PFET and NFET strengths, so we can use V_{diff} to compensate for global V_{th} skew between PFET and NFET devices and maximize noise margins. Figure 4.5(b) shows the high static noise margins in a CMOS inverter with $W_{PFET}=2 \cdot W_{NFET}$ at V_{dd}=300mV and V_{dd}=1.2V. At V_{dd}=300mV, the high and low noise margins are balanced at V_{diff}=70mV.

The offset (V_{off}) is the shift in both the PFET and NFET biases relative to the ground voltage (i.e., $V_{off}=V_{b,NFET}$). A positive offset indicates a forward body bias (which reduces V_{th}) while a negative offset indicates a reverse body bias (which increases V_{th}). As shown in Figure 4.6 for an NFET, V_{th} changes with V_{off} at a rate of 170mV/V. Given this body bias sensitivity, a bias generation resolution of 5mV (which is assumed in later measurements) gives V_{th} resolution of <1mV. Figure 4.6 also shows how NFET on-current (I_{on}) changes with V_{off} at two different supply voltages. The increase in I_{on} with V_{off} is far more dramatic at subthreshold voltages (300mV) than at super-threshold voltages (1.2V).

It was shown in [23][24] that energy is independent of V_{th} as long as the circuit remains in the subthreshold regime. However, these derivations represent PFET and NFET devices with a single composite current expression, so they do not capture the energy dependence on V_{th} mismatch between NFET and PFET devices. Dynamic energy ($C \cdot V_{dd}^2$) is independent of PFET/NFET matching, but leakage energy can be separated into PFET-dependent and NFET-dependent components. Consider the leakage energy for a chain of identical inverters, where $I_{off,total}$ is the total leakage current, t_{total} is the delay of the inverter chain, $I_{off,N}$ and $I_{off,P}$ are the cumulative leakages through NFET and PFET stacks, $t_{p,N}$ and $t_{p,P}$ are the cumulative delays through NFET and PFET stacks, and k is a term accounting for delay degradation due to input slew:

$$\begin{aligned} E_{leak} &= I_{off,total} \cdot t_{total} \cdot V_{dd} = \left(I_{off,N} + I_{off,P}\right) \cdot \left(t_{p,N} + t_{p,P}\right) \cdot V_{dd} \\ &= \left(I_{off,N} + I_{off,P}\right) \cdot \left(\frac{k \cdot C \cdot V_{dd}}{I_{off,N} \cdot e^{\frac{V_{dd}}{m_N \cdot v_T}}} + \frac{k \cdot C \cdot V_{dd}}{I_{off,P} \cdot e^{\frac{V_{dd}}{m_P \cdot v_T}}} \right) \cdot V_{dd} \end{aligned} \quad \text{(EQ. 4.2)}$$

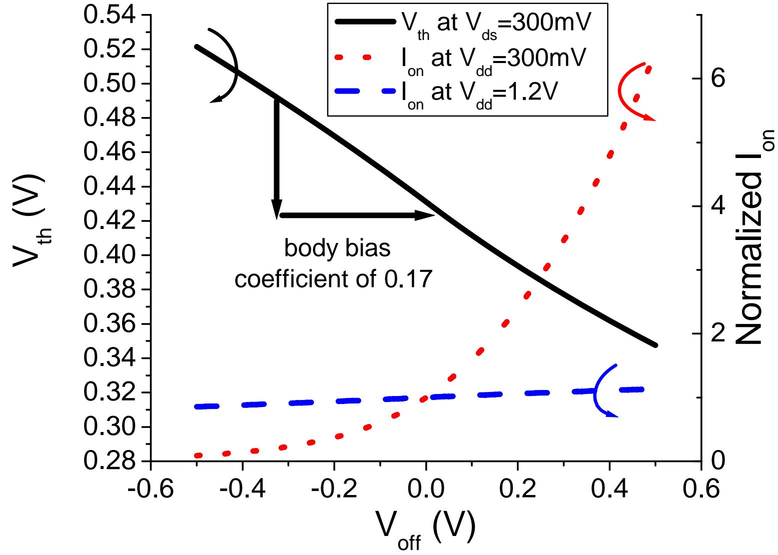


Figure 4.6: Simulated NFET V_{th} and I_{on} as functions of V_{off}

If we make the simplification that the NFET and PFET subthreshold slope factors are identical (i.e., m_p=m_n=m), then we can rearrange the leakage expression to highlight the dependence on NFET/PFET matching and take the derivative with respect to $I_{off,N}/I_{off,P}$:

$$\frac{\partial E_{leak}}{\partial \frac{I_{off,N}}{I_{off,P}}} = \left(1 - \left(\frac{I_{off,P}}{I_{off,N}}\right)^2\right) \cdot k \cdot C \cdot V_{dd}^2 \cdot e^{-\frac{V_{dd}}{m \cdot v_T}}$$

$$0 = \left(1 - \left(\frac{I_{off,P}}{I_{off,N}}\right)^2\right) \quad \text{(EQ. 4.4)}$$

We ultimately arrive at the conclusion that the NFET and PFET off-currents should be equal for minimum energy, which is the same condition that gives us balanced noise margins. For confirmation we simulate a chain of 30 identical inverters switching with an activity rate of 0.2 at V_{dd}=300mV. Inverter chains have been used extensively in previous work to explore the basic sensitivities of subthreshold circuits [23][24] and have proven to be good indicators of the trends observed in more complex circuits. Figure

4.7(a) shows that the energy consumed per cycle (the time it takes to propagate a single switching operation) for the inverter chain is minimized at V_{diff}=150mV, which matches well with the V_{diff} value that balances high and low noise margins (70mV). This simplifies bias generation since we need only match the leakage through PFET and NFET current monitors [58] to achieve both minimum energy and maximum noise margins.

While PFET/NFET mismatch is not captured by the formulas in [23][24], they do suggest that energy is not affected if both PFET and NFET threshold voltages are shifted in the same direction. To test this theory within the context of body biasing, we again simulate a chain of 30 inverters with a switching activity of 0.2 at V_{dd}=300mV over a range of V_{off} values. Figure 4.7(b) shows the energy consumed per cycle and the delay of the inverter chain. With a negative V_{off} (a reverse body bias), energy actually decreases. This seems to contradict the conclusion in [23][24] that energy does not depend on V_{th} in the subthreshold regime, but we make the added observation that inverse subthreshold slope, S_S, depends on the body bias. When a reverse body bias is applied, the depletion capacitance, C_d, reduces and yields improved S_S (Equation 4.5). Leakage energy, which is exponentially dependent on S_S through m (Equation 2), reduces with improved S_S.

$$S_{S.} = 2.3 \cdot v_T \cdot m = 2.3 \cdot v_T \cdot \left(1 + \frac{C_d}{C_{ox}}\right) \qquad \text{EQ. 4.5}$$

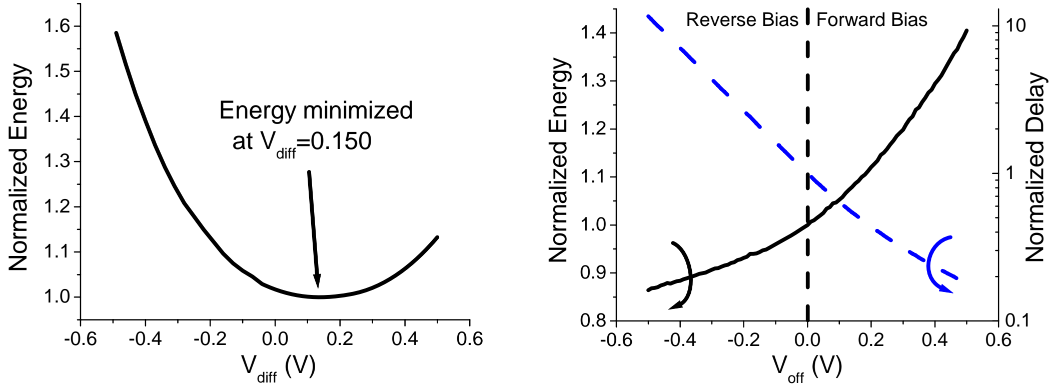


Figure 4.7: (a) Simulated energy consumption for a chain of 30 inverters at V_{dd}=300mV as a function of V_{diff} (b) Simulated energy and delay for the same inverter chain as functions of V_{off}

For positive V_{off} (a forward body bias), the delay of the inverter chain decreases quickly, but the performance improvement comes with a large energy penalty. The observed increase in energy is partially a result of degraded S_S at forward body biases (as predicted by Equation 4.5). Additionally, V_{th} reduces with a forward body bias and pushes the inverter chain into the near-threshold and super-threshold regimes. Outside of the subthreshold regime, the insensitivity of energy to V_{th} no longer holds [23][24].

Given the observed sensitivity of subthreshold circuits to V_{diff} and V_{off}, an effective body biasing strategy is clear. V_{diff} should first be tuned to achieve maximum noise margins and minimum energy. V_{off} can then be used to target a desired performance with only minimal energy consequences.

4.2.2 Body Bias Measurements

The processor described in Section II has been tested to verify our proposed body biasing strategy. Body biases were routed as normal signal nets using minimum width wires. External V_{dd}, body bias and clock generation were used (Figure 4.1) to enable a

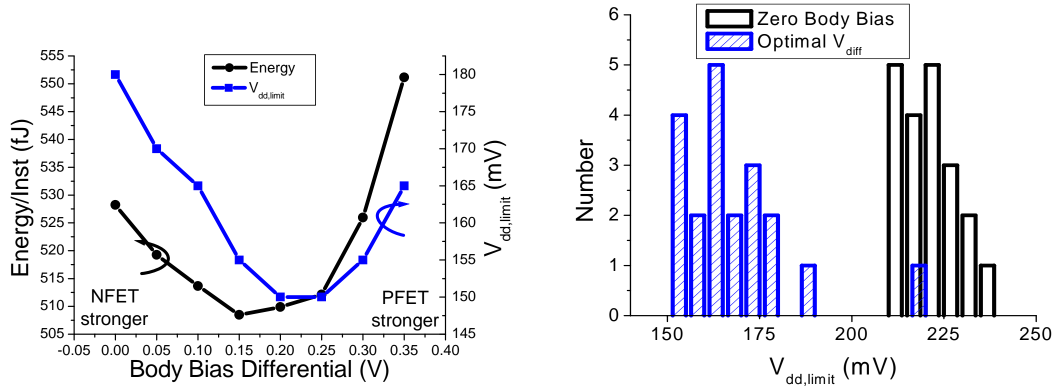


Figure 4.8: (a) Energy and $V_{dd,limit}$ as functions of V_{diff} for a typical die (b) $V_{dd,limit}$ distribution for 20 dies with and without body biasing. The mean $V_{dd,limit}$ reduces from 221mV to 168mV, a 24% improvement

fine-grained exploration of the energy-delay space. In this work, we quantify the energy and delay benefits of body biasing without considering the costs of V_{dd} and body bias regulation. Past work has explored the efficient generation and regulation of both V_{dd} [59] and body bias [25][58].

We first verify the observation that V_{diff} can be used to match PFET and NFET devices to maximize noise margins. Since noise margins are not readily measured for a processor, we use the minimum functional voltage, $V_{dd,limit}$, as a measure of robustness. The value of $V_{dd,limit}$ is extremely sensitive to PFET/NFET matching and is therefore a useful robustness metric [60]. Figure 4.8(a) shows that $V_{dd,limit}$ can be minimized (and noise margins can be maximized) by tuning V_{diff}. The energy consumption for the core (without memories, register file or prefetch buffer) at V_{dd}=300mV is shown for the same processor in Figure 4.8(a). Energy consumption and $V_{dd,limit}$ are minimized at nearly the same value of V_{diff}, thus confirming our simulation-based observations. By selecting the optimal value of V_{diff} for each of 20 measured dies (mean of 150mV across 20 dies), we

find that the mean value of $V_{dd,limit}$ reduces by 24% as compared to the case with zero body bias, as shown in Figure 4.8(b).

Figure 4.9 confirms for a typical die at V_{dd}=300mV that the tuning of V_{off} may be used to achieve an excellent energy-delay trade-off. Between V_{off} values of -400mV and -100mV, delay improves by 3.6X while energy varies by only 1%. Figure 4.9 also shows that the energy-delay trade-off begins to degrade with a forward body bias (V_{off}>0), which is consistent with simulation-based observations in the previous sub-section.

With proper selection of V_{diff} and V_{off} we can align all dies to a desired performance with limited energy penalties while also maintaining maximum noise margins. To demonstrate this, we measure the processor under four different scenarios. In Case 1, body biases are tied to the appropriate V_{dd} and V_{ss} rails (zero body bias). In Cases 2-4, the energy-optimal value of V_{diff} is applied, and V_{off} is chosen with 5mV resolution to meet frequency constraints of 66kHz (worst case frequency in Case 1), 100kHz, and 160kHz. The energy and frequency spreads for each of these cases are shown for 20 dies measured at V_{dd}=300mV in Figure 4.10. The inset table in Figure 4.10 summarizes the data from Cases 1 through 4 when all dies run exactly at the target frequency (66, 100 or 160kHz). A comparison of Cases 1 and 2 in the table shows that mean energy is reduced by ~10% with frequency fixed at 66kHz. Additionally, a comparison of Cases 1 and 4 shows that a 2.4X increase in frequency can be achieved while also achieving a 5% mean energy improvement. This excellent energy-delay trade-off makes body biasing extremely attractive for adaptive subthreshold systems.

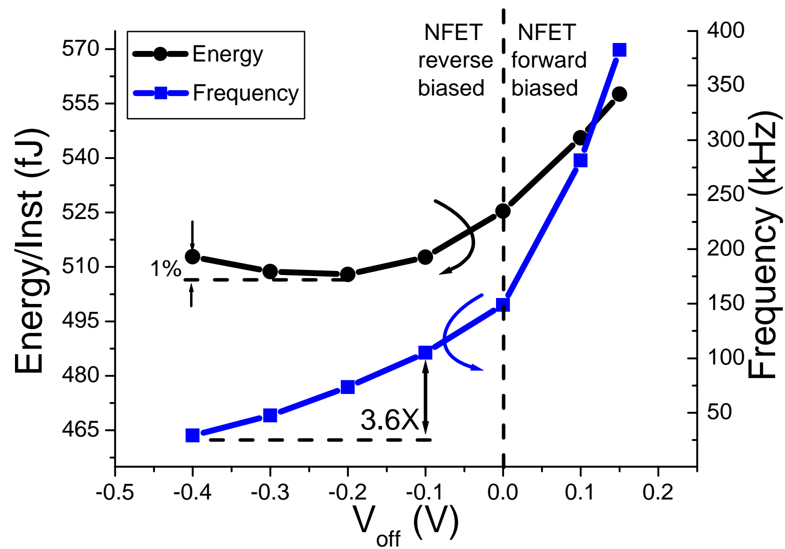
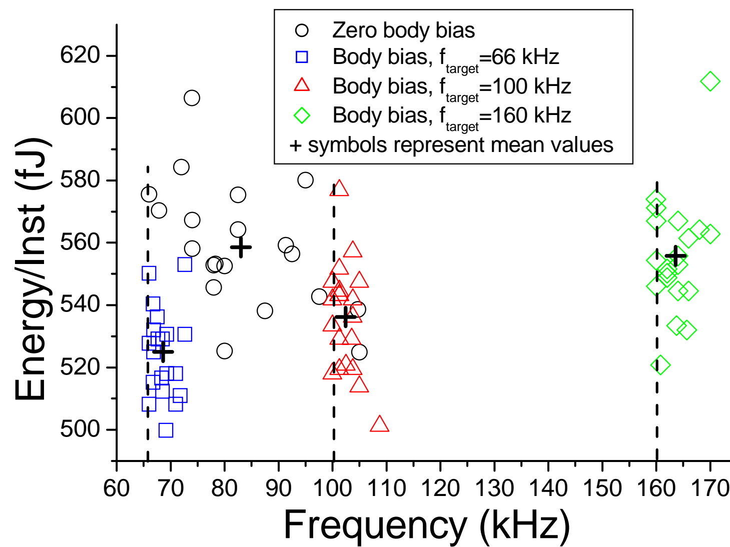


Figure 4.9: Energy and frequency as functions of body bias offset for a typical die



	Frequency (kHz)	μ_{energy} (fJ)	σ_{energy} (fJ)
Case 1: Zero body bias	66	588	22
Case 2: Variable body bias	66	528	14
Case 3: Variable body bias	100	538	17
Case 4: Variable body bias	160	559	20

Figure 4.10: Energy and frequency distributions for 20 dies measured at V_{dd}=300mV

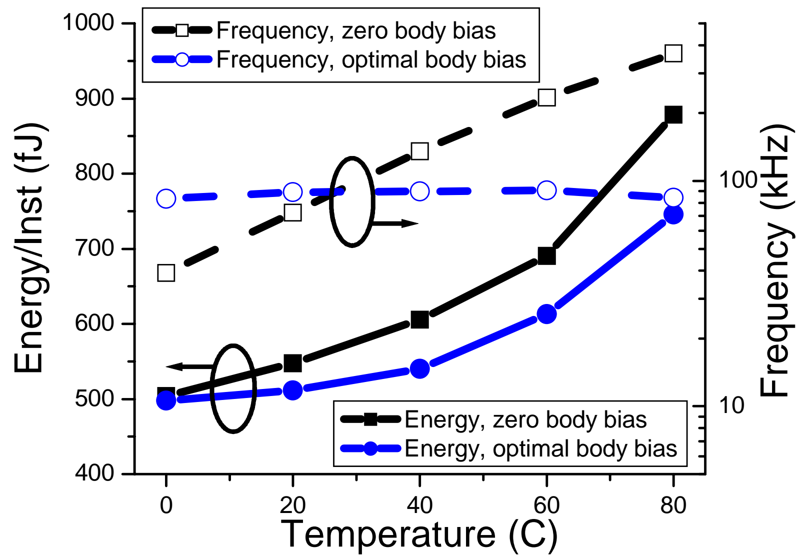


Figure 4.11: Temperature sensitivity of energy and frequency for a typical die at V_{dd}=300mV

The favorable energy-delay trade-off achieved using body biasing can be extended to compensate for temperature variations. Temperature compensation has been demonstrated in the past for subthreshold circuits using simple temperature sensitive bias generation [25]. Figure 4.11 shows the temperature dependence of energy and performance for a typical die at V_{dd}=300mV. Without body biasing, the frequency of the chip increases by ~10X between T=0C and T=80C. For a fixed value of V_{diff}, V_{off} can be tuned to maintain a constant frequency as shown in Figure 4.11. For this particular die, V_{off} changes by 620mV between T=0C and T=80C to maintain constant performance.

4.3 Improving Performance

On-currents in the subthreshold regime can be >5 orders of magnitude lower than super-threshold on-currents, so reduced performance is inevitable. Performance is only a

secondary concern in sensor network processing, but improved performance is necessary to make subthreshold operation viable in the embedded and high performance application spaces. In this section, we begin by comparing voltage scaling and body biasing for improving performance globally. We also look at the use of gate length sizing to achieve local performance improvements

4.3.1. Improving Global Performance

At the block level, body biasing and voltage scaling can both be used to achieve exponential improvements in performance. We are interested in determining which technique gives the better energy-delay trade-off for subthreshold circuits. To do so, it is necessary to understand the energy implications of body biasing and voltage scaling. Consider a simple chain of inverters operating at a subthreshold voltage, $V_{dd,init}$, with zero body bias. A wide range of target frequencies can be achieved by changing $V_{dd,init}$ or by changing V_{off}. The energy consumption of the inverter chain may be modeled as the sum of dynamic energy and leakage energy, where α is the switching activity, t is the maximum delay, and all other quantities are as defined previously:

$$E_{total} = E_{dyn} + E_{leak} = C \cdot V_{dd}^2 \cdot \alpha + I_{off} \cdot t \cdot V_{dd} \qquad \text{EQ. 4.7}$$

The relative energy efficiencies of V_{dd} scaling and body biasing are strong functions of α. To illustrate this, we consider limiting behavior. In the case of very high switching activity, dynamic energy is dominant ($E_{dyn} >> E_{leak}$):

$$E_{total} \approx E_{dyn} = C \cdot V_{dd}^2 \cdot \alpha \qquad \text{EQ. 4.8}$$

In this limit, energy has a quadratic dependence on V_{dd} and is, to first order, independent of V_{off}. For low target frequencies, we should reduce V_{dd} or apply a reverse body bias

(V_{off}<0). A reduction in V_{dd} will yield quadratic energy reductions, while the application of a reverse bias will have no effect on energy. Voltage scaling is therefore more energy efficient for low target performance. Conversely, energy increases quadratically with V_{dd} so body biasing is more energy efficient when the target frequency is high.

In the case of very low switching activity, leakage energy is dominant ($E_{leak}>>E_{dyn}$):

$$E_{total} \approx E_{leak} = I_{off} \cdot \frac{C \cdot V_{dd}}{I_{off} \cdot e^{\frac{V_{dd}}{m \cdot v_T}}} = \frac{C \cdot V_{dd}}{e^{\frac{V_{dd}}{m \cdot v_T}}} \qquad \text{EQ. 4.9}$$

For $V_{dd}>m \cdot v_T$ in the subthreshold and near-threshold regions, leakage energy per cycle increases as V_{dd} decreases [23][24]. Energy therefore increases when V_{dd} is reduced to meet low frequency targets but reduces when V_{dd} is increased to meet high target frequencies. Due to the m dependence of leakage energy, energy reduces with the application of a reverse body bias and increases with the application of a forward body bias. In this case, V_{dd} scaling is more energy efficient for high target frequencies while body biasing is more energy efficient for low target frequencies. Note that our observations are only valid for subthreshold circuits. Outside of the subthreshold region, delay is no longer exponentially dependent on V_{dd} and V_{th}, and the energy-performance trade-offs change dramatically.

Neither of the two limits considered reflects actual circuit behavior since dynamic energy and leakage energy are comparable in a typical subthreshold circuit [23]. For a more realistic comparison of voltage scaling and body biasing, we simulate a chain of 30 inverters with switching activities of 0.05, 0.2, and 1. We select a nominal V_{dd} of 300mV. For voltage scaling data, we sweep V_{dd} from 200mV to 500mV. For body biasing data,

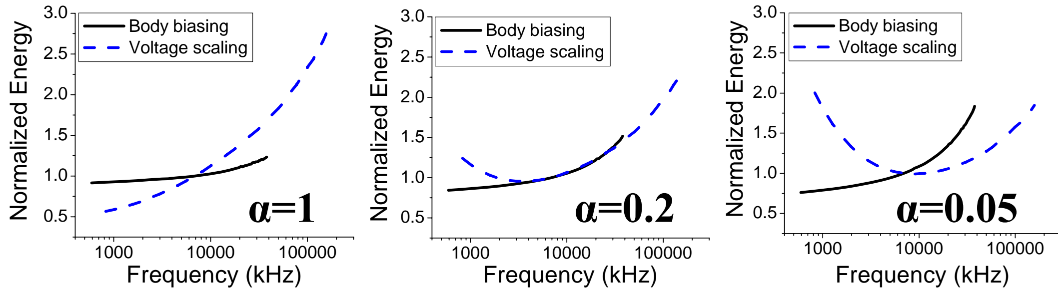


Figure 4.12: Simulated energy and frequency for an inverter chain subjected to voltage scaling and body biasing. Data is plotted for switching activities of 1, 0.2, and 0.05.

we sweep V_{off} from -500mV to 500mV. The data in Figure 4.12 for switching activities of 1 and 0.05 confirm our previous observations about the high activity and low activity limits, respectively. Figure 4.12(b) shows the energy characteristic for a more realistic switching activity of 0.2. In this scenario, body biasing and voltage scaling give similar energy-performance trade-offs for much of the performance range, suggesting that either body biasing or voltage scaling could be used for minimum energy in a typical circuit. Body biasing may be a more attractive option in this case since the low current demands of the body node simplify bias generation as compared to supply voltage regulation.

4.3.2. Global Performance Measurements

To verify the trends observed in the previous sub-section, we measure the energy and performance of the core (without memories, register file or prefetch buffer) over a range of supply voltages and body biases. To evaluate V_{dd} scaling, we fix the body biases at zero and sweep V_{dd} from 260mV to 380mV. To evaluate body biasing, we fix V_{diff} at the energy-optimal value and sweep V_{off} from -400mV to 150mV to tune performance. In both cases, fine-grained regulation and bias generation would be required. The energy

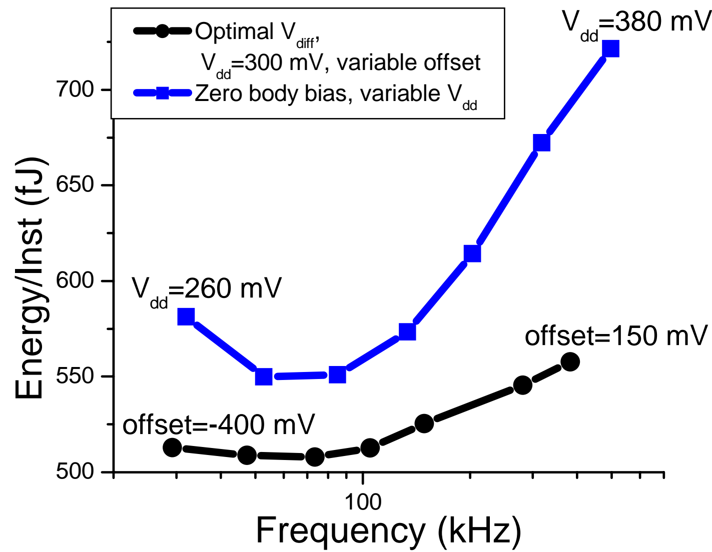


Figure 4.13: A comparison of energy and frequency measurements for variable body bias and variable V_{dd} systems

penalties for these regulators have not been included, though we note that the body node draws very little current, so the body bias is generally simpler to regulate than V_{dd}. Figure 4.13 shows the resulting energy consumption over a frequency range of 30kHz to 300kHz. The characteristic is similar to that of Figure 4.12(b), but we find that body biasing is more energy efficient over the entire frequency range. The observed energy improvement is due to the PFET/NFET matching achieved through tuning of V_{diff}. The ability to achieve PFET/NFET matching as well as a favorable energy-performance trade-off makes body biasing an attractive alternative to voltage scaling in subthreshold circuits with tight performance requirements.

4.3.3. Subthreshold Sizing Strategies

Techniques for improving performance along timing-critical paths are also important for subthreshold circuits. Gate width sizing is typically used to speed up critical

paths in super-threshold circuits, but recent work has shown that gate length sizing can be used to improve drive strength in subthreshold circuits due to reverse short channel effects (RSCE) [61]. Halo doping increases the effective doping at short channel lengths to help combat drain induced barrier lowering (DIBL) [62]. However, since DIBL is much reduced at low V_{dd}, the halo doping overcompensates and increases the V_{th}. Drain current can therefore be increased significantly in subthreshold and near-threshold circuits (0.65V and below in the target process) with a small increase in gate length. The simulated on-current of an NFET device at V_{gs}=250mV and V_{ds}=250mV is shown as a function of total device capacitance (gate capacitance plus drain capacitance) for both increased gate width and increased gate length in Figure 4.14. The current-capacitance trade-off is far more attractive when increasing gate length than when increasing gate width. This discrepancy is largely due to RSCE, but capacitance also increases more slowly with gate length than with gate width. Gate oxide capacitance depends identically on gate width and gate length, but overlap and junction capacitance are independent of gate length. The effectiveness of gate length sizing eventually saturates, suggesting that gate width sizing should be used after the benefits of gate length sizing have been exhausted. Note that the results shown are highly technology dependent, so performance gains will vary from process to process.

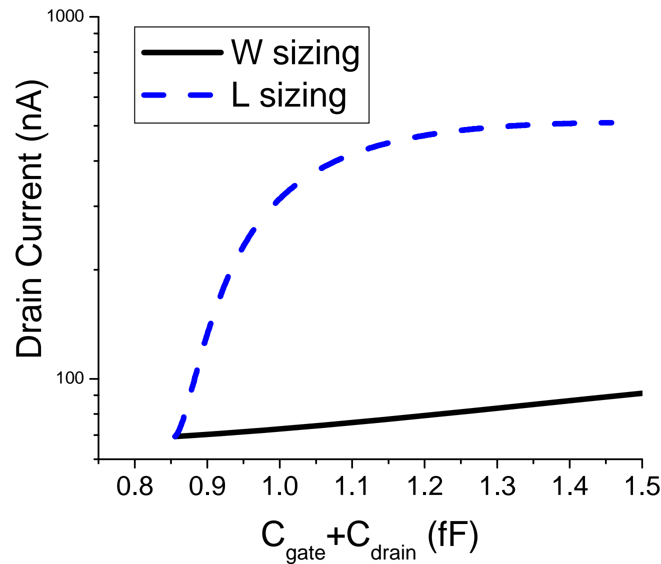


Figure 4.14: Simulated on-current for an NFET as a function of total device capacitance. The trade-off is shown for both gate width and gate length sizing.

4.3.4. Subthreshold Sizing Measurements

To explore gate sizing further, we designed three variants of the processor described in Section II. The first variant (Proc A in Figure 4.3) uses minimum gate sizes in the core. The second variant achieves reduced delay along critical paths using a conventional standard cell library with increased gate widths (Proc B). The third variant achieves reduced delay along critical paths using a custom standard cell library with both increased gate length and increased gate width (Proc C).

Both standard cell libraries were limited to a small set of inverters, 2-input NAND gates, 2-input NOR gates, and flip-flops. The drive strengths in the custom standard cell library used in Proc C were tuned to match those of the conventional standard cell library used in Proc B (i.e., X1, X2, X4 cell strengths drive the same current in both libraries). Each library cell was characterized over a range of low voltages using SPICE.

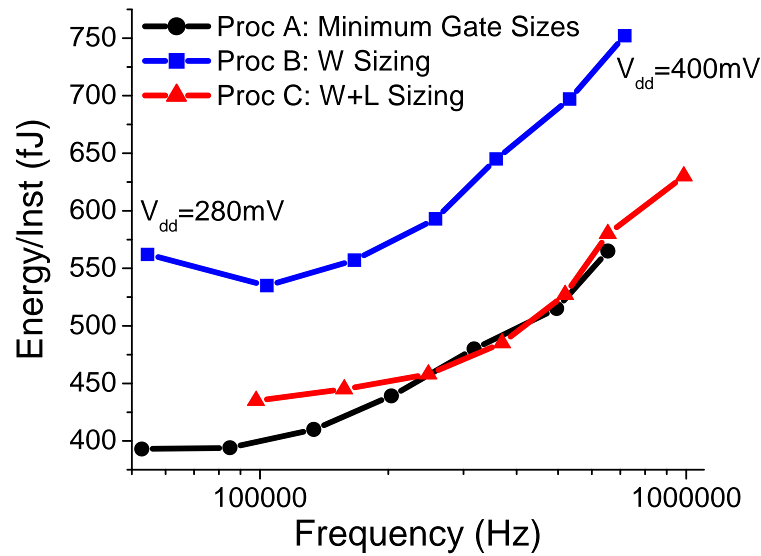


Figure 4.15: Energy and frequency for three sizing strategies for V_{dd}=280-400mV

The core gates sizes were optimized separately for Proc B and Proc C with energy as the objective according to the technique proposed in [63]. It was shown in [63] that the energy of a subthreshold circuit can be reduced by increasing gate sizes along critical paths due to the timing dependence of leakage energy. Proc B and Proc C were therefore designed with different frequency and energy targets that were determined by the characteristics of the standard cell library available during gate sizing. After sizing, Proc B and Proc C had total transistor gate areas that were 98% and 24% larger than the gate area in Proc A, respectively.

Figure 4.15 compares the energy-delay trade-off for the three different sizing strategies for V_{dd}=280mV to V_{dd}=400mV. We find that energy does not improve in Proc B and Proc C relative to Proc A, which is contrary to the conclusions in [63]. In [63], it was shown that the cost of reducing energy rises quickly after the gates along the first few critical paths have been sized up. Further sizing after initial energy gains leads to a

considerable area penalty and potentially an energy penalty if standard cell energy characterization models do not match post-silicon performance. It is likely that the unexpectedly high energy consumption in Proc B and Proc C was caused by this effect.

Though the comparison between Proc B/C and Proc A revealed that the larger gate sizes increase energy consumption, we can still draw valuable conclusions about the effectiveness of gate length sizing by comparing the energy and performance of Proc C to that of Proc B. At V_{dd}=300mV, Proc B and Proc C are 22% and 85% faster than Proc A, respectively. Furthermore, Proc C is both faster and more energy efficient than Proc B over the V_{dd} range shown, confirming the superiority of gate length sizing over gate width sizing. For target frequencies above 200 kHz, the energy consumption of Proc A is comparable to that of Proc B, suggesting that the performance gained from gate length sizing could be alternatively achieved by increasing V_{dd} by 20-30mV.

4.4 Conclusion

The measurements of the Subliminal Processor presented in this chapter provided valuable insights about subthreshold operation. It was shown that energy optimality was achieved in the subthreshold region, thus confirming simulated results presented in [23][24]. Two of the remaining challenges for low voltage operation, increased variability and reduced performance, were also explored. Though the Subliminal Processor was a good initial demonstration of subthreshold operation, standby power minimization was not addressed. In the next chapter, the importance of standby power in cubic-millimeter computers is underscored and a new processor with an ultra-low power standby mode is discussed.

Chapter 5

The Phoenix Processor

Recent work [22][25][28][64][65], including this work, has shown that aggressive supply voltage scaling into the subthreshold regime yields dramatic active mode energy reductions in digital circuits. As active energy reduces, standby energy becomes a problem that can no longer be ignored. Consider a $1mm^3$ computing system that wakes up once every 10 minutes to measure temperature and run simple data analysis routines. During an active period, this system may remain awake for 100ms to run the required routine and then return to standby for another 10 minutes. Using the active power (789nW) and standby power (153nW) measurements for the Subliminal Processor with zero body bias at V_{min}=350mV, the total energy consumed during a 10 minute standby period is 92µJ while active energy only accounts for 79nJ (a standby to active energy ratio >1000). Though the Subliminal Processor has a simple standby mode that includes only clock gating (and does not include other effective techniques like power gating), this example clearly underscores the importance of power consumption in standby mode.

In this chapter, a new test-chip, called the Phoenix Processor, is discussed. The Phoenix Processor addresses the increased importance of standby power in low voltage systems. In addition to a Subliminal-like processor, Phoenix includes special low leakage SRAM, power management, watchdog timers, and a temperature sensor. Phoenix was

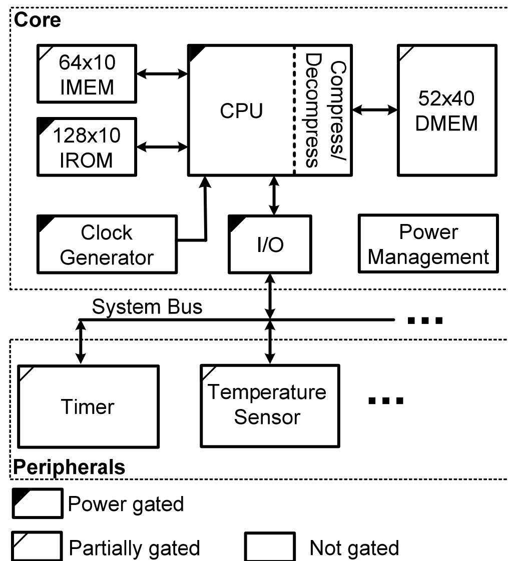


Figure 5.1: The Phoenix Processor

designed for use in 1mm^3 computing with particular attention to standby mode power management. Phoenix was fabricated in an area of 915x915µm^2 in a 0.18µm process, and measurements show that the full system consumes only 35.4pW in sleep mode and 226nW in active mode. Given a 10 minute standby period between active periods of ~100ms (as in the previous example) the Phoenix Processor draws ~73pW on average over its lifetime with a balanced standby energy to active energy ratio of 0.94. With such low power consumption, on-chip batteries or energy scavenging become viable options.

5.1 System Overview

As shown in Figure 5.1, the Phoenix Processor is a modular system with a core unit consisting of an 8-bit CPU, a 52x40-bit data RAM (DMEM), a 64x10-bit instruction RAM (IMEM), a 64x10-bit instruction ROM (IROM) and a power management unit

(PMU). The core serves as a parent to peripheral devices, including a watchdog timer and a temperature sensor. The core and peripheral devices communicate over a system bus using a simple asynchronous protocol. The I/O controller addresses up to 8 peripherals on the system bus for sensing systems requiring additional peripherals.

In typical operating conditions, the Phoenix Processor spends an extended period of time in standby mode (e.g., 10 minutes) and wakes up in response to an exception raised by the watchdog timer (a 0.9pW current-starved oscillator). Once awake, the Phoenix Processor polls the temperature sensor and runs a short routine to process and store the measurement. After completing the data processing routine, Phoenix returns to standby mode.

The power consumption in active mode is dominated by components with high switching activity, such as the CPU. To minimize this source of power consumption we scale voltage aggressively to 0.5V, a sub-threshold voltage (for high-V_{th} devices) or near-threshold voltage (for medium-V_{th} devices) in the target technology. The challenges of low voltage digital design have been covered extensively in recent literature [22][28][65] and will not be the focus of this work. Instead, we place emphasis on accommodations made for standby mode operation. The Phoenix Processor was designed at the device, circuit and architecture levels with the primary goal of standby power minimization. In subsequent sections, we discuss each of the key components of this comprehensive standby mode strategy.

5.2 Technology Selection

Despite its importance to both power and performance, there has been little investigation of technology selection for low voltage circuits. The requirements of sub-threshold and near-threshold circuits are different from those of normal super-threshold circuits, and the optimal technology is therefore different. The required performance is much relaxed in typical low voltage sensing applications, so older technologies can easily meet performance requirements. Furthermore, the long standby time observed in many sensor applications makes cumulative standby leakage energy significant, as we observed earlier in this work. Advanced technology nodes have also been optimized exclusively for super-threshold operation, resulting in sub-optimal noise margins, power and performance [77].

The ideal technology would simultaneously offer small feature sizes and devices with ultra-low leakage. Since no such technology was available for use in academic research, we investigated standard CMOS technologies from 0.25μm to 65nm. Newer technologies tend to offer devices with higher subthreshold leakage but smaller capacitance. Conversely, older technologies offer lower subthreshold leakage but larger capacitance, effectively offering reduced standby power at the expense of active power. A simple analysis using the method in [76] with simple inverter-chain models of a CPU and memory and a duty cycle of 0.001 (1s of active time per 1000s of standby time) reveals that an older 0.18μm technology gives optimal energy. The active power penalty paid for adopting an older technology is easily outweighed by the dramatic reduction in standby power. Note that the 0.18μm technology under study offers a device with a particularly high V_{th}, and further reverse scaling may have been warranted if the 0.25μm

technology offered a similar device. A processor implemented in the energy-optimal 0.18µm technology is 7.7X larger than a similar processor in a 65nm technology, but our analysis reveals that total energy is reduced by 647X. This is an extremely favorable trade-off, especially when the volume of a wireless sensor is dominated by the battery volume (and not die volume).

The selected 0.18µm technology includes a thin-oxide medium-V_{th} device with V_{th}~0.5V and a thick-oxide IO device with V_{th}~0.7V. All retentive gates (i.e., those gates that remain awake in standby mode) are implemented using the high-V_{th} devices, which consume ~1000X less leakage power per unit of gate width than the medium-V_{th} devices. Note that we do not use high-V_{th} devices in non-retentive gates since the minimum dimension is larger than that of the thin-oxide device, which gives both area and active energy penalties. In addition to the selection of an older technology, stack-forcing is used to reduce leakage power further. Leakage reduction due to the stack effect has been shown in previous work to be effective [74]. In our selected technology, stacking two transistors gives ~2X leakage reduction.

5.3 Power Gating Under Relaxed Performance Constraints

A power gating switch, as shown in Figure 5.2, is often used in low power circuits to minimize leakage in non-retentive circuit blocks during standby modes. At normal super-threshold operating voltages (e.g., >1V in 45nm and 65nm designs), a high-V_{th} device is typically used as a power gating switch since it delivers comparable on-current to the nominal device with exponentially smaller off-current. Additionally, wide power gating switches are typically used to minimize the performance penalty of power gating.

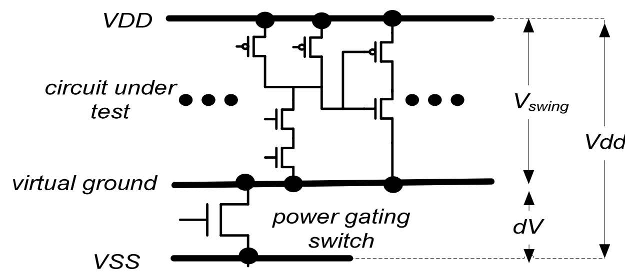


Figure 5.2: A typical power gating switch

For cubic-millimeter computing applications with modest performance requirements, minimizing standby power is the most important goal. In such applications, performance can be sacrificed for lower leakage, which is in stark contrast to the typical approach to power gating. In the Phoenix Processor, we leverage these modest performance requirements with an alternative power gating approach.

Our power gating approach relies on a medium-V_{th} power switch rather than a high-V_{th} switch as in the typical approach. The on-current of the high-V_{th} device is exponentially smaller than that of the medium-V_{th} device at low voltage. Therefore a high-V_{th} device must be sized up ~1000X as compared to a medium-V_{th} device to meet the current demands of the primary circuit, which is implemented using medium-V_{th} devices. The area overhead as well as the power overhead of charging/discharging such a large switch is avoided by using a medium-V_{th} power switch at no performance or leakage penalty. Note that the difficulties associated with high-V_{th} power switches in low voltage circuits as well as the utility of switches with lower V_{th} have been noted previously [82].

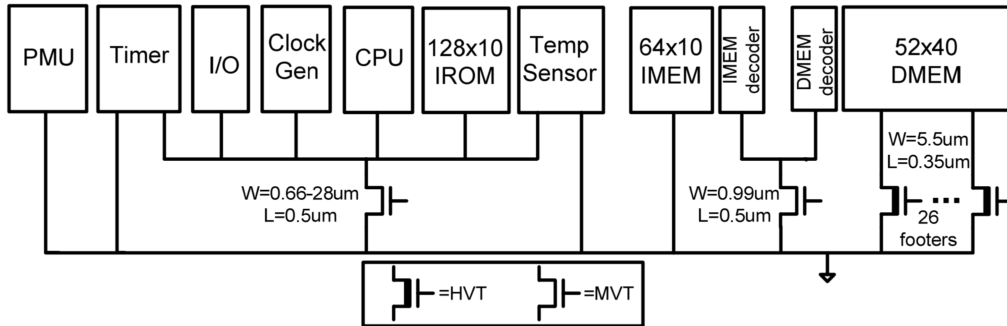


Figure 5.3: Footer allocation in the Phoenix Processor

In addition to using medium-V_{th} power switches, the strength of our power gating switch compared to the circuit under test is smaller than that of the typical power gating approach. A stronger power gating switch minimizes the performance penalty of power gating at the expense of additional leakage during standby mode. Given the modest performance demands for the Phoenix Processor, we choose to reduce standby mode leakage considerably by selecting a very weak power gating switch [73].

In the Phoenix Processor, the medium-V_{th} power switch is only 0.66µm, which is 0.01% of total effective NFET width and 3X larger than the minimum width in the target technology. We increase the length from 0.18µm to 0.50µm to improve inverse subthreshold slope and consequently increase the on-current to off-current ratio. The 0.66µm power gating switch is connected to the CPU and several other logic blocks as shown in Figure 5.3. Simulations with a model of the CPU indicate that the virtual ground rail bounces by a maximum of ~100mV, which is sufficient to guarantee correct logic operation. The non-retentive parts of IMEM and DMEM, such as decoders and output buffers, are connected to a separate power gating switch since the robustness of low voltage memory may be compromised by a voltage drop across the power gating

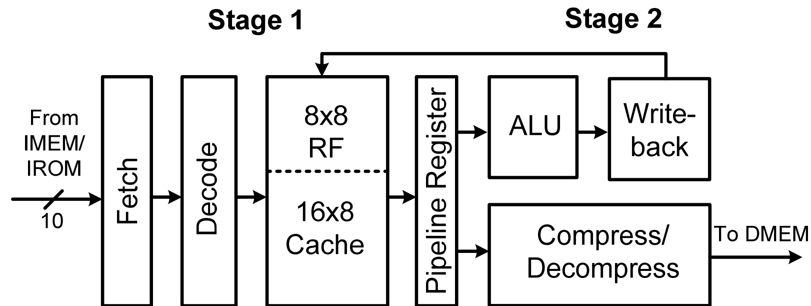


Figure 5.4: CPU diagram

switch. The measured energy and performance implications of our proposed power gating strategy will be discussed in Section 5.8.

5.4 CPU and Instruction Set Design for Standby Mode

In accordance with the conclusions of previous studies of subthreshold processor architectures [79], we have selected a simple CPU architecture with 2-stage pipeline, 8-bit data width, and 10-bit instruction width to reduce active mode power and standby mode power. The instruction set includes support for basic arithmetic computation in typical sensor logging applications. As shown in Figure 5.4, the first pipeline stage consists of instruction fetch and decode as well as a scratch memory with an 8-entry register file and 16-entry cache. The second pipeline stage includes a simple ALU, write-back logic, and a memory interface unit that compresses (decompresses) outgoing (incoming) memory traffic. The ALU includes hardware for addition, subtraction, and shifting. The CPU has been designed to minimize energy in both active and standby modes, as shown in the remainder of this section.

Class	Members	Addressing Mode
Arithmetic	ADD, ADDI, SUB, MOVE, SHR	explicit
Flow Control	BEQZ, JUMPI JUMPR	explicit
Compression	COMP, DECOMP	implicit
	FREE	explicit
Load/Store	LOAD, STORE, STORE_OVER	implicit
Wake	GET_REQ, SEND_REQ, SEND_ACK	implicit
Sleep	HALT	--

Table 5.1: Instruction set architecture overview

Since the computational demands of cubic-millimeter computing applications are typically modest, the CPU was simplified to support a minimum set of operations. Such simplicity reduces decode complexity and eliminates unnecessary switching activity, thus reducing active mode power. Furthermore, elimination of complex operations like multiplication eliminates large, leaky circuit blocks. Since leakage energy can be >30% of total energy in active mode for low voltage circuits [23], the resulting active mode power savings are significant.

Instruction set architecture (ISA) optimization also plays an important role in minimizing power consumption in standby mode. Since the contents of IMEM must be retained in standby mode, it is important to minimize the instruction width. The leakage

penalty of instruction memory can alternatively be eliminated by using flash-based memory, but this requires costly processing steps. The custom ISA for the Phoenix Processor was compressed to an instruction width of only 10 bits by selecting a minimum set of 18 instructions (Table 5.). The benefits of selecting a narrow instruction width will be quantified using measured results in Section 5.8.

Efficient operand encoding also helps to reduce the instruction width. Stack-based ISAs give very short instructions since the operands and destination register are implicitly assumed to be at the top of the stack. However, this small instruction size comes at the cost of flexibility offered by the typical approach in which operands are selected within the instruction from a set of general purpose registers. To simultaneously achieve encoding efficiency and flexibility in operand specification, two instruction types are available in the Phoenix ISA: *explicit operand* and *implicit operand*. *Explicit operand* instructions use a 3-bit opcode and a 7-bit operand specifier similar to a conventional register-register ISA. As shown in Table 5., this instruction format is reserved for arithmetic and flow control instructions, which are used frequently and require flexibility. *Implicit operand* instructions use a 7-bit opcode with a 3-bit modifier and implicitly use special registers R0, R1, and R2 as operand and destination registers. Special instructions like memory load/store and compression/decompression are used infrequently and can therefore use the *implicit operand* format for a small cost.

5.5 DMEM Compression for Standby Mode

While efficient instruction encoding helps minimize the footprint of IMEM, we use data compression to help minimize the footprint of DMEM. Along with fine-grained

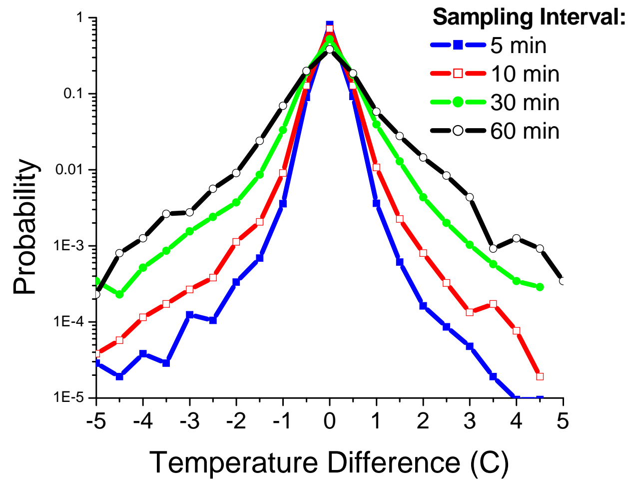


Figure 5.5: Distribution of temperature in Muskegon, MI in 2006 [75] represented as the difference between temporally adjacent measurements

power gating in DMEM (to be discussed in Section 5.6), compression permits fewer DMEM entries to be retained and enables significant power reductions in standby mode. Compression of instruction and data memories has been explored previously [67][68]. The IBM Memory Expansion Technology, for example, uses compression to more than double the size of main memory [67] but requires a complex memory management protocol targeted at server systems. To ensure that the energy overheads of compression do not surpass the standby mode reductions of a compressed DMEM, we adopt a simple compression architecture in the Phoenix Processor.

During compression, words from the 16-entry cache are sequentially converted to compressed words using a compression lookup table. The 512-byte virtual memory is divided into 16-byte blocks, and an entire 16-byte block from the cache must be compressed before being sent to the 266-byte physical memory.

The primary function of the Phoenix Processor is sensor data logging, which has two important consequences for compression. The first consequence is that access to memory is largely sequential (since temporally adjacent measurements are stored in spatially adjacent memory locations), thus limiting the compression/decompression overheads associated with random hopping among 16-byte blocks. The other important consequence concerns compression dictionary selection. Typical sensor data is predictably compressible since two temporally adjacent points are likely to differ by only a small amount. The measurement for a particular time can be stored as the difference between the current and previous measurements. The resulting data distribution is tightly distributed around zero, making dictionary selection simpler. Since the Phoenix Processor includes an on-board temperature sensor, we consider a collection of ambient temperature measurements in Michigan as an example [75]. Figure 5.5 shows the differences between temporally adjacent temperature measurements for a full year at different sampling intervals. In this difference format, 96% of the data falls in the range -1ºC to 1ºC assuming a 10 minute sampling interval. We take advantage of this small range by using a fixed compression dictionary that uses short words to represent values in this range and longer words to represent the rare value outside of this range.

We use Huffman encoding to generate a lookup table-based dictionary using temperature measurements from [75] assuming a temperature precision of 1ºC and a sampling interval of 30 minutes (which was empirically determined to efficiently compress data sampled at intervals ranging from 5-60 minutes). The lookup table converts 8-bit uncompressed data words to compressed words with lengths between 1 and 13 bits. By using a fixed dictionary, the compression operation is simplified significantly,

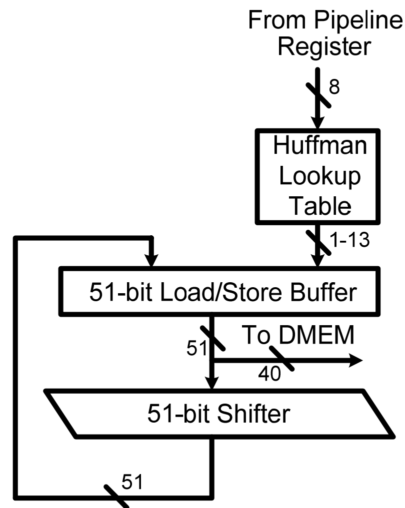


Figure 5.6: Hardware support for compression

minimizing the active energy penalty. While the footprint of compressed data can grow by up to 60% if measured data is not sufficiently similar to the distribution in [75], an editable fixed dictionary could potentially be stored in DMEM to better match the needs of a specific application.

After using the Huffman lookup table to compress an 8-bit data word, the compressed word is shifted by a 51-bit shifter and then stored in a 51-bit load/store buffer (Figure 5.6). Once all entries in a 16-byte block have been loaded in the load/store buffer or the buffer is full, the compressed data is sent to DMEM using one of the 3 load/store instructions supported by the ISA.

Memory allocation is the primary challenge in implementing compression. Fixed length uncompressed blocks from virtual memory are translated to variable length compressed blocks in physical memory, and efficient placement of the variable length blocks within physical memory can be difficult. To address this problem, we divide DMEM into the two partitions shown in Figure 5.7: a statically allocated partition and a

dynamically allocated partition. Each 16-byte block in virtual memory is assigned a 40-bit entry in statically allocated memory. Data is normally stored in the statically allocated partition. However, if a 16-byte block does not fit within its statically allocated entry after compression, the overflow data is stored to an entry in dynamically allocated memory and a 5-bit pointer to the overflow data is stored in the statically allocated entry. A free-list is required to monitor which entries in dynamically allocated memory are available for storage. A priority encoder in the free-list returns the address of the first available entry in the event of an overflow. For compression purposes, the free-list need only monitor the dynamically allocated partition, but we monitor both memory partitions to permit fine-grained power gating (to be discussed in Section 5.6). Including the overhead of the free-list, the Phoenix Processor compression scheme represents 16-byte blocks with a minimum of 41 bits (a compression ratio of 32%). The effectiveness of the proposed compression scheme will be quantified using test-chip measurements in Section 5.8.

5.6 Ultra-Low Standby Power Memory Design

The power consumed by IMEM and DMEM dominates standby mode power since data must be retained in standby mode. In contrast, the CPU and other non-retentive logic can be fully power gated. Minimizing standby power in the IMEM and DMEM is therefore a critical design requirement for the Phoenix Processor. The memories must also be designed for robust operation at low supply voltage to avoid the overhead of a dual supply voltage system.

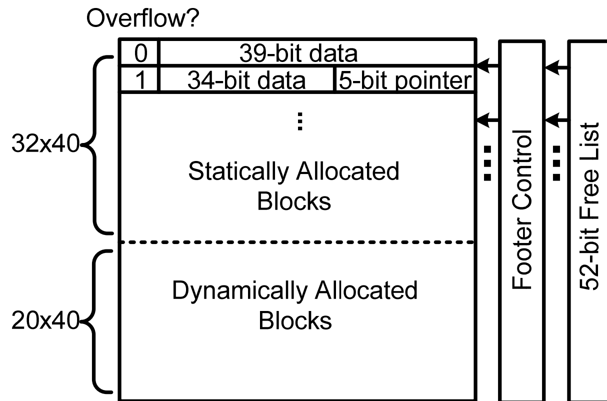


Figure 5.7: Memory support for compression

In the Phoenix Processor, the instruction memory is accessed every cycle by the CPU but does not need to be modified at runtime. Consequently, instruction memory is composed of a 64x10b SRAM (IMEM) and a 64x10b ROM (IROM). Commonly used procedures are stored in IROM while application-specific instructions are stored in IMEM. It is advantageous to put as many instructions in IROM as possible since ROM can be power gated during standby mode. In this work, we use the robust full static CMOS ROM implementation described in [69].

To minimize the standby leakage in retentive cells in IMEM and DMEM, we use the custom ultra-low standby power SRAM cell shown in Figure 5.8. The bitcell transistors (cross-coupled inverters and access transistors) use the high-V_{th} IO devices offered by the selected 0.18μm technology. Although the minimum dimensions of the IO device are larger than those of the thin oxide device, the large leakage reduction justifies the use of the device. We further reduce leakage in the bitcell using stack forcing in the cross-coupled inverters as in other retentive gates. We use a stack height of two because the sensitivity of leakage to stack height becomes linear for larger stacks, as shown in

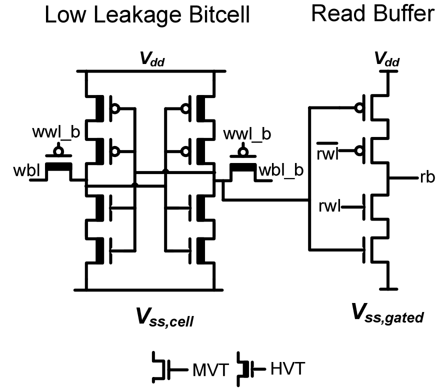


Figure 5.8: Proposed ultra-low standby power SRAM cell

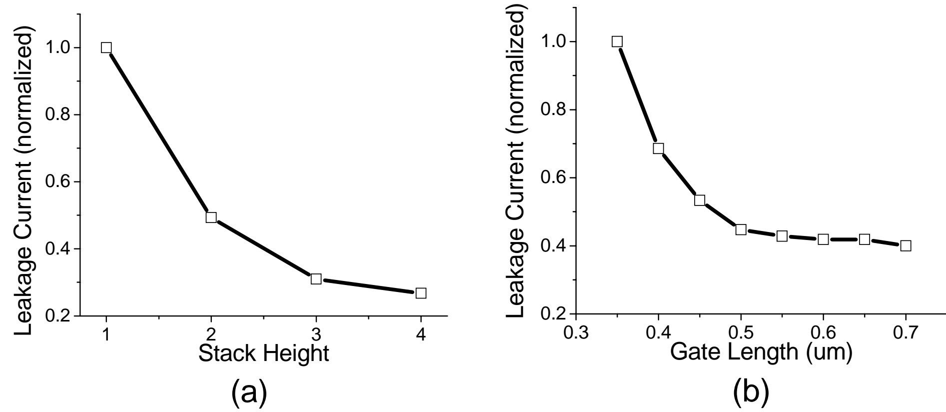


Figure 5.9: Effectiveness of (a) stack forcing and (b) gate length biasing for leakage reduction

Figure 5.9(a). Instead of further stack forcing, we find that increasing the length of the devices in the cross-coupled inverters gives a more area-efficient reduction in leakage. By increasing the length of the transistors from 0.35um to 0.50um, the leakage is reduced by ~2X, as shown in Figure 5.9(b). Stack forcing and gate length biasing are not applied to the access transistors to avoid upsetting write margins.

The proposed bitcell enables dramatic leakage reduction at the expense of active power and area. In the target 0.18μm technology, the area of the proposed cell is 40μm^2, which is 9.1X larger than the traditional 6T cell in [81]. Since the instruction and data

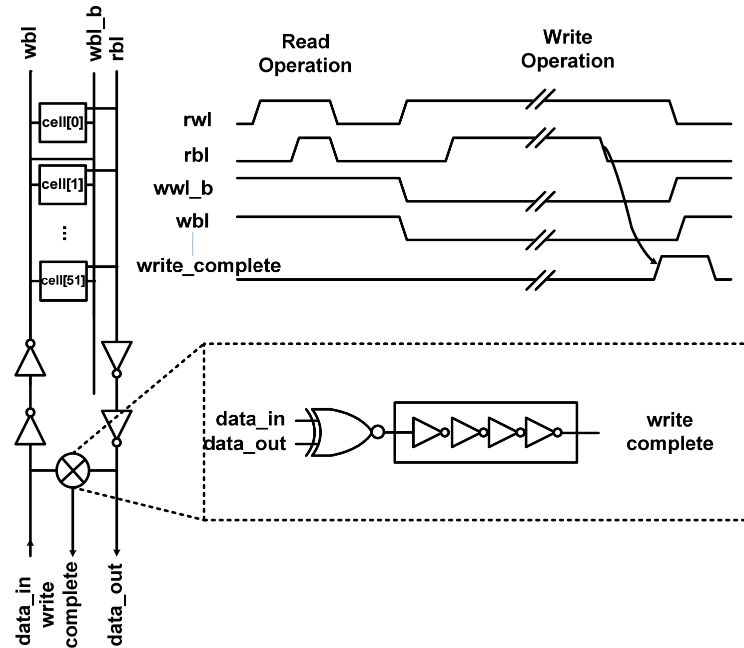


Figure 5.10: Memory column diagram showing completion detection

memory bitcells represent only a fraction of total chip area, the effective area penalty is only ~17%. Even with the larger bitcell, measurements show that the memories contribute only 8% of total active power while delivering a 62X reduction in memory standby power (which is 90% of total standby power) as compared to the 6T bitcell in [81]. Given the dominance of standby energy in a typical sensing application, the proposed bitcell delivers a favorable trade-off.

In addition to operating at low standby power, the proposed SRAM cell in Figure 5.8 includes a full swing 4-transistor read buffer for robust low voltage operation. Read buffers have been previously proposed to decouple read and write margins in low voltage SRAM cells [70][71]. The full swing read buffer drives the bitline to both supply rails, ensuring a robust read. Since the read buffer can be power gated in standby mode without upsetting the bitcell data, it is implemented using medium-V_{th} devices for a negligible

leakage penalty. The use of medium-V_{th} devices ensures a fast read time comparable to paths in the CPU (implemented with medium-V_{th} devices). This is particularly important for the IMEM, which is accessed every cycle and lies on critical timing paths.

While the read delay is comparable to the delay of the CPU, the write operation through the high-V_{th} devices is slow. We therefore adopt an asynchronous write strategy in which the CPU stalls for 2-3 cycles during the write operation. The DMEM asserts a completion signal to alert the CPU when the write operation is finished. As shown in Figure 5.10, the write completion signal is generated by reading the contents of the row being written and comparing to the write data. Since read is single-ended, a replica delays the write completion signal to guarantee that both sides of the cell have been written correctly.

To permit further leakage reduction within DMEM, power gating switches are used to eliminate the standby mode power consumption in non-retentive entries. A particular DMEM entry is power gated only if the free-list (described in Section 5.5) indicates that the entry is unused. Power gating granularity plays an important role in determining total power. Using a single power gating switch for each row allows the memory to grow to precisely match the footprint of the data. However, the width of a power gating switch is determined by the maximum current needed to read/write a single row, so the width of a single power switch changes minimally with power gating granularity. With one power switch allotted per DMEM row, the total leakage power is sub-optimal because the total power gate width is large. For example, the use of 52 switches for the 52 rows of DMEM would require a total power switch width approximately 52 times wider than the case where one footer is used for the entire

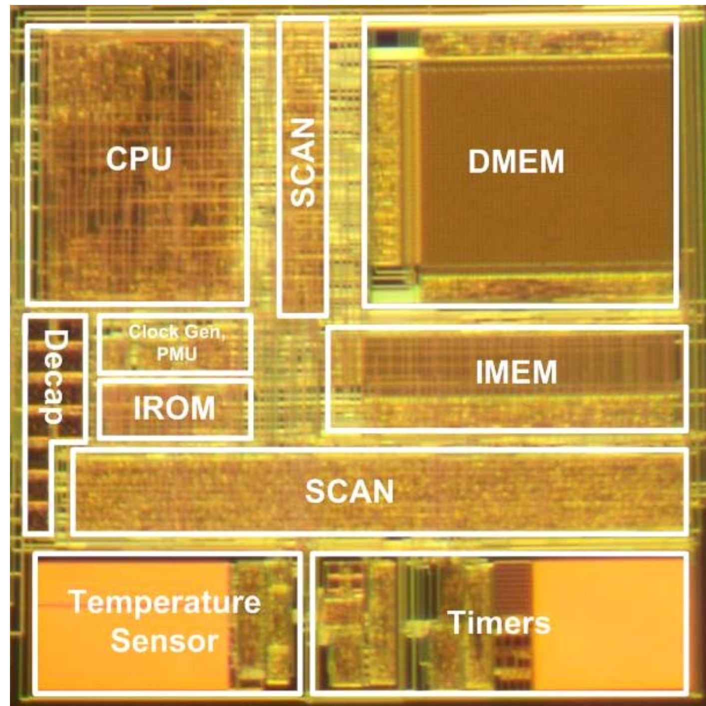


Figure 5.11: Phoenix Processor die photo

DMEM. Higher footer granularity also leads to higher complexity in free-list management and a commensurate increase in standby power. We find that minimum standby power is achieved in the Phoenix Processor when two DMEM rows are grouped with a single footer.

5.7 Test Chip Overview

To demonstrate our standby mode strategy, we fabricated the Phoenix Processor in a 0.18µm process (die photo shown in Figure 5.11). The processor includes 60,332 medium-V_{th} devices and 32,167 high-V_{th} devices in an area of 915x915µm^2. The memory, temperature sensor, and timer blocks were designed using a standard full-custom flow. The CPU and interfaces to memory, temperature sensor, and timer blocks were implemented using both synthesized and semi-custom blocks using a standard tool flow

and a library limited to minimum-sized gates with maximum fan-in of three. As in previous low voltage processors, we routed signal, clock, and power wires using minimum width interconnect to reduce switching energy and improve routing density [22].

5.8 Measured Results

5.8.1 Power and Performance Results

The measured frequency and energy consumed per clock cycle are shown in Figure 5.12 as functions of V_{dd} for one test application. The frequency is determined by sweeping the clock frequency, running the test application, and noting the frequency above which the contents of memory are corrupted. The test application runs a short iterative sequence that writes a known list of numbers to DMEM, in the process exercising all timing critical instructions. Power is measured during execution using a high precision ammeter. At the target voltage of 0.5V, the die highlighted in Figure 5.12 operates at 106kHz with only 2.8pJ consumed per cycle, which corresponds to only 297nW. Figure 5.13 and Figure 5.14 show distributions of maximum operating frequency (mean of 121kHz) and active power at 60kHz (mean of 226nW) for 13 dies at V_{dd}=0.5V. The consequences of variability are particularly important at low operating voltages and have been covered extensively in previous work [22][78].

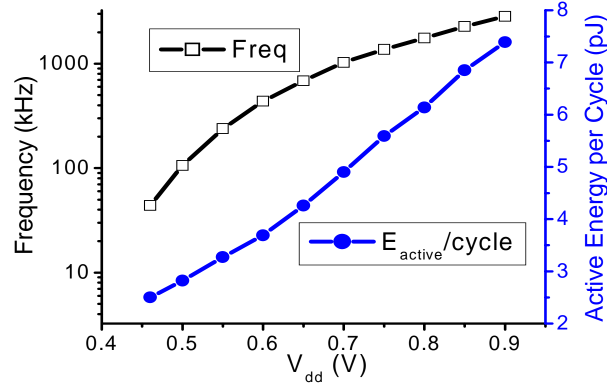


Figure 5.12: Measured frequency and energy consumption

Work	Technology	Architecture	Frequency	Active Energy	Standby Power
[80]	0.13µm	8-bit 8051 with custom peripherals	8 MHz	18.8 pJ/cycle	53.6 µW
[78]	65nm	16-bit MSP430	434 kHz	27.3 pJ/cycle	1 µW
[22]	0.13µm	Custom 8-bit	354 kHz	3.5 pJ/instruction	153 nW
[94]	0.25µm	Custom 8-bit	500 kHz	12 pJ/instruction (core only)	13-20 nW
This work[1]	0.18µm	Custom 8-bit	121 kHz	3.8 pJ/cycle (at 60 kHz)	35.4 pW

Table 5.2: Comparison to other low voltage technologies ([1]mean values across 13 measured dies are presented for this work)

Figure 5.15 shows that the mean standby mode power consumption for the same 13 dies at V_{dd}=0.5V is 35.4pW with 50% of DMEM entries retained. The IMEM and DMEM consume 89% of standby power while the power gated CPU consumes only 7% of the power. For a typical sensing application in which the sensor remains active for 1000 cycles every 10 minutes, these measurements indicate that the average power consumption is only 42pW.

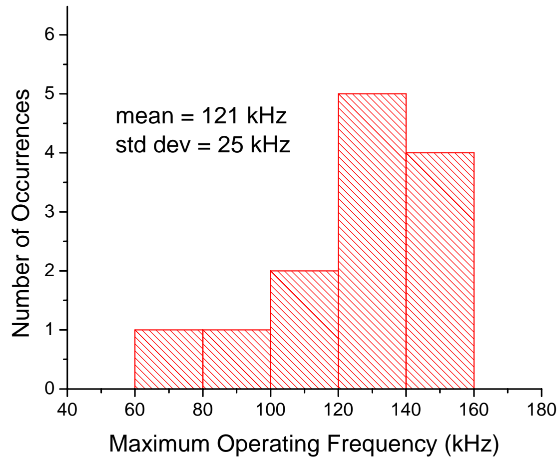


Figure 5.13: Measured frequency distribution for 13 dies at V_{dd}=0.5V

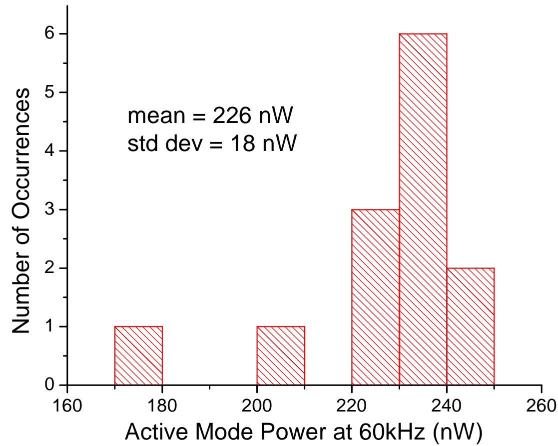


Figure 5.14: Measured active mode power distribution at 60 kHz for 13 dies at V_{dd}=0.5V

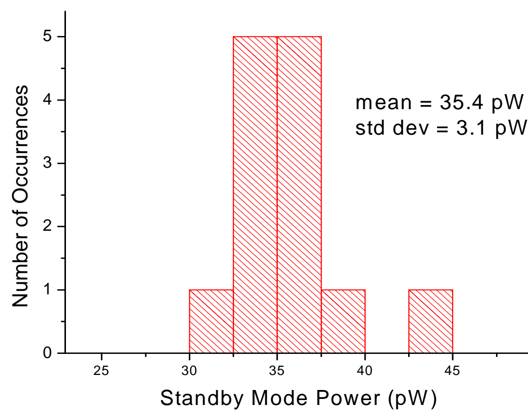


Figure 5.15: Measured standby mode power distribution for 13 dies at V_{dd}=0.5V

The reported standby power is lower than that reported in any previous work. The standby power consumption of the Phoenix Processor is compared to previous ultra-low power microprocessors in Table 5.2. It should be noted that each microprocessor listed was implemented using a different architecture and specifications, so direct comparison can be difficult.

5.8.2 Power Gating Results

To further investigate our proposed power gating approach, we sweep the footer width on the CPU and measure the energy and performance implications. Figure 5.16(a) shows the maximum operating frequency of the CPU as a function of footer width. Frequency reduces by 5X as the footer size approaches the minimum of W=0.66µm and L=0.5µm. This performance penalty leads to greater active energy consumption per operation since leakage energy increases with clock period in active mode. The power consumption through the power gating switch results in an additional energy penalty. However, the standby leakage power savings from the narrow footer width, shown in Figure 5.16(b), easily offsets these penalties and reduces the total energy for the Phoenix Processor. Figure 5.17 confirms that the total energy consumption is 3.8X lower for the small footer (W=0.66µm) than the large footer (W=28µm) assuming 1000 instructions are executed every 10 minutes. The small footer saves several orders of magnitudes of total energy compared to a design with no power switch.

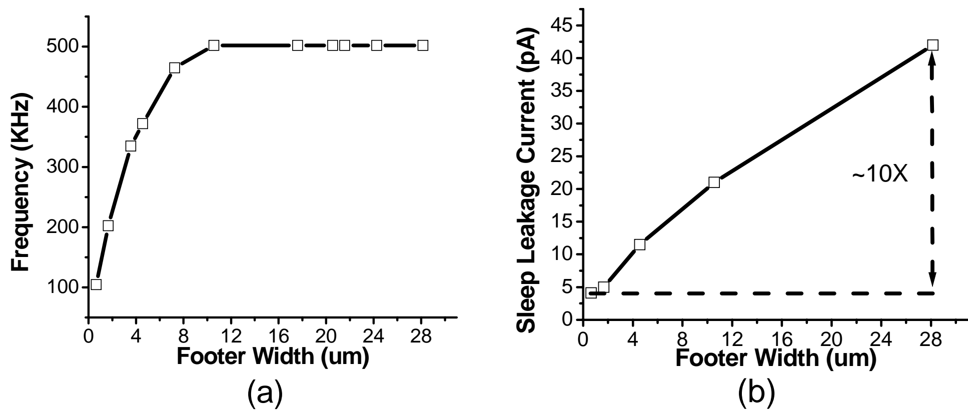


Figure 5.16: Measured (a) frequency and (b) standby leakage as functions of CPU footer width

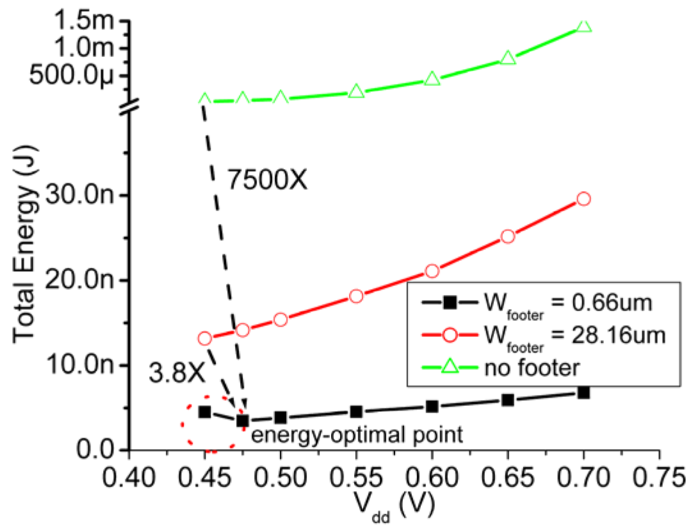


Figure 5.17: Total energy consumption assuming 1000 instructions are executed every 10 minutes

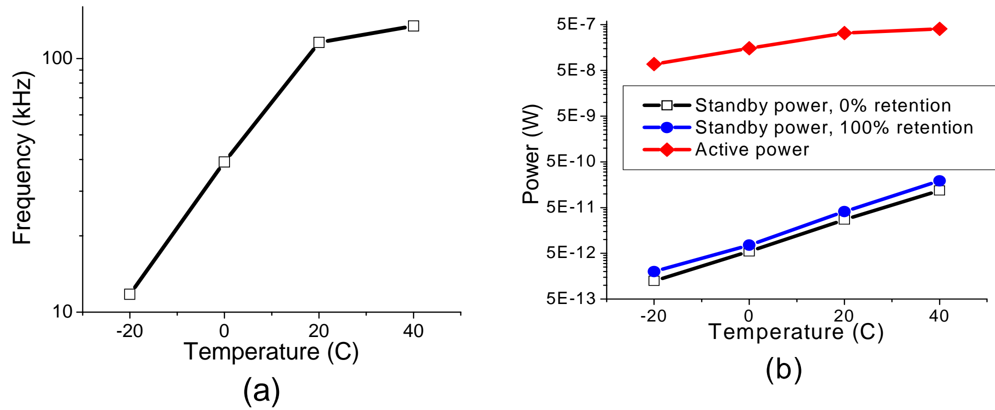


Figure 5.18: Measured (a) frequency and (b) power as functions of temperature

5.8.3 Memory Results

The IMEM and DMEM consume 7.1fW/bitcell (not including the overhead of decoders and row drivers). The IMEM alone accounts for 39% of the total standby power (including the overhead of decoders and drivers), which underscores the importance of instruction set optimization. If the instruction width had been set to 15 bits instead of 10 bits, measurements of a typical die show that the standby power of IMEM would have increased by 20%, which equates to ~8% increase in total standby power.

Unlike IMEM, the power consumed by DMEM, which amounts to 51% of total standby power at 50% retention, can be reduced significantly by compressing the data and by changing the number of DMEM entries retained to match the footprint of compressed memory. For a typical die, the DMEM consumes 22pW with all entries retained and 7.5pW with all entries power gated due to the overhead of maintaining the free-list (i.e., the overhead of data compression).

To quantify the system-level benefits of compression and fine-grained power gating in memories, consider an ambient temperature sensing application in which the Phoenix Processor wakes up once every 10 minutes and runs a 1000 cycle routine to

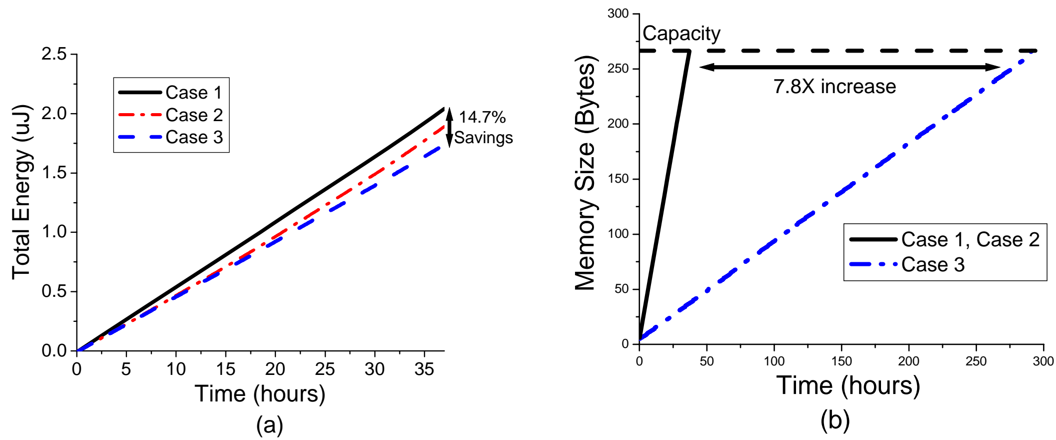


Figure 5.19: Computed time profiles of (a) energy and (b) memory size for a temperature measurement routine

measure temperature and store the measured data. Using the measured temperature dependence of frequency and power consumption shown in Figure 5.18 and a subset of the temperature profile from [75], we can compute the reduction in energy due to compression and power gating in DMEM over the lifetime of the chip. For this case study we assume that temperature is measured to a precision of 1ºC. Figure 5.19(a) shows the total energy consumed over 37 hours (the time period over which uncompressed memory fills to capacity) for 3 cases. In Case 1, neither compression nor power gating are used in DMEM. In Case 2, power gating is used, but compression is not used. Finally, in Case 3, both power gating and compression are used. The use of power gating within DMEM (Case 2) reduces energy consumption by 7.3%, and the use of both power gating and compression (Case 3) reduces energy consumption by 14.7%. Compression also increases the effective size of DMEM and enables the processor to remain active for 7.8X longer before memory fills to capacity, as shown in Figure 5.19(b).

5.9 Conclusion

Standby power was shown in this chapter to be extremely important for many $1mm^3$ computing applications. To address the problem posed by standby power, we developed the Phoenix Processor, which leverages low standby power techniques at the device, circuit, and architecture levels. Measurements show that Phoenix consumes 226nW in active mode and only 35.4pW in standby mode. Given a 10 minute standby period between active periods of ~100ms (as in the example at the beginning of this chapter) the Phoenix Processor draws ~73pW on average over its lifetime. With such low power consumption, the 1.55V thin film battery described in Chapter 1 [15] would last for more than 2 years with an area of only $1mm^2$ (assuming 100% power efficiency in DC-DC conversion). Though the techniques described in this chapter bring the power consumption of digital components within the realm of $1mm^3$ computing, the power consumption of sensors, actuators, and radios is still a major barrier. This problem is considered in the next chapter with the design of a low voltage CMOS image sensor.

Chapter 6

An Ultra-Low Voltage CMOS Image Sensor

The first five chapters of this work focus on the design of robust low voltage microprocessors, critical elements in any 1mm^3 system. However, the microprocessor is only one of several components limited by power constraints. Radios and many types of sensors and actuators consume considerable power and, in many cases, dominate the power budget of a wireless system. In light of this, the focus in this chapter is shifted to the design of ultra-low power sensors. Though many sensors are MEMS-based, conventional CMOS processes offer the ability to sense a number of quantities including temperature and light. This chapter focuses specifically on the design of an ultra-low power CMOS image sensor. Such a sensor will play a critical role in emerging wireless applications that demand video and still image capture; applications ranging from untethered surveillance motes to artificial retinas.

As shown in previous chapters, low voltage operation is becoming an increasingly attractive option for low power digital logic and memory. This chapter explores the application of low voltage techniques to a CMOS image sensor. In addition to reducing energy within the image sensor, robust low voltage operation will ensure compatibility with on-chip low voltage logic in 1mm^3 computing systems. The typical approach to CMOS image sensor design is described in Section 6.1, and in Section 6.2, an ultra-low power CMOS image sensor optimized for highly energy constrained applications is

proposed. The proposed pixel architecture and array architecture enable operation at a near-threshold supply voltage of 0.6V and below. Measurements of a 0.13µm test-chip at 0.6V indicate that a 128x128 image sensor array consumes only 1.9µW at V_{dd}=0.6V and 3.3fps with a signal-to-noise ratio (SNR) of 24.8dB. While low voltage operation tends to reduce energy at the expense of increased noise, such a trade-off may be attractive in many applications requiring simple image analysis.

6.1 Typical CMOS Image Sensor Operation

Pixel readout in an image sensor is typically accomplished using the 3T active pixel sensor (APS) design shown in Figure 6.1. The photodiode node, N_{PD}, is first pre-charged using M1 and is subsequently discharged by photocurrent for an extended period (called the integration time). After this extended period, the voltage at N_{PD} must be read out and converted to a digital number. Transistor M2 and a shared current source act as a source follower that copies the voltage at N_{PD} onto the bitline. The bitline voltage is then read using a conventional analog-to-digital converter (for example, a sigma-delta converter [87]).

The 3T APS structure has proven to be robust at high voltage but relies on the buffering of an analog voltage to a noisy bitline that is shared by many leaking devices. At low voltage, the cumulative bitline leakage of inactive pixels begins to approach the read current of the active pixel (in a manner reminiscent of SRAM [93]), making it very difficult to distinguish between many analog values. This problem is exacerbated by the high current variability observed at low voltage. Our solution to this problem is described in the next section.

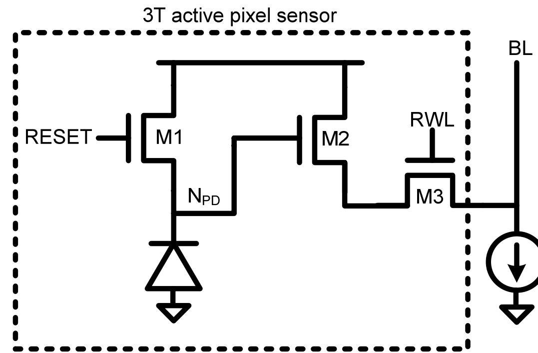


Figure 6.1: A conventional 3T active pixel sensor design

6.2 A Low Voltage CMOS Image Sensor

To address the aforementioned problems confronting low voltage 3T APS structures, we use an in-pixel comparator to convert the photodiode voltage into a digital pulse with a width linearly related to the photodiode voltage. In doing so, we drive the bitline with only digital values, thus avoiding the aforementioned problems associated with low I_{on}/I_{off} and high current variability. Similar approaches have been proposed in the past [83][84], with recent work [85] showing that high SNR can be maintained at supply voltages as low as 1.35V with such techniques.

The column architecture and a timing diagram are shown in Figure 6.2. Before integration, the photodiode voltage and bitline are pre-charged to V_{DD} (Step 1). After integrating the photocurrent in the pixel of interest (Step 2), the photodiode voltage, $V_{photodiode}$, is compared to a voltage ramp using an in-pixel comparator. When the ramp voltage drops below $V_{photodiode}$, the bitline is discharged (Step 3). The time at which the bitline switches is a linear function of $V_{photodiode}$ and is measured using a counter (Step 4). The use of a digital comparator avoids the functionality problems faced by the source follower in the traditional APS pixel. Furthermore, the effects of variation are minimized since V_{th} variations simply shift the switching threshold of the comparator, a problem

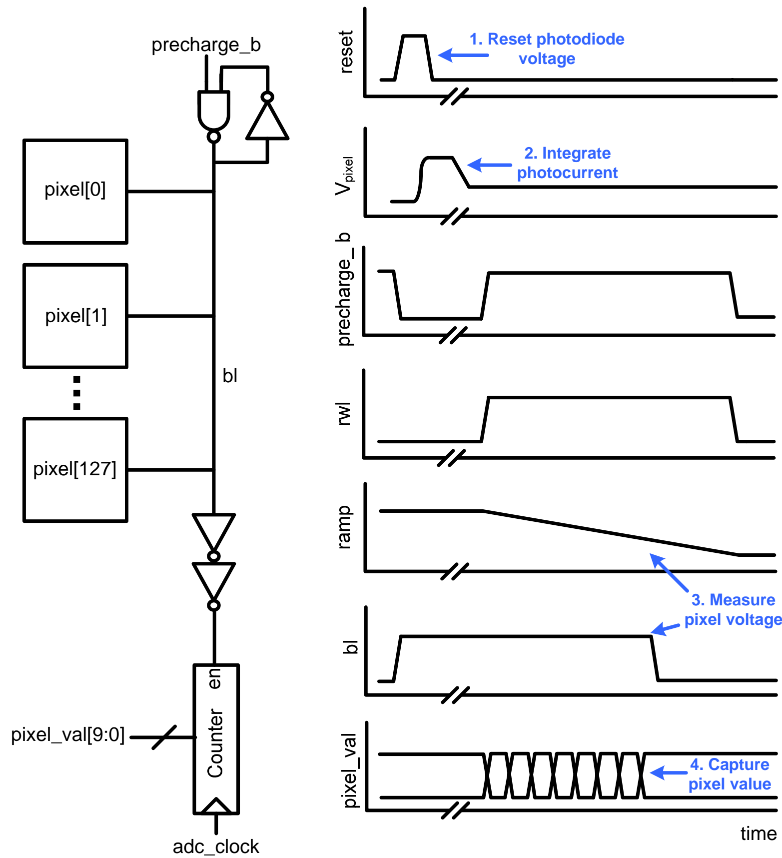


Figure 6.2: Column architecture and timing diagram for a read operation

easily corrected with delta reset sampling [86]. Delta reset sampling is implemented on chip with little overhead by using the implicit addition operation performed by counters (i.e., count down during the measurement phase and up during the offset cancellation phase).

Figure 6.3 shows the pixel, which contains three components: a pixel control block, a comparator, and a read buffer. Inside the pixel control block, the photodiode node, N_{PD}, is reset before an integration period using device M2. Device M1 sets the voltage at N_{PD} below V_{DD} to ensure a linear response from other components. A 2T comparator structure (M3, M4) is used to leverage unique device sensitivities at low voltage. Since subthreshold devices act as excellent current sources, the currents through

devices M3 and M4 are exponential functions of their respective gate voltages only. Consequently, $V_{photodiode}$ and the voltage at *ramp* can be compared simply by looking at the currents through M3 and M4. Initially, the current through M4 is greater than the current through M3, and the comparator output is held low. As *ramp* reduces, the current through M3 becomes larger and eventually exceeds the current through M4, which switches the comparator output high. As shown by the voltage transfer characteristics in Figure 6.3, the exponential dependence of current on gate-source voltage gives high gain and a fast transition at the comparator output. The gain is further improved using the current limiters shown in Figure 6.3, which are shared by all pixels in a column. The increased gain provided by the current limiters help minimize short circuit current that occurs during evaluation, though this current is not completely eliminated as will be shown when discussing test-chip measurements. A 2T read buffer (M5, M6) also helps to drive the bitline strongly and overrides leakage due to un-accessed pixels on the bitline.

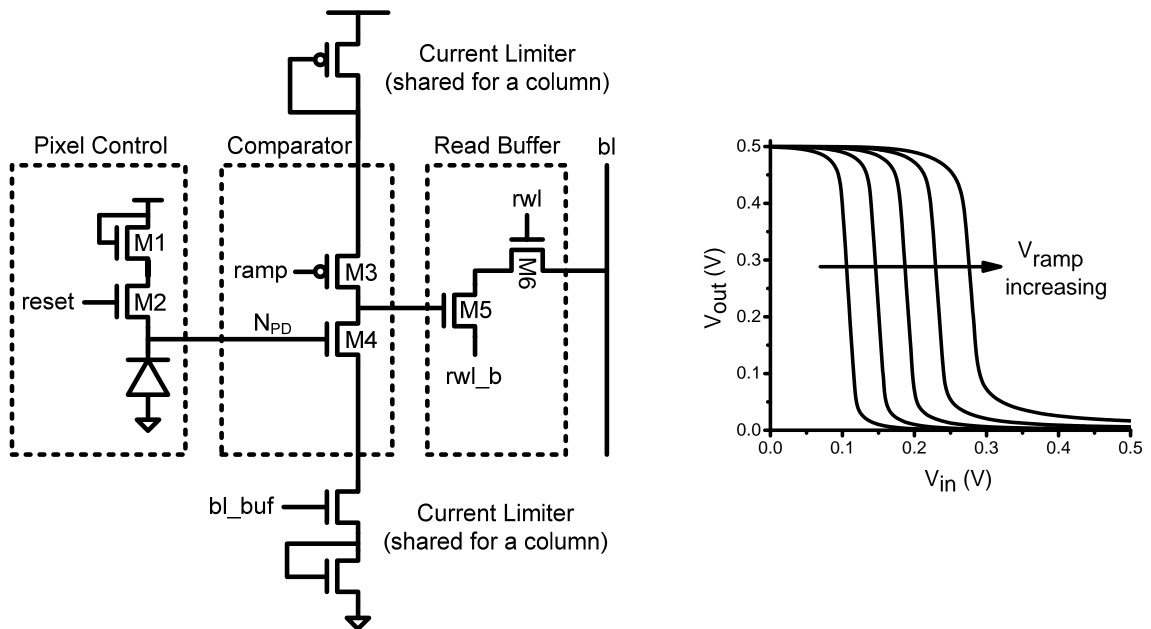


Figure 6.3: Low voltage pixel architecture and comparator voltage transfer characteristic

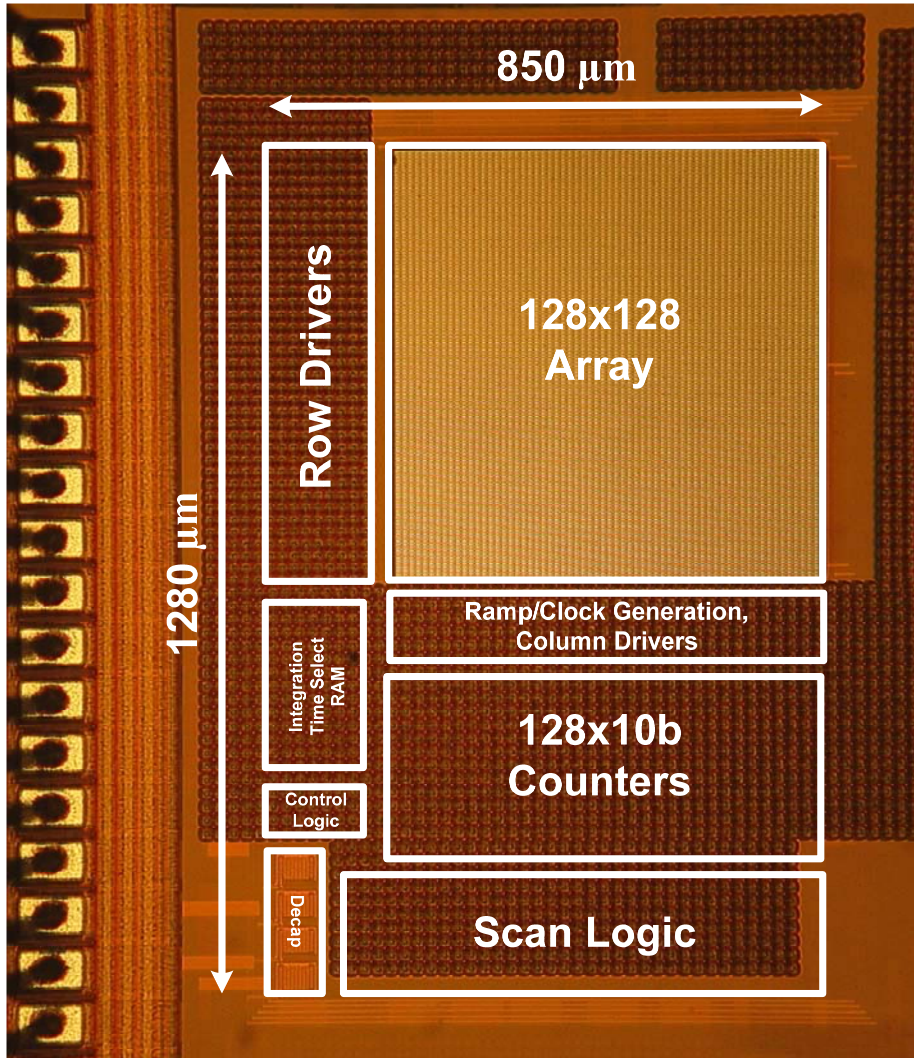


Figure 6.4: Die photo

6.4 Test-Chip Overview

We have fabricated a 128x128 low voltage image sensor in a 0.13µm bulk logic technology to demonstrate our proposed techniques (die photo shown in Figure 6.4). In the target technology, V_{th}~400mV at V_{ds}=50mV. Each 5x5µm pixel contains a 7.9µm^2 n-diffusion/p-substrate photodiode with both polyimide and silicide layers removed. Increased gate widths and lengths are used throughout the pixel to reduce V_{th} and

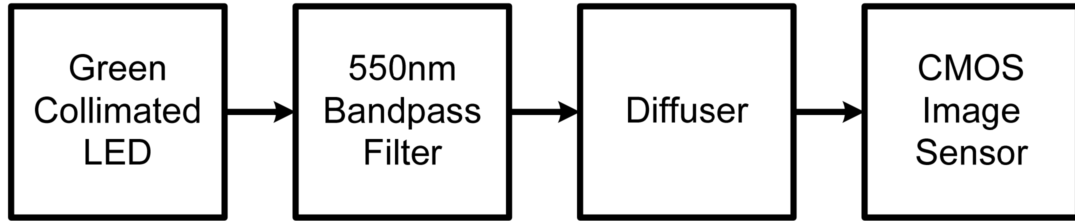


Figure 6.5: Test setup

subthreshold swing variations. Internal generators for *ramp* and *adc_clock* signals are included on chip though are bypassed during testing to enable fine-grained tuning.

To characterize the image sensor, we use the test setup shown in Fbigure 6.5. A green collimated LED is projected on a 550nm bandpass filter to reduce wavelength uncertainty to only +/-10nm. The filtered light is passed through a diffusing lens that scatters the light in all directions. Finally, this scattered light uniformly illuminates the CMOS image sensor. All data described in the remainder of this section are captured under an irradiance of 35mW/m^2. Rather than sweeping this light intensity to characterize the sensor, we sweep integration time, which has the effect of varying the number of incident photons.

6.5 Mean Responsivity

Figure 6.6 shows the mean pixel value over the entire array as a function of the number of incident photons (i.e., incident light). Each pixel value has been averaged over 100 frames to eliminate temporal variations, and delta reset sampling has not been performed. Ideally, the pixel value should increase linearly as the number of incident photons increases. The image sensor gives monotonic output, though the observed non-linearity near saturation affects the fidelity of a captured image and is undesirable. Much

of this non-linearity results from innate non-linearity in the proposed pixel and could be eliminated with an alternative pixel structure.

Dark current (i.e., subthreshold leakage and junction leakage at the photodiode) is also an important quantity since it limits dynamic range and introduces noise. Figure 6.7 shows the measured pixel value as a function of integration time under dark conditions. Note that contribution of dark current as measured by pixel value is comparable to that of the incident light in Figure 6.6 (which was captured using the same range of integration times as Figure 6.7). Such high dark current is a threat to the noise characteristics, a topic considered extensively in the next sub-section.

6.6 The Effects of Variability

As previous chapters showed, the increased sensitivity to variability is a serious concern in low voltage circuits. In an image sensor, random process variability in the pixel and in analog-to-digital conversion hardware results in fixed pattern noise (FPN), which is spatial variation in measured pixel value under uniform illumination. A portion of FPN can normally be eliminated using correlated double sampling or delta reset sampling [86]. To measure FPN without delta reset sampling, we first capture an image in which temporal variations are eliminated by averaging over 100 frames. We then estimate FPN by measuring the standard deviation of the pixel values. Under dark conditions, FPN is 5.1% of the maximum pixel value. CMOS image sensors operating at nominal voltage may have FPN that is <10% of this value [85], which clearly underscores the need for more effective control of variability at low voltage.

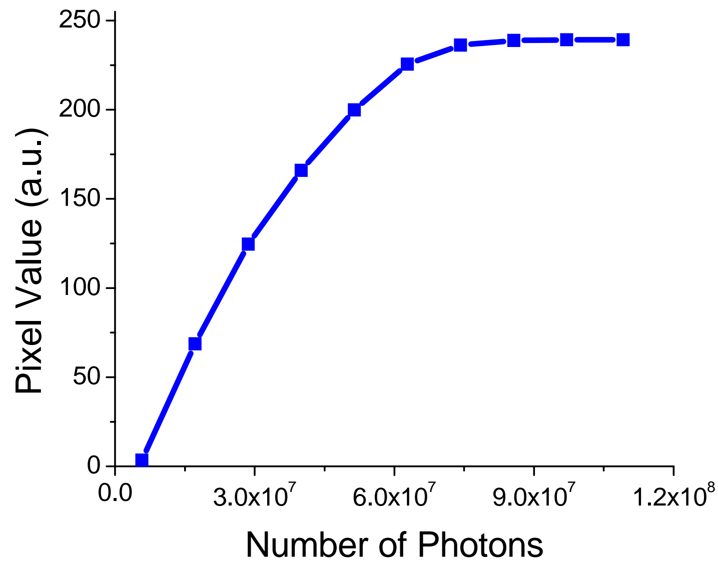


Figure 6.6: Mean responsivity over 100 frames

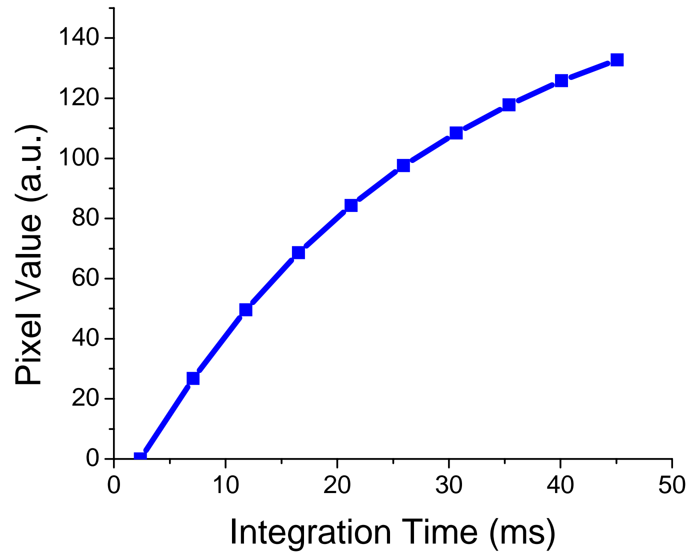


Figure 6.7: Mean pixel value as a function of integration time over 100 frames

The signal-to-noise ratio (SNR) measures the combined effects of FPN and temporal variations (which are caused by a combination of voltage fluctuations, switching noise, and light non-uniformities). To measure SNR, we normalize the mean pixel value at each location (over 100 frames) to the standard deviation in pixel value (over the same 100 frames) and convert to decibel (dB) representation. Figure 6.8 shows the SNR at V_{dd}=0.6V as a function of the incident light. As the number of incident photons increases, the SNR approaches its maximum value of 24.8dB (a ratio of ~17:1). As shown in Figure 6.9, the maximum SNR reduces with V_{dd}, going as low as 22.7dB (a ratio of ~14:1) at V_{dd}=0.5V. The observed trends in SNR again underscore the need for more effective variability control at low voltage.

6.7 Power Consumption and the Role of Voltage Scaling

Thus far, we have focused on the non-ideality introduced by low voltage operation. However, voltage scaling does offer dramatically reduced power consumption, as shown in Figure 6.10 at 3.3fps and V_{dd}=0.6V. Due to limitations at the computer-PCB interface, frame-rates above 3.3fps are not possible. During power measurement, external ramp signals and clock signals are used to drive the image sensor, but internal ramp and clock generators are run in parallel to simulate the effect of these blocks on total power. Power varies between 1.9µW and 3.4µW for different levels of incident light. This non-intuitive behavior is a result of short circuit current consumed in the 2T in-pixel comparator, which is dependent on the number of incident photons.

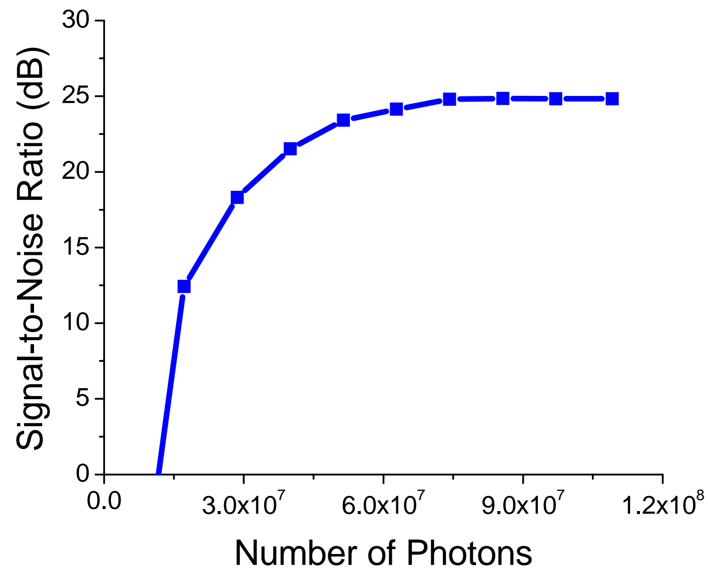


Figure 6.8: Signal-to-noise ratio as a function of incident light

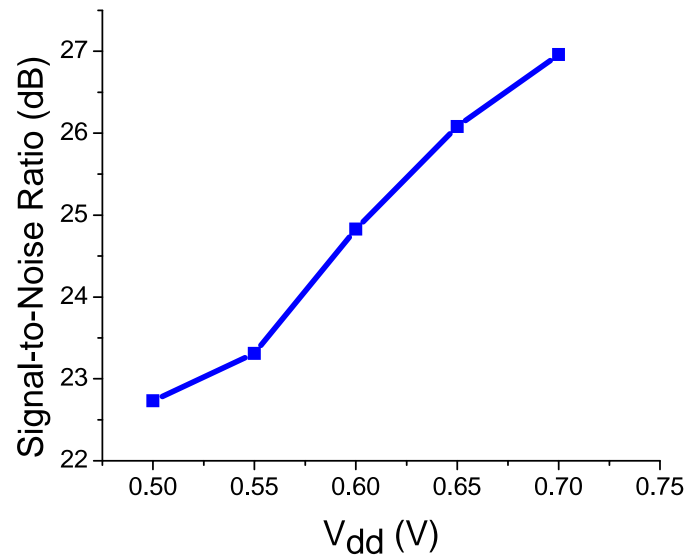


Figure 6.9: Peak signal-to-noise ratio as a function of V_{dd}

At V_{dd}=0.6V, the image sensor consumes 564nJ/frame. This number reduces to 188nJ/frame at V_{dd}=0.5V. The proposed image sensor is compared to previous work in Table 6.1. The proposed design compares favorably to the low power image sensor in [85], which consumes 5.8μJ/frame at 9.6fps after scaling to account for different array sizes. The image sensor proposed in [87] consumes only 480nJ/frame at 30fps, which is comparable to the value reported in this work. However, the measurements quoted in [87] do not capture the power overhead of a decimation filter, a critical element to the chosen architecture.

Note that the energy/frame metric is meant to be insensitive to frame-rate. However, in our image sensor test-chip, the leakage current during standby (i.e., a frame-rate of zero) is equivalent to 20-35% of the power at 3.3fps. At low frame-rates, this leakage results in a significant power overhead. This overhead would be amortized over multiple frames at higher frame rates, suggesting that our reported energy/frame metric could improve with a test setup that could support frame-rates greater than 3.3fps.

Figure 6.11 shows the V_{dd} sensitivity of power in the image sensor. Power consumption drops as low as 620nW at V_{dd}=0.5V and exhibits a very high sensitivity to V_{dd}, well in excess of the expected quadratic (V_{dd}^2) dependence. Short circuit current in the 2T comparator is again the culprit. The gate source voltage on the NFET device in the comparator increases with V_{dd} giving a near-exponential increase in short circuit current.

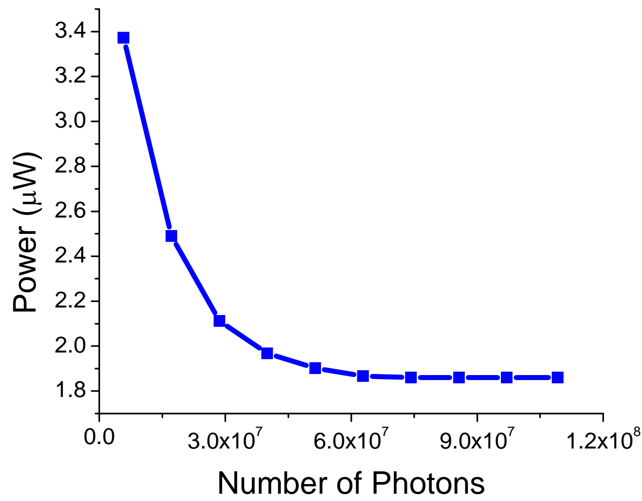


Figure 6.10: Power at V_{dd}=0.6V as a function of incident light

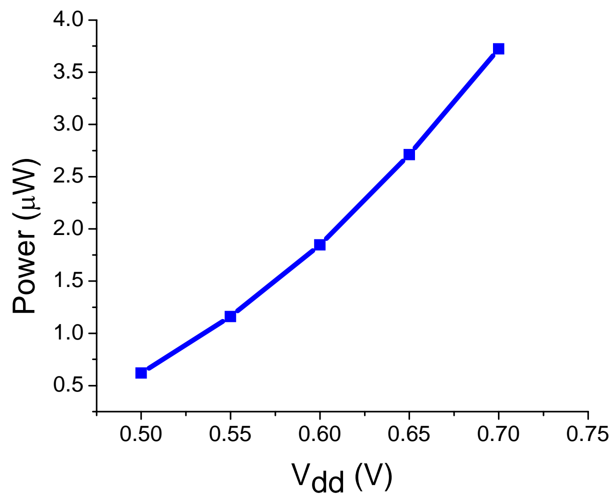


Figure 6.11: Power at maximum SNR as a function of V_{dd}

Image Sensor	V_{dd} (V)	Frame Rate (fps)	Energy/Frame (nJ/frame)	Comment
[85]	1.35	9.6	5800	Data rescaled for 128x128 array size
[87]	3.3	30	480	Does not include power of decimation filter
This work	0.5	3.3	188	

Table 6.1: Comparison to previous work

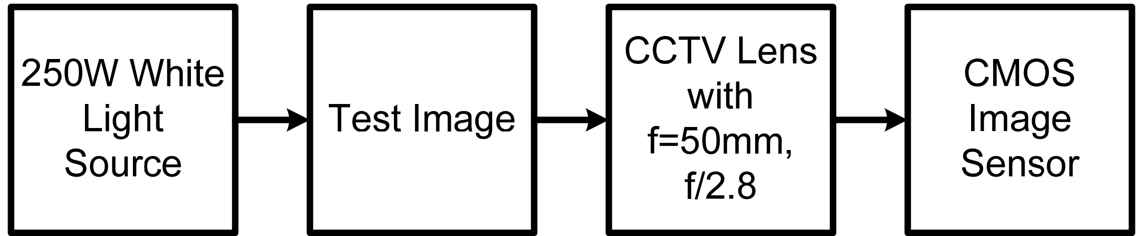


Figure 6.12: Image capture test setup

6.8 Test Image Capture

The previous sections showed that extremely low power operation is possible at low voltage at the expense of a reduced SNR. In this section, we explore the practical consequences of this added noise in an actual test image. To capture a test image, we use the test setup shown in Figure 6.12. A 250W white light source is used to illuminate a test image. The reflected light is focused by a CCTV lens with a focal length of 50mm and an f-number of f/2.8. The focused light is then projected on the image sensor.

As a demonstration, we use the 8-bit test image shown in Figure 6.13(a), which has 256x256 pixels. Since our pixel array contains 128x128 pixels, we show a down-sampled 128x128 version of the image in Figure 6.13(b) for reference. The image captured by our test-chip is shown in Figure 6.13(c). The image in Figure 6.13(c) clearly shows many of the finer details in the original image. However, the relatively low SNR observed at low voltage is clear from the high spatial noise in Figure 6.13(c).

6.9 Conclusion

The test chip measurements in this chapter clearly demonstrate that extremely low power operation can be achieved by applying low voltage techniques to a CMOS image sensor. However, this power reduction comes at the cost of increased noise. While the noise characteristics of the proposed image sensor may be insufficient for many imaging

Figure 6.13: (a) Actual 256x256 8-bit image (b) Image downsampled to 128x128 (c) Image captured by CMOS image sensor

applications, the test image in Figure 6.13(c) shows that simple object recognition may be possible for less demanding applications. A more detailed exploration of low voltage pixel architectures is merited and should be considered for any future work in low power image sensors.

Chapter 7

Conclusion

Cubic-millimeter computing is quickly becoming a reality, though power consumption remains as one of the critical barriers to further progress. This work focused on power reduction in digital components and placed specific emphasis on the use of low voltage operation to achieve this goal. Chapter 2 focused on the basic characteristics of subthreshold circuits and Chapter 3 examined the evolution of these characteristics as transistors scale. These concepts were explored further within the context of two low voltage processor test-chips in Chapters 4 and 5. The Phoenix Processor, in particular, demonstrated that low voltage operation combined with an aggressive standby mode strategy enables dramatic power reductions and brings on-chip power sources within the realm of possibility. Finally, conventional digital circuits were left behind in Chapter 6, and the focus was instead shifted to applying low voltage circuit techniques to an ultra-low power CMOS image sensor.

The work presented in this thesis has demonstrated the viability of low voltage operation. However, a great deal of work remains before we see widespread adoption of low voltage design techniques. In particular, the challenges presented by variability, which were discussed thoroughly in Chapters 2, 4, and 6, have not yet been adequately addressed. Despite the solutions proposed in this work (in particular, those of Chapter 4),

variability is still a serious threat to functionality and energy efficiency. Furthermore, as designers consider low voltage operation for performance-constrained design, delay variability will become a serious problem. While variability presents a significant problem, it is one that can be solved. The solution will require novel transistor geometries, careful circuit design, and adaptive architectures that can detect and correct for variability. A great deal of innovation in adaptive architectures, in particular, is already underway [88][89].

Though the power consumption of digital components was the primary focus of this work, the power consumption in sensor and radio components must also be reduced to make 1mm^3 computing a reality. MEMS-based sensing is continually offering new low power solutions, and research in ultra-wideband [91] and ultra-low power wake-up receivers [92], among other topics, offers hope for low power radios. In parallel with the development of new radios, it will also be important to re-design system architectures so that radios are required to communicate less frequently.

Improvements to batteries and energy scavenging devices will likely ease some of the pressure to design low power components but will also create new circuit problems. For example, the efficient conversion from high voltage (1.5V or higher) to subthreshold voltages is a complicated problem that has only recently begun to receive attention [59][78]. Hybrid power sources containing both batteries and energy scavenging devices also create an interesting set of problems for circuit designers including efficient battery re-charging using scavenged energy.

In addition to the component-level challenges mentioned above, the integration of multiple low power components in a single 1mm^3 package continues to be a challenge.

This is challenging not only because of the innate difficulties presented by millimeter-scale packaging but also because of logistics. The assembly of a $1mm^3$ computer requires expertise in the design of digital circuits, radios, sensors, packaging, and power sources in addition to any application-specific knowledge that may be relevant (e.g., knowledge about biocompatibility). To assemble such a diverse team can be difficult, particularly within academia, where the integration of multiple components is often deemed unworthy of research. However, a number of universities have already begun to show significant demonstrations of highly integrated systems [4][6][90]. These systems are the precursors to highly evolved cubic-millimeter systems that will undoubtedly solve a wide range of problems in the coming years.

References

[1] “eKo Pro Series System for Environmental Monitoring.” Crossbow Technology. 3 Dec. 2008. <http://www.xbow.com/Eko/index.aspx>.

[2] “MPXY8300: Microcontroller, Pressure Sensor, X-Z Accelerometer and RF Transmitter.” Freescale Semiconductor. 3 Dec. 2008. <http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MPXY8300&webpageId=M0yrt9vvk&nodeId=0112699vvk&fromPage=tax>.

[3] “EndoSure® Wireless AAA Pressure Sensor.” CardioMEMS. 3 Dec. 2008. <http://www.cardiomems.com/content.asp?display=patient+pb&view=endo%20sure>.

[4] K. Wise, A. Sodagar, Y. Yao, M. Gulari, G. Perlin, K. Najafi, “Microelectrodes, Microelectronics, and Implantable Neural Microsystems,” *Proceedings of the IEEE* ***96*****,** No. 7, pp. 1184-1202, 2008.

[5] R. Harrison, “The Design of Integrated Circuits to Observe Brain Activity,” *Proceedings of the IEEE* ***96***, No. 7, pp. 1203-1216, 2008.

[6] E. Zellers, S. Reidy, R. Veeneman, R. Gordenker, W. Steinecker, G. Lambertus, H. Kim, J. Potkay, M. Rowe, Q. Zhong, C. Avery, H. Chan, R. Sacks, K. Najafi, K. Wise, “An Integrated Micro-Analytical System for Complex Vapor Mixtures,” *International Conference on Solid-State Sensors, Actuators and Microsystems*, pp. 1491-1496, 2007.

[7] H. Quigley, A. Broman, “The Number of People with Glaucoma Worldwide in 2010 and 2020,” *British Journal of Ophthalmology****90***, pp. 262-267, 2006.

[8] B. Warneke, M. Last, B. Liebowitz, K. Pister, “Smart Dust: Communicating with a Cubic-Millimeter Computer,” *Computer* **34,** No. 1, pp. 44-51, 2001.

[9] I. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, “Wireless Sensor Networks: A Survey,” *Computer Networks****38***, No. 4, pp. 393-422, 2002.

[10] J. Bates, N. Dudney, B. Neudecker, A. Ueda, C. Evans, “Thin-Film Lithium and Lithium-Ion Batteries,” *Solid State Ionics* ***135***, No. 1-4, 33-45, 2000.

[11] H. Kulah, K. Najafi, “An Electromagnetic Micro Power Generator for Low-Frequency Environmental Vibrations,” *International Conference on Micro Electro Mechanical Systems*, pp. 237-240, 2004.

[12] “MSP430 Ultra-Low Power Microcontrollers.” Texas Instruments. 3 Dec. 2008. <http://focus.ti.com/mcu/docs/mcuprodoverview.tsp?sectionId=95&tabId=140&familyId=342>.

[13] G. Ono, T. Nakagawa, R. Fujiwara, T. Norimatsu, T. Terada, M. Miyazaki, K. Suzuki, K. Yano, Y. Ogata, A. Maeki, S. Kobayashi, N. Koshizuka, K. Sakamura, “1-cc Computer: Cross-Layer Integration with 3.4-nW/bps Link and 22-cm Locationing,” *Symposium on VLSI Circuits*, pp. 90-91, 2007.

[14] U. Schnakenberg, P. Walter, G. vom Bogel, C. Kruger, H.C. Ludtke-Handjery, H.A. Richter, W. Specht, P. Ruokonen, W. Mokwa, “Initial investigations on systems for measuring intraocular pressure,” *Sensors and Actuators* **85**, No. 1-3, pp. 287-291, 2000.

[15] Y.-S. Lin, S. Hanson, F. Albano, C. Tokunaga, R. Haque, K. Wise, A. Sastry, D. Blaauw, D. Sylvester, “Low-Voltage Circuit Design for Widespread Sensing

Applications," *International Symposium on Circuits and Systems*, pp. 2558-2561, 2008.
[16] K. Roy, S. Mukhopadhyay, H. Mahmoodi-Meimand, "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," *Proceedings of the IEEE*, Vol. 91, pp. 305-327, 2003.
[17] 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 Journal of Research and Development*, Vol. 50, No. 4/5, pp. 469-490, 2006.
[18] B. Zhai, S. Hanson, D. Blaauw, D. Sylvester, "Analysis and Mitigation of Variability in Subthreshold Design," *International Symposium on Low Power Electronics and Design*, pp. 20-25, 2005.
[19] L. Chang, D. Fried, J. Hergenrother, J. Sleight, R. Dennard, R. Montoye, L. Sekaric, S. McNab, A. Topol, C. Adams, K. Guarini, W. Haensch, "Stable SRAM Cell Design for the 32nm Node and Beyond," *Symposium on VLSI Technology*, pp. 128-129, 2005.
[20] R. Swanson, J. Meindl, "Ion-Implanted Complementary MOS Transistors in Low-Voltage Circuits," *Journal of Solid-State Circuits*, Vol. 7, pp. 146-153, 1972.
[21] H. Soeleman, K. Roy, "Ultra-low power digital subthreshold logic circuits," *International Symposium on Low Power Electronics and Design*, pp. 94-96, 1999.
[22] S. Hanson, B. Zhai, M. Seok, B. Cline, K. Zhou, M. Singhal, M. Minuth, J. Olson, L. Nazhandali, T. Austin, D. Sylvester, D. Blaauw, "Exploring Variability and Performance in a Sub-200mV Processor," *Journal of Solid-State Circuits* **43**, No. 4, pp. 881-891, 2008.
[23] B. Zhai, D. Blaauw, D. Sylvester, K. Flautner, "Theoretical and Practical Limits of Dynamic Voltage Scaling," *Design Automation Conference*, pp. 868-873, 2004.
[24] B. Calhoun, A. Chandrakasan, "Characterizing and Modeling Minimum Energy Operation for Subthreshold Circuits," *International Symposium on Low Power Electronics and Design*, pp. 90-95, 2004.
[25] C. H. Kim, K. Soeleman, K. Roy, "Ultra-Low-Power DLMS Adaptive Filter for Hearing Aid Applications," *Transactions on Very Large Scale Integration Systems*, Vol. 11, No. 6, pp. 1058-1067, 2003.
[26] A. Wang and A. Chandrakasan, "A 180 mV FFT processor using subthreshold circuit techniques," *International Solid-State Circuits Conference*, 2004, pp. 292-293.
[27] B. Calhoun, A. Wang, A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *Journal of Solid-State Circuits*, Vol. 40, pp. 1778-1786, 2005.
[28] B. Zhai, L. Nazhandali, J. Olson, A. Reeves, M. Minuth, R. Helfand, S. Pant, D. Blaauw, T. Austin, "A 2.60pJ/Inst Subthreshold Sensor Processor for Optimal Energy Efficiency," *Symposium on VLSI Circuits*, pp. 154-155, 2006.

[29] B. Calhoun, A. Chandrakasan, "Analyzing Static Noise Margin for Sub-threshold SRAM in 65nm CMOS," *European Solid-State Circuits Conference*, pp. 363-366, 2005.

[30] L. Chang, Y. Nakamura, R. Montoye, J. Sawada, A. Martin, K. Kinishita, F. Gebara, K. Agarwal, D. Acharyya, W. Haensch, K. Hosokawa, D. Jamsek, "A 5.3GHz 8T-SRAM with Operation Down to 0.41V in 65nm CMOS," *Symposium on VLSI Circuits*, pp. 252-253, 2007.

[31] N. Verma and A. P. Chandrakasan, "A 65nm 8T sub-V, SRAM employing sense-amplifier redundancy," *International Solid-State Circuits Conference*, pp. 327-328, 2007.

[32] 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," *International Solid-State Circuits Conference*, pp. 329-330, 2007.

[33] J. Kulkarni, K. Kim, K. Roy, "A 160 mV, Fully Differential, Robust Schmitt Trigger Based Sub-threshold SRAM," *International Symposium on Low Power Electronics and Design*, pp. 171-176, 2007.

[34] S. Hanson, B. Zhai, D. Blaauw, D. Sylvester, A. Bryant, X. Wang, "Energy Optimality and Variability in Subthreshold Design," *International Symposium on Low Power Electronics and Design*, pp. 363-365, 2006.

[35] J. Meindl, J. Davis, "The Fundamental Limit on Binary Switching Energy for Terascale Integration (TSI)," *Journal of Solid-State Circuits*, Vol. 35, pp. 1515-1516, 2000.

[36] J. Kwong, A. Chandrakasan, "Variation-Driven Device Sizing for Minimum Energy Sub-threshold Circuits," *International Symposium on Low Power Electronics and Design*, pp. 8-13, 2006.

[37] J. Kim, N. Hardavellas, K. Mai, B. Falsafi, J. Hoe, "Multi-Bit Error Tolerant Caches Using Two-Dimensional Error Coding," *International Symposium on Microarchitecture*, pp. 197-209, 2007.

[38] W. Haensch, et al., "Silicon CMOS devices beyond scaling," *IBM J. Res. & Dev.*, pp. 339-361 (2006).

[39] B. Yu, et al., "Short-Channel Effect Improved by Lateral Channel-Engineering in Deep-Submicrometer MOSFET's," *IEEE Trans. Elect. Devices*, pp. 627-634 (1997).

[40] Z.K. Lee, M.B. McIlrath, D.A. Antoniadis, "Two-Dimensional Doping Profile Characterization of MOSFET's by Inverse Modeling Using I-V Characteristics in the Subthreshold Region," *IEEE Trans. Elect. Devices*, pp. 1640-1649 (1999).

[41] B.C. Paul, A. Raychowdhury, K. Roy, "Device optimization for digital subthreshold logic operation," *IEEE Trans. Elect. Devices*, pp. 237-247 (2005).

[42] W. Zhao, Y. Cao, "New Generation of Predictive Technology Model for Sub-45nm Design Exploration," *Int. Symp. on Quality Elect. Design,* pp. 585-590, 2006.

[43] E.P. Gusev, et al., "Advanced High-κ Dielectric Stacks with PolySi and Metal Gates: Recent Progress and Current Challenges," *IBM J. Res. & Dev.*, pp. 387-410 (2006).

[44] Z. Luo, et al., "High Performance and Low Power Transistors Integrated in 65nm Bulk CMOS Technology," *Int. Electron Devices Meeting*, pp. 661-664, 2004.
[45] *The Int. Technology Roadmap for Semiconductors*, 2005.
[46] E.J. Nowak, "Maintaining the benefits of CMOS scaling when scaling bogs down," *IBM J. Res. & Dev*, pp. 169-180 (2002).
[47] Y. Taur, T.H. Ning, *Fundamentals of Modern VLSI Devices*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
[48] G. Chen, D. Blaauw, T. Mudge, D. Sylvester, N.S. Kim, "Yield-Driven Near-Threshold SRAM Design," *International Conference on Computer-Aided Design,* pp. 660-666, 2007.
[49] A.J. Bhavnagarwala, X. Tang, J.D. Meindl, "The Impact of Intrinsic Device Fluctuations on CMOS SRAM Cell Stability," *IEEE J. Solid-State Circuits **36***, pp. 658-665, 2001.
[50] A. Asenov, A. R. Brown, J.H. Davies, S. Kaya, and G. Slavcheva, "Simulation of Intrinsic Parameter Fluctuations in Decananometer and Nanometer-Scale MOSFETs," *IEEE Trans. Elect. Devices*, pp. 1837-1852, 2003.
[51] P. Stolk, F. Widdershoven, D.B.M Klaassen, "Modeling Statistical Dopant Fluctuations in MOS Transistors," *IEEE Trans. Elect. Devices*, pp. 1960-1971, 1998.
[52] D.J. Frank, Y. Taur, M. Ieong, H.-S.P. Wong, "Monte Carlo Modeling of Threshold Variation Due to Dopant Fluctuations," *Symp. VLSI Technol.*, pp. 169-170, 1999.
[53] L. Nazhandali, M. Minuth, B. Zhai, J. Olson, T. Austin, D. Blaauw, "A second-generation sensor network processor with application-driven memory optimizations and out-of-order execution," *International Conference on Compilers, Architecture and Synthesis for Embedded Systems*, pp. 249-256, 2005.
[54] J. Chen, L. Clark, Y. Cao, "Robust Design of High Fan-In/Out Subthreshold Circuits," *International Conference on Computer Design*, pp. 405-410, 2005.
[55] 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," *International Solid-State Circuits Conference*, pp. 200-201, 2002.
[56] K. Bernstein, D.J. Frank, A.E. Gattiker, W. Haensch, B.L. Ji, S.R. Nassif, E.J. Nowak, D.J. Pearson, N.J. Rohrer, "High-performance CMOS variability in the 65-nm regime and beyond," *IBM Journal of Research and Development*, Vol. 50, No. 4/5, pp. 433-449, 2006.
[57] J. T. Kao, M. Miyazaki and A. P. Chandrakasan, "A 175-mV multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *Journal of Solid-State Circuits*, Vol. 37, No. 11, pp. 1545-1554, 2002.
[58] A. Bryant, J. Brown, P. Cottrell, M. Ketchen, J. Ellis-Monaghan, E. Nowak, "Low-Power CMOS at Vdd=4kT/q," *Device Research Conference*, pp. 22-23, 2001.
[59] Y. Ramadass, A. Chandrakasan, "Minimum Energy Tracking Loop with Embedded DC-DC Converter Delivering Voltages Down to 250mV in 65nm CMOS," *International Solid-State Circuits Conference*, pp. 64-65, 2007.

[60] G. Ono, M. Miyazaki, "Threshold-Voltage Balance for Minimum Supply Operation," *Journal of Solid-State Circuits*, vol. 38, No. 5, pp. 830-833, 2003.
[61] T. Kim, H. Eom, J. Keane, C. Kim, "Utilizing Reverse Short Channel Effect for Optimal Subthreshold Circuit Design," *International Symposium on Low Power Electronics and Design*, pp. 127-130, 2006.
[62] Y. Taur, E. Nowak, "CMOS Devices Below 0.1µm: How High Will Performance Go?," *International Electron Devices Meeting*, pp. 215-218, 1997.
[63] S. Hanson, D. Sylvester, D. Blaauw, "A new technique for jointly optimizing gate sizing and supply voltage in ultra-low energy circuits," *International Symposium on Low Power Electronics and Design*, pp. 338-341, 2006.
[64] B. Calhoun, A. Chandrakasan, "Ultra-Dynamic Voltage Scaling Using Sub-threshold Operation and Local Voltage Dithering in 90nm CMOS," *International Solid-State Circuits Conference*, pp. 300-301, 2005.
[65] A. Wang, A. Chandrakasan, "A 180mV FFT Processor Using Subthreshold Circuit Techniques," *International Solid-State Circuits Conference*, pp. 292-293, 2004.
[66] M. Seok, S. Hanson, Y.-S. Lin, Z. Foo, D. Kim, Y. Lee, N. Liu, D. Sylvester, D. Blaauw, "The Phoenix Processor: A 30pW Platform for Sensor Applications," *Symposium on VLSI* Circuits, 2008.
[67] R.B. Tremaine, P.A. Franaszek, J.T. Robinson, C.O. Schulz, T.B. Smith, M.E. Wazlowski, P.M. Bland, "IBM Memory Expansion Technology (MXT)," *IBM Journal of Research and Development* **45**, No. 2, pp. 271-286, 2001.
[68] C. Lefurgy, P. Bird, I. Chen, T. Mudge, "Improving Code Density Using Compression Techniques," *International Symposium on Microarchitecture*, pp. 194-203, 1997.
[69] M. Seok, S. Hanson, J. Seo, D. Sylvester, D. Blaauw, "Ultra-Low Voltage ROM Design," *Custom Integrated Circuits Conference,* to appear 2008.
[70] L. Chang, D. Fried, J. Hergenrother, J. Sleight, R. Dennard, R. Montoye, L. Sekaric, S. McNab, A. Topol, C. Adams, K. Guarini, W. Haensch, "Stable SRAM Cell Design for the 32nm Node and Beyond," *Symposium on VLSI Technology*, pp. 128-129, 2005.
[71] B. Calhoun, A. Chandrakasan, "A 256kb Sub-threshold SRAM in 65nm CMOS," *International Solid-State Circuits Conference*, pp. 2592-2593, 2006.
[72] B. Zhai, D. Blaauw, D. Sylvester, S. Hanson, "A sub-200mV 6T SRAM in 0.13µm CMOS," *International Solid-State Circuits Conference,* pp. 332-333, 2007.
[73] M. Seok, S. Hanson, D. Sylvester, D. Blaauw, "Analysis and Optimization of Sleep Modes in Subthreshold Circuit Design," *Design Automation Conference*, pp. 694-699, 2007.
[74] S. Narendra, S. Borkar, V. De, D. Antoniadis, A. Chandrakasan, "Scaling of Stack Effect and its Application for Leakage Reduction," *International Symposium on Low Power Electronics and Design*, pp. 195-200, 2001.
[75] "Muskegon Meteorological Data," Great Lakes Environmental Research Laboratory, National Oceanic and Atmospheric Administration, 20 August 2008, <http://www.glerl.noaa.gov/metdata/mkg/archive/>.

[76] M. Seok, D. Sylvester, D. Blaauw, "Optimal Technology Selection for Minimizing Energy and Variability in Low Voltage Applications," *International Symposium on Low Power Electronics and Design*, 2008.
[77] S. Hanson, M. Seok, D. Sylvester, D. Blaauw, "Nanometer Device Scaling in Subthreshold Circuits," *Design Automation Conference*, 2007.
[78] J. Kwong, Y. Ramadass, N. Verma, M. Koesler, K. Huber, H. Moormann, A. Chandrakasan, "A 65nm Sub-V_t Microcontroller with Integrated SRAM and Switched-Capacitor DC-DC Converter," *International Solid-State Circuits Conference*, pp. 318-319, 2008.
[79] L. Nazhandali, B. Zhai, J. Olson, A. Reeves, M. Minuth, R. Helfand, S. Pant, T. Austin, and D. Blaauw, "Energy Optimization of Subthreshold-Voltage Sensor Network Processors," *International Symposium on Computer Architecture*, pp. 197-207, 2005.
[80] M. Sheets, F. Burghardt, T. Karalar, J. Ammer, Y.H. Chee, J. Rabaey, "A Power-Managed Protocol Processor for Wireless Sensor Networks," *Symposium on VLSI Circuits*, pp. 212-213, 2006.
[81] C.H. Diaz, et al., "A 0.18µm CMOS Logic Technology with Dual Gate Oxide and Low-k Interconnect for High-Performance and Low-Power Applications," Symposium on VLSI Technology, pp. 11-12, 1999.
[82] H. Kawaguchi, K. Nose, T. Sakurai, "A CMOS Scheme for 0.5V Supply Voltage with Pico-Ampere Standby Current," *International Solid-State Circuits Conference*, pp. 192-193, 1998.
[83] M. Nagata, M. Homma, N. Takeda, T. Morie, A. Iwata, "A Smart CMOS Imager with Pixel Level PWM Signal Processing," *Symposium on VLSI Circuits*, pp. 141-144, 1999.
[84] D. Yang, A. Gamal, B. Fowler, H. Tian, "A 640x512 Image Sensor with Ultrawide Dynamic Range Floating-Point Pixel-Level ADC," *Journal of Solid-State Circuits* ***34***, No. 12, pp. 1821-1834, 1999.
[85] K. Kagawa, S. Shishido, M. Nunoshita, J. Ohta, "A 3.6pW/frame-pixel 1.35V PWM CMOS Imager with Dynamic Pixel Readout and No Static Bias Current," *International Solid-State Circuits Conference*, pp.54-55, 2008.
[86] A. El Gamal, H. Eltoukhy, "CMOS Image Sensors," *IEEE Circuits and Devices Magazine* ***21***, No. 3, 2005.
[87] Z. Ignjatovic, M. Bocko, "A 0.88nW/pixel, 99.6 dB Linear-Dynamic-Range Fully-Digital Image Sensor Employing a Pixel-Level Sigma-Delta ADC," *Symposium on VLSI Circuits*, pp. 23-24, 2006.
[88] D. Blaauw, S. Kalaiselvan, K. Lai, W-H. Ma, S. Pant, C. Tokunaga, S. Das, D. Bull, "Razor II: In-Situ Error Detection and Correction for PVT and SER Tolerance," *International Solid-State Circuits Conference*, 2008.
[89] S. Ghosh, S. Bhunia, K. Roy, "CRISTA: A New Paradigm for Low-Power, Variation-Tolerant, and Adaptive Circuit Synthesis Using Critical Path Isolation," *Transactions on Computer-Aided Design of Integrated Circuits and Systems* ***26***, No. 11, pp. 1947-1956, 2007.
[90] R. Harrison, R. Kier, S. Kim, L. Rieth, D. Warren, N. Ledbetter, G. Clark, F. Solzbacher, C. Chestek, V. Gilja, P. Nuyujukian, S. Ryu, K. Shenoy, "A

Wireless Neural Interface for Chronic Recording," *Biomedical Circuits and Systems Conference*, pp. 125-128, 2008.

[91] D. Wentzloff, F. Lee, D. Daly, M. Bhardwaj, P. Mercier, A. Chandrakasan, "Energy Efficient Pulsed-UWB CMOS Circuits and Systems," *International Conference on Ultra-Wideband*, pp. 282-287, 2007.

[92] N. Pletcher, S. Gambini, J. Rabaey, "A 65μW, 1.9 GHz RF to Digital Baseband Wakeup Receiver for Wireless Sensor Nodes," *Custom Integrated Circuits Conference*, pp. 539-542, 2007.

[93] B. Zhai, S. Hanson, D. Blaauw, D. Sylvester, "A Variation-Tolerant Sub-200 mV 6-T Subthreshold SRAM," *Journal of Solid-State Circuits*, pp. 2338-2348, 2008.

[94] B. Warneke, K. Pister, "An Ultra-Low Energy Microcontroller for Smart Dust Wireless Sensor Networks," *International Solid-State Circuits Conference*, pp. 316-317, 2004.