

# Lifetime Improvement in Body Sensor Networks

---

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

---

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy (Electrical Engineering)

by

Yousef Shakhsheer

December 2013



# Abstract

Body sensor networks (BSNs) promise to provide significant benefits to the healthcare domain. BSNs consist of multiple nodes that sense, process, and transmit health data and an aggregator that manages nodes, processes data, and passes information between the nodes and the base station. Though BSNs have tremendous potential for improving health care, their practical adoption must overcome technical and social challenges such as form factor, battery life, and reliability. BSN nodes will not be adopted if they are unsightly, large and bulky, or require frequent battery changes or charging. The focus of this work is to improve BSN node and aggregator lifetime to improve the overall BSN lifetime. Improved BSN lifetime will augment remote, long-term monitoring of chronically-ill patients, firefighters, and athletes.

The most desirable BSN lifetime is an infinite device lifetime; battery power constrains the lifetime of the device to a finite period of time. Energy harvesting mechanisms such as solar power, thermoelectric generation, and piezoelectric provide an alternative power source to these energy constrained devices for the possibility of infinite device lifetime. For a desired BSN node form factor of less than  $1\text{cm}^3$ , energy harvesting mechanisms can produce  $50\text{-}100\mu\text{Ws}$ . With careful design and tight system integration, BSN nodes can achieve this power consumption and potentially achieve an indefinite lifetime. This work presents the first wireless biosignal acquisition chip powered solely from a thermoelectric harvester and/or RF power with integrated supply regulation, analog front end, power management, subthreshold digital signal processing, and a transmitter. This work also investigates BSN architecture decisions such as tradeoffs between custom controllers and generic microcontrollers, to inform

future designs. Additionally, this work demonstrates the first-implemented on-chip, closed-loop power management system capable of adjusting node power consumption to the amount of energy harvested and explores the power management design space.

Aggregators cannot operate exclusively from energy harvesting due to their high processing/communication requirements and the inability to harvest a sufficient amount of energy. Therefore, the aggregator has become the determining factor in the BSNs lifetime. To extend lifetime of the aggregator and the whole BSN, we can leverage the aggregator's variable workload to improve battery lifetime by applying a fine-grained dynamic voltage scaling (DVS) scheme; this workload changes with amount of data that needs to be processed and the number of nodes with which are being communicated. This method, called Panoptic ("all-inclusive") Dynamic Voltage Scaling (PDVS), extends DVS to a finer granularity in space and time, allowing for much more flexible and energy efficient design and therefore a longer battery lifetime for the aggregator and a longer lifetime for the system. This work applies PDVS to a DSP data-flow processor as a proof of concept to show energy savings over multiple benchmark workloads and characterizes the overheads of PDVS.

# Approval Sheet

This dissertation is submitted in partial fulfillment of the requirements for the degree of  
Doctor of Philosophy (Electrical Engineering)

Yousef Shakhsheer

---

Yousef Shakhsheer

This dissertation has been read and approved by the Examining Committee:

Benton Calhoun

---

Benton Calhoun, Advisor

Mircea Stan

---

Mircea Stan, Committee Chair

John Lach

---

John Lach

Harry Powell

---

Harry Powell

Kevin Skadron

---

Kevin Skadron

Accepted for the School of Engineering and Applied Science:

James H. Aylor

---

James H. Aylor, Dean, School of Engineering and Applied Science

December 2013

*To my family.*

# Acknowledgements

I have been very blessed to have a lot of great people in my life who have supported me, collaborated with me, and helped me through graduate school. Without them, I would not be the person I am today.

I am very grateful to my advisor, Professor Ben Calhoun, for his guidance during my time in Bengroup. He has been my mentor, allowing me to grow as a researcher, an engineer, and a person. I am thankful for the opportunity to work on three really interesting, challenging, once-in-a-lifetime projects. I have learned so much from him. Ben has always been one of my biggest supporters and patient with me when I struggled and made mistakes. Thank you for everything.

I'm thankful for having a very involved Ph.D. Committee. Professor John Lach, Professor Harry Powell, Professor Mircea Stan, and Professor Kevin Skadron have helped guide my dissertation work from the beginning of my graduate school experience through classes and projects to completion of my dissertation work. I appreciate the insight they've given me through the years, the conversations we've had, and how they always pushed me to strive for more.

I have been very fortunate to be surrounded by so many Bengroup members throughout my career that have helped me grow as a researcher and as a person. They have made the paper deadlines, tapeouts, and testing bearable and even somewhat fun. In our group, I've spent the most time working with Kyle Craig, Yanqing Zhang, and Alicia Klinefelter. I am thankful for all that time together and our friendship. I appreciate the conversations (both

work and personal) we've had, the countless intellectual arguments, spending time together outside of work, and all the help they have given me throughout the years. I appreciate how we have been able to enjoy the successes together and been able to work together when we were struggling. Some of my most memorable times in grad school have been successfully fixing LVS errors on our PDVS test chip at 3AM (which is unheard of for me), teaching YQ how to drive, and calling Alicia "new girl" for years. I am very thankful for all the help that Aatmesh Shrivastava has given me throughout the years and our friendship; he is truly the master of multiple trades. Another Bengroup member that had a big impact on my graduate school career was Dr. Randy Mann. Randy got me excited about research and provided invaluable life advice that I still carry with me and have tried to pass on to other graduate students. I have so many good memories from Bengroup. Thanks to everyone in Bengroup: Dr. Jiajing Wang, Dr. Satya Nalam, Dr. Joe Ryan, Sudhanshu Khanna, Steve Jocke, Jim Boley, Seyi Ayorinde, Peter Beshay, Patricia Gonzalez, Divya Akella, Yu Huang, He Qi, Arijit Banerjee, Terry Tigner, Abhishek Roy, Farah Yahya, Chris Lukas, and Harsh Patel. You all have been an integral part of my graduate school experience. I have enjoyed bouncing ideas around and hanging out.

I am grateful for the other graduate students that helped shape my work. Ben Boudaoud helped me time and time again. Whether it was embedded design or fleshing out ideas for this dissertation, he was always there when I needed him. I am grateful to my good friend, Saad Arrabi. He has been a brother to me and made graduate school interesting and fun with interesting intellectual conversations. I am grateful having had the chance to work with Nate Roberts; working with him was fun and productive. I am very thankful to Stuart Wooters, Jonathan Bolus, Jeff Brantley, Helen Zhang, and Jason Silver. I value my personal and professional interactions with them.

I cannot imagine having gotten my Ph.D. without the love and support of my family. They are always there for me. My mom and dad have always supported me and provided for me in every way possible. They have been my role models from a young age, showing me

that it is possible to have a successful work life and family life. They have exemplified the core values that I hold so dear and continue to guide me through new situations. My brother has always been there to support me in every way possible. He has always been a role model I've looked up to from a young age. I also thank him for proofreading my dissertation.

Throughout my graduate school experience, my friends have always been there to support me. Friends are the family you choose and I've been so blessed to have them in my life. They have always been there to listen to me gripe, give me a place to escape graduate school, and give me different perspectives when I needed it. They have made my life a lot more fun and full and have helped me achieve news successes.

And last but not least, I am very grateful to my Seminole Trail Volunteer Fire Department family. This group of remarkable people has taught me so much about helping people, remaining calm in high pressure situations, and how to be a leader. I thank the chiefs, line officers, and firefighters for the support they've given me for nine years and the life lessons they've taught me throughout the years.

# Contents

<b>Contents</b>	<b>viii</b>
List of Tables . . . . .	xi
List of Figures . . . . .	xii
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Thesis . . . . .	3
1.2.1 Enabling Energy Harvesting in BSN Nodes . . . . .	4
1.2.2 Power Management for Energy Harvesting-Powered BSN Nodes . . . . .	4
1.2.3 Improving Battery Lifetime in BSN Aggregators . . . . .	4
1.3 Approach . . . . .	4
1.3.1 Improving Device Lifetime in BSN Nodes . . . . .	5
1.3.2 Improving Battery Lifetime in BSN Aggregators . . . . .	6
1.4 Dissertation Contributions and Organization . . . . .	7
<b>2 Background</b>	<b>10</b>
2.1 Body Sensor Networks . . . . .	10
2.1.1 Structure . . . . .	11
2.1.2 BSN Nodes . . . . .	12
2.1.3 Aggregator . . . . .	13
2.1.4 BSN Requirements and Challenges . . . . .	14
2.1.5 Energy Sources . . . . .	15
2.1.6 Implementation . . . . .	18
2.1.7 General Strategies for Extending Lifetime in BSNs . . . . .	19
<b>3 Enabling Energy Harvesting in BSN Nodes</b>	<b>27</b>
3.1 Related Work . . . . .	29
3.1.1 COTS . . . . .	29
3.1.2 ASIC . . . . .	30
3.2 BSN Node . . . . .	31
3.2.1 Architecture Overview . . . . .	33
3.2.2 Energy Harvesting/ Supply Regulation Subsystem . . . . .	34
3.2.3 Closed-Loop Power Management . . . . .	37
3.2.4 Flexible Biosignal Datapath . . . . .	38
3.2.5 System Measurements . . . . .	48

3.3	BSN Architectural Explorations . . . . .	51
3.3.1	Microcontrollers . . . . .	52
3.3.2	Bus Architecture . . . . .	59
3.4	Summary and Conclusions . . . . .	63
<b>4</b>	<b>Power Management for Energy Harvesting-Powered BSN Nodes</b>	<b>65</b>
4.1	Related Work . . . . .	67
4.2	Power Management Explorations . . . . .	68
4.2.1	Sampling Node Health . . . . .	69
4.2.2	Number of Operating Modes . . . . .	73
4.2.3	Single-Cycle Modification . . . . .	74
4.3	Revision 1 . . . . .	77
4.3.1	Sampling the Health of the Node . . . . .	78
4.3.2	Stoplight . . . . .	78
4.3.3	Threshold Values . . . . .	79
4.3.4	Measured Results . . . . .	80
4.4	Revision 2 . . . . .	81
4.4.1	Chip Architecture . . . . .	81
4.4.2	Programmability/Flexibility . . . . .	84
4.4.3	Sampling the Energy . . . . .	84
4.4.4	Reduction of Lock up . . . . .	85
4.4.5	Results . . . . .	86
4.5	Conclusions . . . . .	87
<b>5</b>	<b>Improving Battery Lifetime in BSN Aggregators</b>	<b>88</b>
5.1	Related Work . . . . .	91
5.2	Test Chip . . . . .	92
5.2.1	Test Setup . . . . .	94
5.2.2	PDVS Overheads . . . . .	95
5.2.3	Chip Results . . . . .	96
5.3	Number of Resources . . . . .	99
5.3.1	Test Setup . . . . .	100
5.3.2	Results . . . . .	100
5.4	$V_{DD}$ Switching . . . . .	104
5.5	Summary and Conclusions . . . . .	106
<b>6</b>	<b>Conclusions</b>	<b>108</b>
6.1	Summary of Contributions . . . . .	108
6.2	Team and Individual Contributions . . . . .	110
6.2.1	Broad Impact of this Work . . . . .	110
6.2.2	Conclusions and Open Problems . . . . .	111
<b>A</b>	<b>Acronyms</b>	<b>116</b>
<b>B</b>	<b>Publications</b>	<b>119</b>

Contents

x

**Bibliography**

**122**

# List of Tables

2.1	Common Energy Harvesting Mechanisms [1]	15
3.1	DPM Instruction Set Architecture	40
3.2	Connections for Bus 1 and Bus 2	42
3.3	Measured Performance Summary [2]	51
3.4	Projected BSN Power Using State-of-the-Art Components	52
3.5	MCU vs Accelerators: Energy Efficiency per Sample	52
3.6	MCU Equivalent Instructions	55
3.7	Microcontroller Case Studies	57
3.8	Performance Comparisons with State-of-the-Art BSN Nodes [2]	63
4.1	Energy/Power of RO vs ADC Time-Multiplexing	72
4.2	Calculated Time and Energy Overheads of Different Power Management Schemes	77
4.3	DPM Operating Mode	79
5.1	PDVS Chip Summary	93
5.2	DVS State of the Art Implementation Comparisons [10]	107

# List of Figures

1.1	An example of BSNs. This BSN consists of multiple nodes and an aggregator. This aggregator interfaces to a base station to share information with other stakeholders, such as doctors and emergency personnel [3]. . . . .	2
2.1	An example of a BSN. This BSN consists of multiple nodes that acquire different biosignals. Each node communicates with the aggregator. . . . .	11
2.2	A power consumption breakdown of a BSN node (TEMPO 3.1 [4]) with gyroscopes off [5]. . . . .	13
2.3	Communication between BSN nodes and the aggregator. The aggregator is responsible for node coordination, aggregating data from the nodes, and interfacing with the base station. This asymmetric approach allows for lower power consumption within the node. . . . .	14
2.4	Sensor power budgets for desired lifetime. [6]. . . . .	16
2.5	Correlation of energy harvesting mechanisms over an average workday. Each segment represents the average amount of power that an individual could expect to harvest at any given time when all sources are being deployed [3].	16
2.6	Ideal energy-delay curve of non-DVS and DVS [5]. . . . .	20
2.7	Illustration of supply voltage at 0.5 workload in a a. non-DVS scheme and b. DVS scheme. . . . .	21
2.8	DVS voltage generation schemes. a. DC-DC converter approach. b. Power switch approach. . . . .	22
2.9	Generating multiple energy values with two voltages through dithering. . . . .	23
3.1	High-level diagram of conventional BSN solution (top) and the proposed (bottom) solution with energy harvesting, integrated power management, and ultra-low power flexible DSP architecture [2]. . . . .	32
3.2	System block diagram for the proposed chip comprising the energy harvesting/supply regulation, analog front-end (AFE), subthreshold digital signal processing, and transmitter subsystems [2]. . . . .	32
3.3	System block diagram for the proposed chip comprising the energy harvesting/supply regulation, analog front-end (AFE), subthreshold digital signal processing, and transmitter subsystems [7]. . . . .	35
3.4	Measured startup sequence for the chip. A RF pulse kick-starts $V_{\text{Boost}}$ , allowing the boost converter to turn on. The chip is able to sustain a usable voltage from harvested energy (VTEG) from then on [2]. . . . .	36

3.5	Measured results from an on-body thermal-harvesting experiment with 4x4 cm <sup>2</sup> COTS TEG [7]. . . . .	37
3.6	The DPM is responsible for power management, node control, data flow management, and overseeing all on-node processing [8]. . . . .	38
3.7	Controls from the DPM to a generic accelerator block [8]. . . . .	38
3.8	DPM instruction formats [8]. . . . .	38
3.9	Block diagram of the AFE, including the path for monitoring $V_{CAP}$ [7]. . . . .	42
3.10	Block diagram of subthreshold data processing subsystem [7]. . . . .	43
3.11	Energy-delay curves for MCU, RR+AFib accelerator, and 30-Tap, 1-Channel FIR [2]. . . . .	44
3.12	Integration of DPM and GPP MCU. DPM and MCU share the same memory, and instructions are steered to the right processor. The MCU can be power-gated when idle. In this way, both efficient control of the node and generic processing are implemented [7]. . . . .	45
3.13	Measured system experiment showing correct data acquisition and streaming from the transmitter. Total power of 397 $\mu$ W prevents long use of this mode from harvested power [2]. . . . .	49
3.14	Measured system results for an R-R extraction algorithm. Measured system results are for acquiring ECG, extracting R-R intervals, and sending RF updates for R-R every 5s. Total power in this mode is 19 $\mu$ W drawn from a 30mV input. Measured accuracy results for R-R also shown [2]. . . . .	49
3.15	Measured system AFib demo experiment using R-R extractor and AFib accelerator. Normal and atrial fibrillation heart waveforms from MIT-BIH database [9]. Last 8 beats of raw ECG are stored in DMEM and streamed over TX if AFib is detected. Total chip power in this mode is 19 $\mu$ W from a 30mV input [7]. . . . .	50
3.16	Current breakdown for R-R extraction experiment. The current distribution is roughly evenly distributed amongst main contributors, and the originally power-hungry transmitter consumption is now nearly mitigated [7]. . . . .	50
3.17	Annotated chip die photo [2]. . . . .	51
3.18	a. Measured energy-delay curve for the DPM and MCU for NOPs, ENs, and BUS instructions. b. Measured power consumption of the DPM and MCU . . . . .	56
3.19	a. Structure of global decoding scheme. b. Structure of local decoding scheme. . . . .	60
3.20	a. Simulated active and idle energy at 0.5V of global and local decoding schemes. b. Routing complexity of the global and local decoding schemes. . . . .	61
3.21	Average bus energy of 20 and 40 addresses as a function of utilization. . . . .	62
4.1	RO configuration for digitizing $V_{Boost}$ . . . . .	70
4.2	Simulated number of RO pulses per window over the range of voltages. . . . .	71
4.3	a. Simulated average power of RO and ADC. b. Simulated energy of RO and ADC. . . . .	72
4.4	a. Simulated active and idle energy at 0.5V of the number of operating modes of a power manager. b. Simulated average energy of power manager based on activity factor for different numbers of operating modes. . . . .	74

4.5	Override structure of the DPM stoplight. The stoplight compares the $V_{\text{Boost}}$ value to the threshold, selects the operating mode, and outputs control bits to the chip [8]. . . . .	79
4.6	Measured DPM closed-loop power management response [2]. . . . .	80
4.7	Top level layout of the second BSN node. . . . .	82
4.8	Block diagram of the second BSN node. . . . .	83
4.9	a. Simplified model of energy harvester, capacitor, and load. b. Simulated sequence of “lockup.” The DPM switches between green and yellow mode due to transmission. . . . .	86
4.10	Measured DPM revision energy per operation. . . . .	87
5.1	Structure of PDVS. . . . .	90
5.2	Block diagram of the PDVS data flow processor. SRAMs and control serve four data paths for direct comparison of PDVS with $SV_{\text{DD}}$ and $MV_{\text{DD}}$ [10]. . . . .	92
5.3	Structure of single- $V_{\text{DD}}$ . . . . .	94
5.4	Structure of multi- $V_{\text{DD}}$ . . . . .	94
5.5	Test setup for the PDVS chip. a. Test board. Daughterboard mates with the test board. b. FPGA daughterboard [11]. . . . .	95
5.6	a. Simulated level conversion overhead varying $V_{\text{DDL}}$ for both the adder and multiplier. b. Simulated virtual- $V_{\text{DD}}$ switching overhead varying $V_{\text{DDL}}$ for both the adder and multiplier [11]. . . . .	95
5.7	a. Measured energy of adder and multiplier vs. $V_{\text{DD}}$ b. Measured vs simulated delay of adder vs $V_{\text{DD}}$ c. Measured vs. simulated delay of multiplier vs $V_{\text{DD}}$ d. Average measured energy (w/ overheads) vs. workload across 4 different DFGs [10]. . . . .	97
5.8	Change in average power & instantaneous power as the workload changes over time; power waveform shows dithering between two rates to achieve an intermediate rate, resulting in near optimal average energy savings [10]. . . . .	97
5.9	a-c. Measured energy benefit (including overhead) of PDVS & $MV_{\text{DD}}$ vs. $SV_{\text{DD}}$ for single function single rate (SFSR) & single function multi rate (SFMR) at 67% and 50% at constant area d. Area benefit of PDVS over $MV_{\text{DD}}$ [10]. . . . .	98
5.10	Die photo of the 90nm PDVS test chip [10]. . . . .	99
5.11	Number of clock cycles required for the 100% rate. . . . .	101
5.12	Simulated active energy of FIR4 and FIR8 benchmarks. Legend: M- $M_{\text{DD}}$ . P- PDVS. Number- number of resources. . . . .	102
5.13	Simulated active energy of FIR12 and FIR16 benchmarks. Legend: M- $M_{\text{DD}}$ . P- PDVS. Number- number of resources. . . . .	103
5.14	a. Simulated energy per operation while switching from $V_{\text{DDH}}$ to $V_{\text{DDL}}$ . b. Simulated energy of running three operations with and without a NOP across various $V_{\text{DDs}}$ . c. Timing diagram of the switching methodologies [11]. . . . .	104

# Chapter 1

## Introduction

### 1.1 Motivation

Body sensor networks (BSNs) have the potential to revolutionize the medical field. BSNs (Figure 1.1) consist of multiple nodes on, near, or within a human body which provide sensing, processing, and communication capabilities and an aggregator, which coordinates the nodes, processes additional data, and communicates between the nodes and the base station. The base station serves as a link to stakeholders such as doctors and first responders. The information acquired by BSNs can be used to improve health care through drug delivery, augmented sensory stimulation for the deaf or blind, and improved movement of prosthetic limbs [12]. Though BSNs have tremendous potential for improving health care, their practical adoption must overcome technical and social challenges, such as form factor, battery life, reliability, privacy, interoperability, and ease of use. BSN nodes will not be used if they are too inconvenient (require frequent battery changes or charging), uncomfortable, or unsightly [5].

The focus of this dissertation is to improve operational lifetime in BSN nodes and aggregators while adhering to their form factor requirements. Improvements to BSN lifetime would enable longer term monitoring in chronically-ill patients, first responders, and athletes.

The most desirable lifetime for BSNs is an infinite lifetime. BSN nodes and aggregators

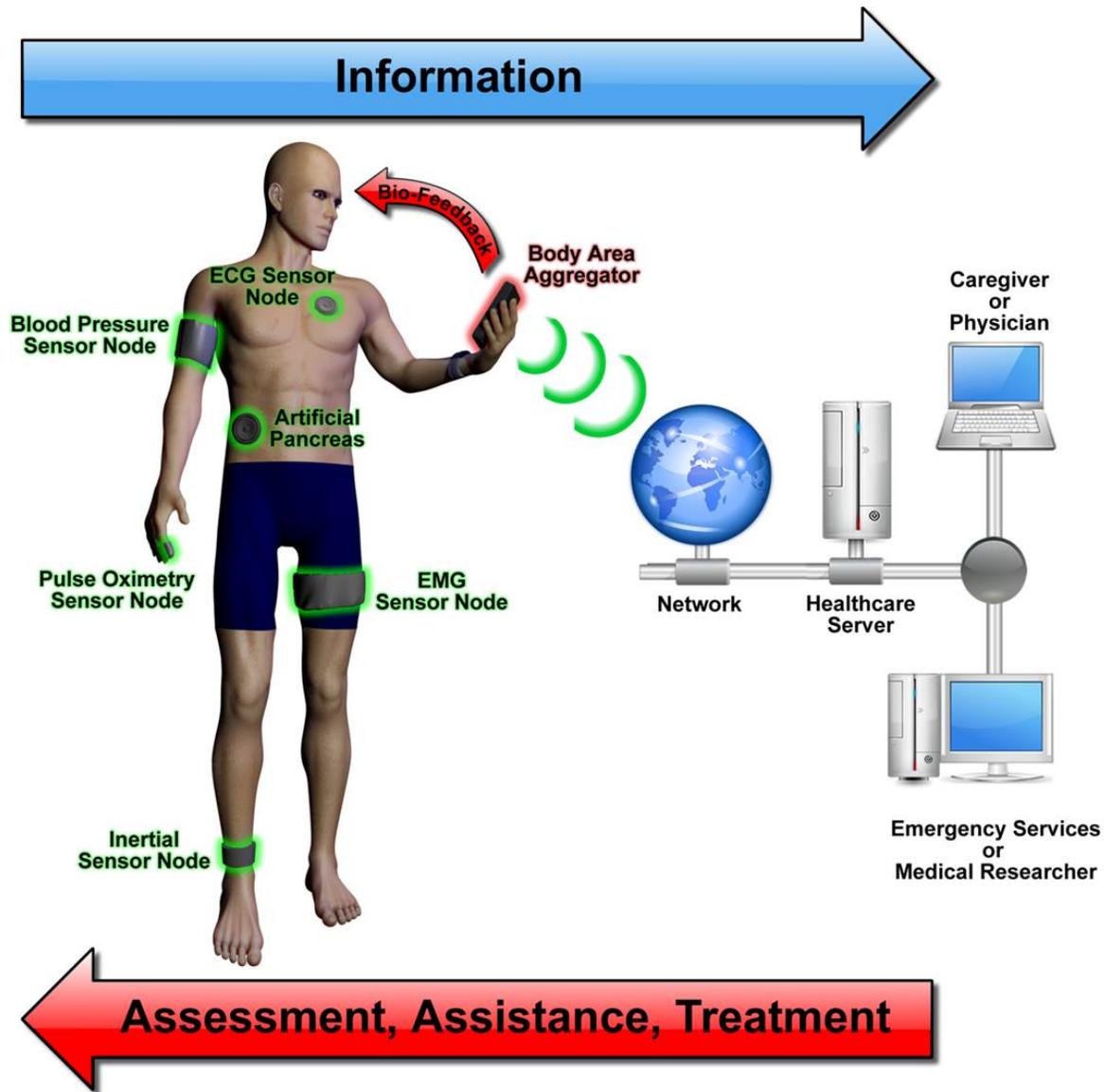


Figure 1.1: An example of BSNs. This BSN consists of multiple nodes and an aggregator. This aggregator interfaces to a base station to share information with other stakeholders, such as doctors and emergency personnel [3].

are powered by a battery in most applications [4] [13]; these batteries store a finite amount of energy and, thus, constrain the BSN's lifetime. Due to the BSN node form factor constraint (less than  $1\text{cm}^3$ ), the chosen batteries must be small. The target size for BSN devices limits the energy budget to a maximum of 100s Joules, which is 2-3 orders of magnitude less than a cellular phone battery [12].

Energy harvesting mechanisms such as solar power, thermoelectric generation, and

piezoelectric, provide an alternative, desirable power source to these energy constrained devices. These sources can provide an indefinite source of power, and, thus, provide the possibility of an infinite BSN lifetime. For the desired form factor, energy harvesting mechanisms can produce 50-100 $\mu$ Ws. There are several issues with using harvested energy. High power operations, such as transmitting data wirelessly, can consume 100s of  $\mu$ Ws and would likely exceed the power budget set by energy harvesting. Additionally, energy harvesters' power output is highly environment-dependent [14]. For example, solar harvesting provides reduced power in cloudy conditions or at night. However, the benefits of powering the node from energy harvesting and the prospects of an infinite lifetime outweigh these manageable issues.

BSN aggregators, which are commonly cellular phones, are not capable of operating solely off harvested energy due to their high processing/communication requirements and the inability of energy harvesters to obtain a sufficient amount of energy to power the aggregator. Therefore, aggregators must be powered by a battery. Assuming BSN nodes are capable of being powered off of energy harvesting and are capable of an infinite lifetime, BSN aggregators would be the limiting factor in BSN lifetime. Without these aggregators, the BSN nodes would not be able to communicate with essential stakeholders such as doctors or emergency medical technicians, thus rendering BSNs ineffective. Therefore, we must find a way to also improve battery lifetime within BSN aggregators.

## 1.2 Thesis

To improve lifetime within BSN nodes, we can power BSN nodes exclusively from energy harvesting by lowering node power consumption through tight system integration, selection of energy efficient components, application of low power principles, and heavy duty-cycling. To improve lifetime within BSN aggregators, we can apply fine-grained dynamic voltage scaling (DVS) to leverage its variable workload. Improvements in BSN node and aggregator lifetime

will result in an improvement in BSN lifetime.

### 1.2.1 Enabling Energy Harvesting in BSN Nodes

A BSN node design utilizing low power digital, analog and radio frequency (RF) with intelligent duty cycling and tight system integration will achieve an average power consumption that is capable of running exclusively from harvested energy ( $<50\mu\text{W}$ ), thus providing the node with a potentially infinite lifetime.

### 1.2.2 Power Management for Energy Harvesting-Powered BSN Nodes

A closed loop, energy harvesting-specific threshold-based power management scheme that utilizes single-cycle power modification will be capable of adjusting node power consumption to varying rates of harvested energy, resulting in a longer lifetime in energy harvesting powered BSN nodes and a reduced probability of node death.

### 1.2.3 Improving Battery Lifetime in BSN Aggregators

Application of fine-grained dynamic voltage scaling (DVS) using the Panoptic DVS scheme will result in higher energy savings on a system level compared to single- $V_{\text{DD}}$  and multi- $V_{\text{DD}}$  alternatives, resulting in a lifetime improvement in battery-powered electronics, such as a BSN aggregator.

## 1.3 Approach

We separate the problem of improving BSN system lifetime into two aspects: improving device lifetime in BSN nodes and improving device lifetime in BSN aggregators.

### 1.3.1 Improving Device Lifetime in BSN Nodes

BSN nodes are commonly powered by a battery [4] [13] [15]. Traditionally, battery size is increased to improve lifetime in battery-powered devices. However, we cannot increase battery size within the BSN node to increase the device lifetime due to the form factor requirements. Harvesting ambient energy is an appealing alternative to battery power. The extreme power limits ( $<50\mu\text{Ws}$ ) and varying amounts of power from energy harvesting BSN devices necessitate a fundamental departure from the traditional design of wireless embedded sensing.

To power BSN nodes exclusively from harvested power, nodes must consume ultra-low power (ULP). It is infeasible to use commercial off the shelf (COTS) components to build a system powered solely from energy harvesting that is capable of achieving an indefinite node lifetime. COTS BSNs can consume tens of mA during operation and will likely fail to meet the required form factor due to the large number of discrete components.

Instead, we pursue an integrated ULP ASIC (application specific integrated circuits) system-on-chip (SoC) approach. ASICs achieve very efficient operation because they are hardwired to perform a specific task or set of tasks. To ensure sustained operation of the node using harvested energy, on-node processing to reduce the amount of data transmitted, power management, and ULP circuits are key. We utilize recent advances in energy harvesting, low voltage boost circuits, dynamic power management, subthreshold processing, bio-signal front-ends, and low power radio transmitters to achieve an integrated, reconfigurable wireless BSN SoC for ECG, EMG, and EEG applications with completely battery-free operation. This SoC is capable of running indefinitely from harvested energy. We use this SoC as a proof of concept that BSN nodes can be powered solely from energy harvesting and as a platform to investigate important architectural decisions to inform future BSN node designs.

Utilizing low power blocks for energy harvesting-powered BSNs does not ensure an infinite lifetime. Energy harvesters' power output is highly environment-dependent. Operating in a low power mode that ignores the time varying nature of power available to the node will result

in node death when the node's power consumption exceeds the energy harvester's power output. Node power consumption must adapt to the varying nature of harvested energy. Therefore, a power management system is required that is capable of tracking or sampling the health of the node (power consumption of the node versus energy harvesting power) and adjusting the power consumption accordingly. To date, no ASIC energy harvesting-specific power managers have been implemented. This work investigates two methods of sampling the health of the node and power consumption modification techniques to adapt to this varying energy. These techniques improve lifetime in energy harvesting-powered BSN nodes.

### 1.3.2 Improving Battery Lifetime in BSN Aggregators

BSN aggregators cannot operate solely off of energy harvesting; energy harvesting cannot produce sufficient energy for the aggregator's power consumption. To maintain the aggregator's current form factor or reduce it, we assume that we cannot increase the battery size and, therefore, must improve energy efficiency to augment device lifetime. We leverage the aggregator's variable workload to improve battery lifetime. Workload varies as a result of changes in the amount of data that needs to be processed or changes in the number of nodes with which are communicated with. This system occasionally requires high performance, but its varying workload requirements remain below this upper limit for the majority of its lifetime. Designing the system in a static fashion to support this peak performance can substantially increase the total system energy.

Dynamic voltage scaling (DVS) provides the ability to trade-off energy and delay in system with varying workloads. When the system's highest performance is not required, DVS scales the supply voltage to match the workload, providing quadratic energy savings while still meeting processing requirements. Traditional DVS implementations suffer from coarse spatial (the ability to assign different components in a design to different voltages) and temporal (speed at which the voltage to a component can change) granularities. Most DVS implementations are limited to a spatial granularity at the microprocessor core level to

entire chip [16] [17] [18]. DVS techniques generally rely on DC-DC converters to adjust the supply voltage. These off-chip DC-DC converters traditionally limit temporal granularity because they required tens to hundreds of microseconds to adjust the supply voltage [19]. The coarse spatial and temporal granularity of traditional DVS limits the energy efficiency of these systems and reduces the amount of energy that can be saved.

To improve battery life, we propose applying a method called Panoptic ("all-inclusive") Dynamic Voltage Scaling (PDVS) [20] within the aggregator to reduce computational power. To improve spatial and temporal granularity, PDVS uses multiple PMOS header switches at the component level to provide a local voltage from a discrete set of chip-wide shared voltages. To show the advantages and overheads of this approach and compare PDVS to common circuit topologies, we apply PDVS to a DSP data-flow processor. We use this chip as a proof of concept and to further investigate PDVS design decisions.

By enabling energy harvesting with energy harvesting-specific power management in BSN nodes and applying PDVS in BSN aggregators, we can improve the lifetime of BSNs, enabling these devices to collect data for a longer period of time for monitoring chronically ill patients, first responders, and athletes and improve patient care.

## 1.4 Dissertation Contributions and Organization

Body sensor networks (BSNs) have the potential to revolutionize the medical field and improve patient care. Information acquired by BSNs is used to improve healthcare through improving the duration of healthcare monitoring, drug delivery, and telemedicine. Improvements to device lifetime will improve user compliance and the duration of the healthcare monitoring. This dissertation discusses improvements to BSN node lifetime and BSN aggregators separately to improve the lifetime of BSNs as a whole.

## **Background**

Chapter 2 gives an in-depth tutorial on body sensors networks (BSNs) and dynamic voltage scaling (DVS). This tutorial provides a foundation to better understand this dissertation.

## **Enabling Energy Harvesting in BSN Nodes**

Chapter 3 focuses on improving lifetime in BSN nodes by enabling energy harvesting. This chapter presents the first wireless BSN node powered solely from a thermoelectric harvester and/or RF power with integrated supply regulation, analog front-end, power management, subthreshold digital signal processing, and a transmitter. This system has lower power, a lower minimum input supply voltage (30mV), and more complete system integration than all other reported wireless BSN nodes. This BSN node represents a significant advancement in system integration for ULP systems. Using the measurements from this fabricated BSN node, we explore the benefits of custom microcontrollers and local decoding bus topologies to inform future BSN node design.

## **Power Management for Energy Harvesting-Powered BSN Nodes**

Chapter 4 focuses on improving lifetime in BSN nodes powered solely by energy harvesting by implementing a power management scheme. We demonstrate the first implemented energy harvesting-specific power management system that is capable of adjust node power consumption as the amount harvested energy varies, thus improving lifetime. We describe the second revision of that power manager, which adds flexibility and scalability. We show the benefits of single cycle power modification and explore methods for sampling the health of the BSN node to inform future energy harvesting-specific power management designs.

## Improving Battery Lifetime in BSN Aggregators

Chapter 5 focuses on improving lifetime in the battery constrained BSN aggregator by applying PDVS to improve energy efficiency. We demonstrate the first full processor with PDVS, single-clock cycle  $V_{DD}$  switching,  $V_{DD}$  dithering, and the ability to switch between high performance DVS operation and a subthreshold mode of operation in silicon as a proof of concept of PDVS. We use this chip data to explore energy savings compared to single- $V_{DD}$  and multi- $V_{DD}$  alternatives. This chapter shows the benefits of PDVS as the number of components is varied, and the overheads and benefits of different speeds and sequences of  $V_{DD}$ -switching.

## Conclusion

Chapter 6 concludes the dissertation with a summary of the work. It also describes some key areas for future work to further improve lifetime in energy constrained devices.

# Chapter 2

## Background

<sup>1</sup> To provide a baseline for understanding the improvement in lifetime we achieve in this work, this chapter summarizes important, basic concepts body sensor networks (BSNs) such as structure, energy harvesting mechanisms, and circuit techniques for reducing power.

### 2.1 Body Sensor Networks

BSNs promise to provide significant benefits to the healthcare domain by enabling continuous monitoring and logging of patient bio-signal data, which can help medical personnel to diagnose, prevent, and respond to various illnesses such as diabetes, asthma, and heart attacks [21]. BSNs address the weaknesses of traditional patient data collection and medical care, such as imprecision (qualitative human observation) and under-sampling (infrequent assessment).

BSNs instrument the human body and its immediate surroundings. BSNs are capable of measuring physiological (i.e. heart rate, blood pressure, body temperature), physical (i.e. acceleration, posture), and ambient (i.e. light, ambient temperature, ozone) data for an extended period of time. This information can be used to augment bodily functions

---

<sup>1</sup>This chapter is based on the published papers titled: "Energy Efficient Design for Body Sensor Nodes" [YS7]

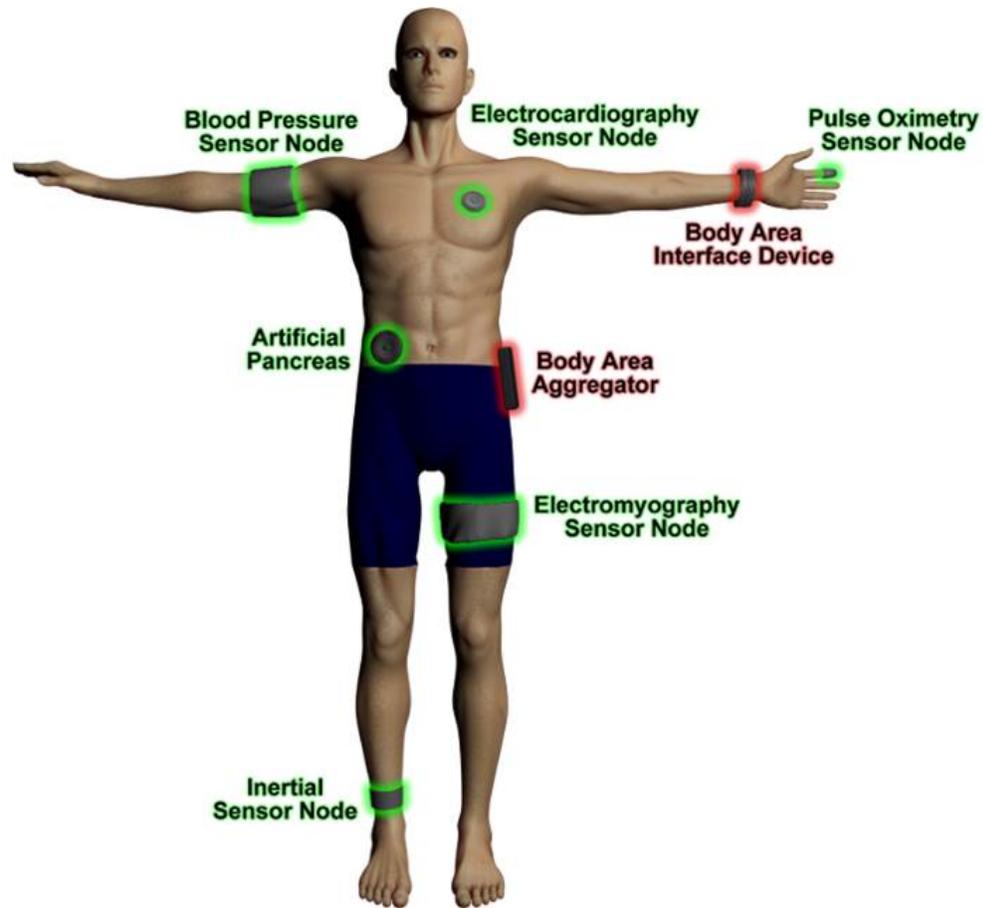


Figure 2.1: An example of a BSN. This BSN consists of multiple nodes that acquire different biosignals. Each node communicates with the aggregator.

through drug delivery, sensory stimulation for the deaf or blind, and maintaining blood sugar. Additionally, BSNs can provide new opportunities to extend healthcare to remote areas or reduce the amount of site visits for patients through telemedicine and can help protect those exposed to potentially life-threatening environments, such as soldiers, firefighters, and space explorers [3].

### 2.1.1 Structure

Each BSN consists of multiple interconnected nodes on, near, or within a human body and an aggregator, which together provide sensing, processing, and communication capabilities, as shown in Figure 2.1. BSNs typically use a star network and star-mesh hybrid network topology

for communication. In the star network topology, all nodes connect to the aggregator, which allows for high data throughput and simplified routing; no node-to-node communication is allowed. The star-mesh hybrid network topology allows communication between all nodes and the aggregator as well as node-to-node communication. Both topologies exploit the resource asymmetry between the aggregator and the node. Aggregators have more processing and communication capabilities as well as a bigger battery, allowing high powered operations to be allocated there rather than the resource-constrained node [3].

### 2.1.2 BSN Nodes

BSN nodes interface to their patient or their surroundings. These nodes require a very small profile to remain unobtrusive and wearable. BSN nodes typically consist of an energy source, one or more sensors, some processing capabilities, and a radio to communicate with an aggregator.

Sensing is fundamental in BSNs. Sensors fall into three categories: physiological, biokinetic, and ambient. Physiological sensors measure blood pressure, glucose, body temperature, blood oxygen, electrocardiography (ECG), electroencephalography (EEG), and electromyography (EMG). Biokinetic sensors measure motion and gait. Ambient sensors measure environmental conditions such as humidity, light, sound pressure level, and temperature.

Signal processing is used to extract valuable information from captured data from the sensors, such as transient data and trends. On average, processing data consumes less power than transmitting the data wirelessly; the reduction in the amount of data that needs to be transmitted reduces the data rate and the power consumption of the wireless radio. These characteristics create a trade-off between processing and communication: On-node signal processing will consume power to extract information, but also reduces the radio's data rate and power consumption.

On-node communication enables node coordination and transmission of raw biosignals or processed/compressed data. The node is capable of communicating with the aggregator, to

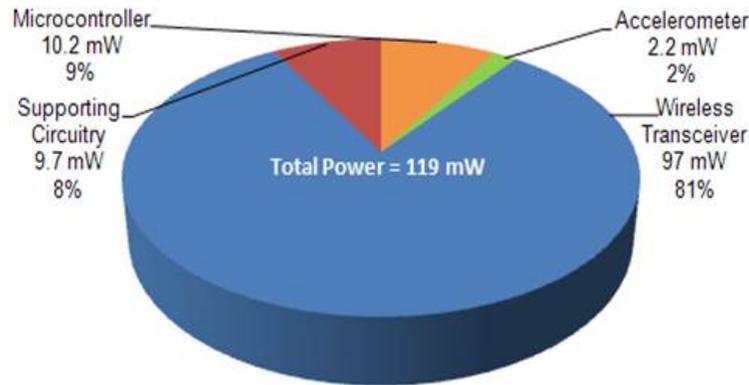


Figure 2.2: A power consumption breakdown of a BSN node (TEMPO 3.1 [4]) with gyroscopes off [5].

another node, or to multiple nodes, based on the network topology chosen. Communication power can dominate of the power budget of the node (Figure 2.2) [3].

### 2.1.3 Aggregator

The BSN aggregator plays a very important and often overlooked role in BSNs. The aggregator serves as the interface between the numerous BSN nodes and the base station. Aggregators are responsible for collecting data from these multiple BSN nodes and coordinating these nodes. Depending on the application, the aggregator is capable of turning nodes on and off and sending instructions to the nodes to set or adjust the sampling rate, transmission rate, and select the biosignal modality (Figure 2.3). Aggregators are also capable of providing a user interface to provide bio-feedback to patients or athletes. Additionally, they can possess its own sensing and processing capabilities.

In recent implementations, designers have used cellular phones as an aggregator [15] [22] [23]. Others have built their own custom aggregator [24]. However, improvements and widespread adoption of wireless protocol such as Bluetooth, cellular, and IEEE 802.11, interactive user interfaces such as touch screens, highly capable processors, and the likelihood of the user already owning a cellular phone make using cellular phones as aggregators attractive .



Figure 2.3: Communication between BSN nodes and the aggregator. The aggregator is responsible for node coordination, aggregating data from the nodes, and interfacing with the base station. This asymmetric approach allows for lower power consumption within the node.

### 2.1.4 BSN Requirements and Challenges

BSN's adoption must overcome formidable technical and social challenges (ie.. form factor, battery life, reliability, safety, security, ease of use, etc.) [3]. This section focuses primarily on the challenges of form factor, lifetime, and functionality.

The wearable nature of BSN devices requires a very small node form factor; users will not wear the node if it is big, bulky, and inconvenient. BSN nodes should ultimately have extremely small volumes of  $1\text{cm}^3$  or less. This form factor significantly limits the hardware resources and the power budget and conflicts with the long lifetime requirement. Battery energy density does not effectively scale down to the small sizes necessitated by the form factor. The target size for BSN nodes limits the battery energy budget to a maximum of 100s of Joules [12]. This necessitates investigation of power reduction techniques and alternatives

to batteries, such as energy harvesting, to extend lifetime. The next section discusses energy harvesting in detail.

The physical nature of BSNs introduces new challenges. The human body is an unpredictable physical environment that changes through the user’s motions and actions. This constant change creates changing requirements for sensing, actuation, and radio communication. Most BSN applications require different operations if a person is walking, sleeping, exercising, or has a medical emergency. BSNs will, therefore, acquire data from different sensors, utilize different actuators, and will have different requirements on communication and latency of data. At the same time, rapid physical movements change the radio communication channel characteristics (affecting power consumption) and as well as opportunities for energy harvesting (affecting power generation) if energy harvesting is being utilized. Thus, BSNs must be able to adapt to rapidly changing environment to be effective [12]

### 2.1.5 Energy Sources

Table 2.1: Common Energy Harvesting Mechanisms [1]

Power Source	Power density
Solar (outside)	15 (mW/cm <sup>2</sup> )
Solar (inside)	10 ( $\mu$ W/cm <sup>2</sup> )
Temperature (5 °C Gradient)	40 ( $\mu$ W/cm <sup>2</sup> )
Human Power	330 ( $\mu$ W/cm <sup>2</sup> )
Vibration	375 ( $\mu$ W/cm <sup>2</sup> )
Acoustic Noise	960 (nW/cm <sup>2</sup> )

Battery power significantly constrains the lifetime of the device. In traditional battery-powered systems, lifetime refers to the time between battery replacements or recharges. To improve lifetime in battery-powered devices, the battery size is traditionally increased. However, we cannot increase battery size within the BSN node due to the form factor requirements. Therefore, to extend battery lifetime, we must reduce the average current

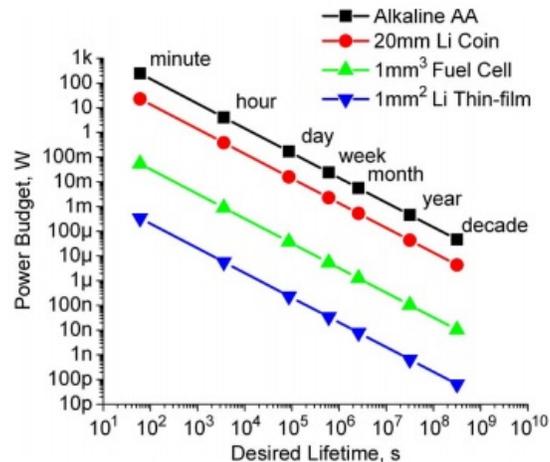


Figure 2.4: Sensor power budgets for desired lifetime. [6].

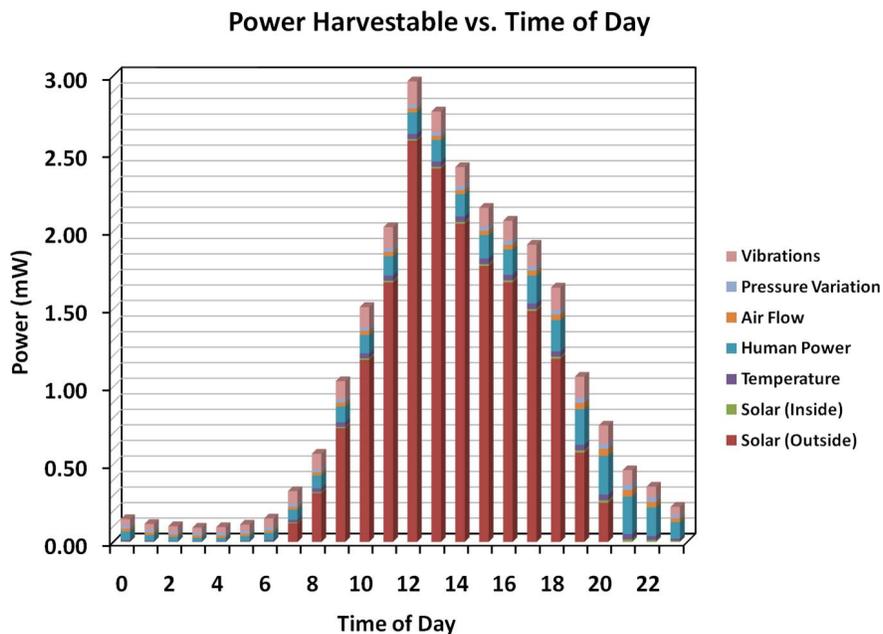


Figure 2.5: Correlation of energy harvesting mechanisms over an average workday. Each segment represents the average amount of power that an individual could expect to harvest at any given time when all sources are being deployed [3].

consumption of the node. Figure 2.4 shows the maximum power budget as a function of the desired system lifetime based [6]. However, this still yields a finite lifetime.

Harvesting ambient energy offers an appealing alternative to battery-based operation. Harvesting energy generates energy from ambient sources such as sunlight, temperature

gradient, or vibration. These sources can provide an indefinite source of power and, thus, provide the possibility of an infinite device lifetime.

Table 2.1 shows the power density of some common sources [1]. The amount of power obtained from the energy harvester directly depends on the area of the energy harvester. For the desired BSN node form factor, energy harvesting mechanisms can produce 50-100 $\mu$ Ws [14]. A variety of energy harvesting mechanisms exist, and the correct energy harvesting source is highly dependent on the application. For example, utilizing vibrations in a patient that has Parkinson's disease could effectively power a BSN node. However, utilizing energy harvesting a source introduces many challenges.

First, energy harvesting sources present a very non-ideal supply voltage. Depending on the mechanism, energy harvesting mechanism output varying voltages that are at extremely low (sub-100mV) voltage levels. Second, high peak current levels, caused by high-powered components such as wireless radios, cause significant voltage drops in energy harvesting sources. Peak current levels in commercial transceivers range between of 10-30mA. Third, the power output of energy harvesters is highly environment-dependent, thus varying over time. For example, solar cells cannot harvest energy at night. Additionally, thermoelectric generation power output changes as the temperature changes. Figure 2.5 shows a time profile of different energy harvesting mechanisms and how they vary ideally over the day [3]. This, however, assumes ideal conditions and does not show any changes in the environment, such as shorter or colder winter days.

These energy harvesting-specific challenges necessitate a change in design methodology from traditional battery-powered design. First, energy harvesting powered BSNs require a boost converter to upconvert the energy harvester's low output voltages to stable, useable supply to operate the BSN node. Second, excess power/energy should be stored for future use in a capacitor or a rechargeable battery when the power produced by the harvester exceeds the power consumed by the node. This energy can be used for high power operations, such as wireless transmission, and when the energy harvester's power output is not sufficient for

the node power consumption. Third, to sustain operation, the BSN node must consume less power than the energy harvesting mechanism produces or temporarily store energy for future use. Last, the node should have some form of power management in which the node power consumption is adjusted when the output of the power harvester dips.

The metrics for evaluating energy harvesting systems are different from those used for battery powered systems. Harvested energy is different from battery energy in two ways. First, it is an inexhaustible resource which, if appropriately used, can allow the node to operate indefinitely where as a battery which is a limited resource. Second, there is an uncertainty associated with its availability and measurement, compared to the energy stored in the battery which that is known deterministically [25]. Thus, power management methods based on battery status are not always applicable to energy harvesting systems.

### **2.1.6 Implementation**

There are two main approaches taken to designing these BSN nodes: a commercial off the shelf (COTS) approach and application specific integrated circuits (ASIC) approach. Both approaches will be explained in the following subsections.

#### **COTS Platforms**

COTS BSN nodes consist of separately packaged components integrated onto a printed circuit board. Most COTS platforms include sensors, an input analog amplifier, a digital filter, a microcontroller, a battery, a reference oscillator, and a radio transceiver. COTS nodes provide solid development platforms that are flexible and easy to build for rapid prototyping. This allows for development of processing algorithms, measurement methods, and communication protocols. Many COTS nodes employ Bluetooth or Zigbee radios for communication. The radio's power consumption is a substantial part of the node's power consumption, reaching the 100s of mW range [26] [27] [28]. The high energy consumption of these COTS-based

nodes' limit the battery lifetime of a COTS platform to the range of 1 to 3 days [12], thus requiring a battery change and making adoption of these nodes less likely.

### **ASIC Platforms**

ASIC are custom designed chips. These custom nodes are application specific instead of flexible and generic. ASICs achieve very efficient operation because they are hardwired to do a specific task or set of tasks. However, they perform other functions either inefficiently or are unable to perform them at all. Complete ASICs nodes attempt to integrate all parts of the BSN on-die. This includes a power management system, processing and control units, an interface to biosignals or sensors, and a radio. Reduction in the number of passive components results in a smaller form factor and a reduction in cost.

Custom BSN chip design continues to be an emerging field, and there are a limited number of full systems. This may be a result of the expertise in a diverse set of areas such as analog front end design, low power digital design, power management design, and radio design. These systems-on-chips (SoCs) require the expertise in a diverse set of areas such as analog front end design, low power digital design, power management design, and radio design as well as tight integration [12]. However, these ASIC nodes have the potential to improve BSN node lifetime and form factor, leading to more social adoption and better results.

#### **2.1.7 General Strategies for Extending Lifetime in BSNs**

In this section, we examine several techniques to reduce power and energy in COTS and ASIC BSN nodes. We look at dynamic voltage scaling, subthreshold operation, power gating, on-node computation, and use of accelerators to reduce power and energy to improve lifetime.

##### **Dynamic Voltage Scaling**

Dynamic voltage scaling (DVS) is a common approach to reducing energy in circuits [29] [30] [31]. In DVS, the circuit's supply voltage is increased or decreased at run time based on

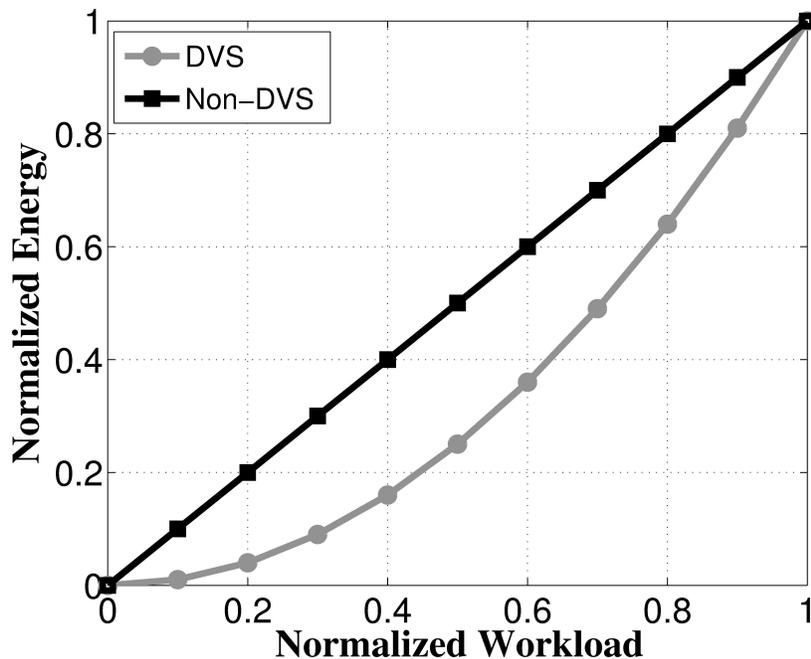


Figure 2.6: Ideal energy-delay curve of non-DVS and DVS [5].

circumstances (i.e. energy requirements and scheduling deadlines). This circuit can be as big as a whole chip or as small an individual component. As voltage is increased, active energy increases and delay decreases. As voltage is decreased, active energy decreases and the delay increases.

Active energy of a circuit is characterized by:

$$E_{active} = C_{switch} V_{DD}^2 \quad (2.1)$$

where  $C_{switch}$  refers to the circuit's capacitive load that is switching and  $V_{DD}$  refers to the circuit's voltage. Note that active energy is time independent, unlike power. A reduction in voltage results in a quadratic energy savings for an approximately linear reduction in performance, thus, making voltage a very strong knob of energy reduction.

Figure 2.6 shows the energy of a circuit with and without ideal DVS as the workload changes. Ideal DVS refers to a system in which all voltages are available with no overheads. Workload refers to the amount of data needed to be processed in a set period of time. For

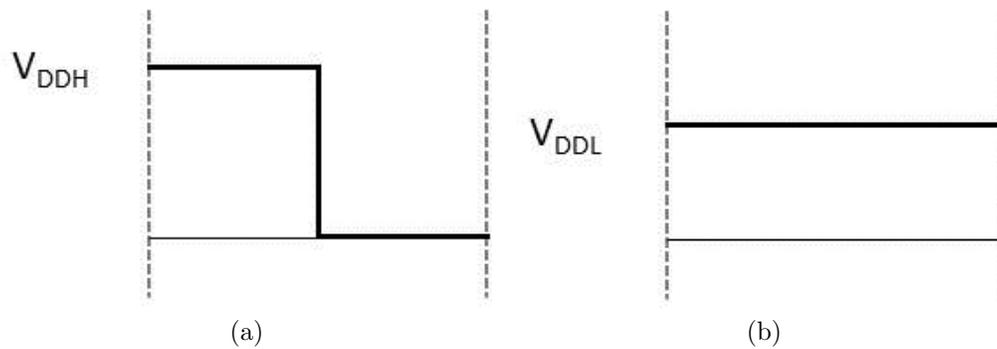


Figure 2.7: Illustration of supply voltage at 0.5 workload in a. non-DVS scheme and b. DVS scheme.

example, if we assume that a BSN aggregator which processes 100 pieces of data in a set period of time has a normalized workload of 1.0, an aggregator that processes 50 pieces of data in the same period of time has a normalized workload of 0.5.

Ideal DVS systems are capable of achieving a workload of 1.0 at the same energy as a non-DVS system at the highest speed. As the workload decreases, the non-DVS system completes certain tasks ahead of their deadline and the processor enters a low-leakage sleep mode for the remainder of the time (Figure 2.7a). This corresponds to a linear energy savings under the assumption that idling consumes no leakage power/energy.

In DVS systems, however, the performance level is reduced by lowering the voltage during periods of low utilization such that the circuit finishes each task just in time, stretching each task to its deadline, as shown in Figure 2.7b. The circuit/system active energy scales quadratically. Therefore, choosing the lowest voltage for an operation maximizes energy savings.

**Voltage Generation** In ideal DVS, an infinite number of voltages are available to the circuit. Therefore, the lowest possible voltage meeting the timing constraint is available. In COTS devices, a separate voltage regulator or DC-DC converter chip is utilized. In ASIC devices, DVS is implemented in two methods (Figure 2.8): a DC-DC converters capable of varying its voltage output is assigned to each voltage island (a group of circuits that

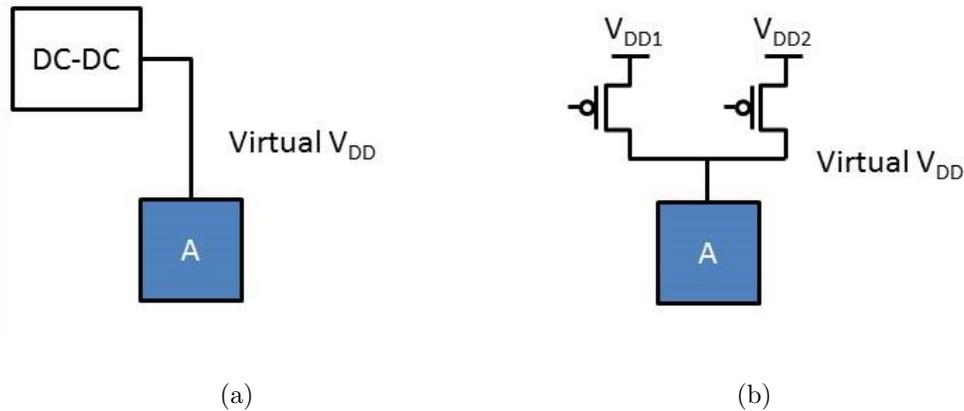


Figure 2.8: DVS voltage generation schemes. a. DC-DC converter approach. b. Power switch approach.

share voltage that is independent of other islands) or by switching between different stable voltages available to the circuit using power switches [32]. The rest of this DVS section refers specifically to ASIC systems.

Switching a DC-DC converter’s voltage output of takes a long time [19]. Power switches, primarily PMOS headers, can be used instead to connect the circuit to one of a finite set of voltages routed throughout the chip and switch voltages much faster than DC-DC converters [33]. The use of power switches enables an approach called voltage dithering. Voltage dithering was proposed as a low overhead implementation of DVS to provide near-optimum power savings using only a few discrete voltages [32]. Voltage dithering uses a small number of discrete voltages to approximate the ideal DVS curve by operating for part of the time at a higher voltage and the remainder of the time at a lower voltage. This averages the performance and achieves any effective performance rate. Figure 2.9 shows dithering to achieve the intermediate rates. Note that dithering does not result in quadratic savings.

**DVS Overheads in ASIC Systems** The implementation of any DVS scheme incurs area, delay, and energy overheads over non-DVS schemes. These overheads are a result of switching

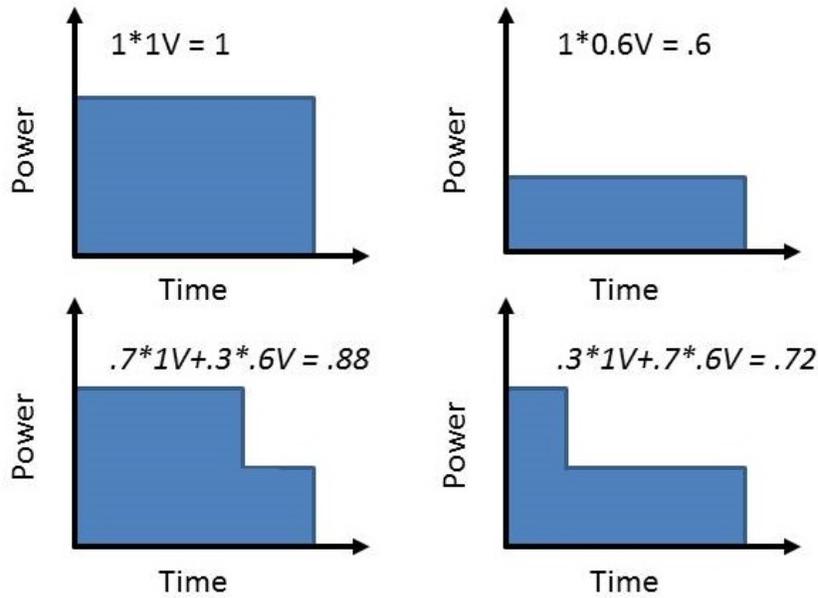


Figure 2.9: Generating multiple energy values with two voltages through dithering.

voltage, interfacing between different voltage domains, and overheads specific to the DC-DC converter and the power switch implementations.

To switch voltages, the virtual  $V_{DD}$  rail (Figure 2.8), which serves as the effective supply to the circuit, must be charged or discharged. Switching from a higher voltage to a lower voltage will incur no delay or energy overhead. Switching from a lower voltage to a higher voltage will incur a delay and energy overhead. In both implementations, the switch delay and energy are directly proportional to the size of the block being switched. In the power switch DVS implementation, the switching delay also depends directly on the size of the header. The header switch size is directly proportional to the switching speed.

In systems with multiple voltage islands, each island has a possibility of being at a different voltage. Communication between these islands occurs at different voltages; they should be level converted to prevent short circuit current. These level converters introduce area, energy and delay overheads.

In the DC-DC converter DVS implementation, each voltage island requires its own DC-DC converter, which results in an area penalty.

In the power switch DVS implementation, the introduction of the headers adds an additional resistance between the power supply and the circuit. This header dissipates power, thus dropping voltage across the header. The voltage drop results in a speed degradation and an additional energy overhead. Additionally, the voltage rails that are routed throughout the chip add an additional area overhead.

For DVS to be effective and improve lifetime, the energy savings from switching to a lower voltage must outweigh the energy overheads for switching, level converting, and, if using a power switch DVS scheme, the power switch energy overheads while still meeting performance constraints.

### Subthreshold Operating in ASIC Systems

Subthreshold (sub- $V_T$ ) operation in ASIC digital integrated circuits provides energy efficient processing. Sub- $V_T$  circuits use a supply voltage that is below the threshold voltage,  $V_T$ , of the transistors. The transistors are off by conventional definitions, but the change in transistor gate-to-source voltage ( $V_{GS}$ ) produces a difference in sub- $V_T$  conduction current that allows static digital circuits to operate robustly. Sub- $V_T$  circuits operate much slower than they would be at superthreshold, but their low speeds are sufficient for many BSN operations (up to tens of MHz). Due to the quadratic relationship between energy and the supply voltage, the main advantage of sub- $V_T$  operation is a reduction in energy consumption of over 10X compared to traditional circuit implementations. Sub- $V_T$  operation has been shown to minimize energy per operation in CMOS circuits [34].

There are some challenges to making sub- $V_T$  digital circuits work. Most notably, the reduced  $I_{on} / I_{off}$  ratio combines with process variations in the threshold voltage to increase the potential for circuit failure. Sub- $V_T$  circuits also must be level converted to interface with superthreshold design, such as radios or sensors. Nevertheless, sub- $V_T$  operation is an emerging approach that can help extend lifetime within BSN nodes [35].

## Power gating

Power gating is a common solution to reduce leakage current in idle circuits in COTS and ASIC designs. Applications with infrequent data acquisition can take advantage of power gating between active periods. Blocks of various sizes (from full cores to individual components) are power gated by disconnecting the voltage supply or ground from the block when it is idle. Many COTS components, such as [36], have built-in enable pins to turn off the COTS component. The duration these components can be power gated for is heavily dependent on the component's turn on energy and time as well as the application (i.e. sampling rating). Power gating can be implemented by inserting a PMOS header or NMOS footer between the block and its voltage supply or ground in both COTS and ASIC designs, reducing leakage current. When the block needs to operate normally, the power switch turns on and reconnects the virtual voltage rail to its nominal level.

## On-Node Computation

Wireless transmission of sensed data is the largest power consumer in most current BSNs [37]. This problem is particularly acute in medical BSN applications, in which sensor data rates may be relatively high. Significant power reduction can be achieved through on-node signal processing and data management; this can drastically reduce the number of bits that need to be transmitted. This includes traditional data compression along with signal processing techniques such as pattern classification and feature detection algorithms. Low power signal processing therefore becomes increasingly important to BSN power efficiency.

## Accelerators

Dedicated accelerators lower the energy processing compared to using a general purpose processor (GPP). GPPs exhibit poor energy efficiency due to the overhead of fetching and decoding the instructions that are required to perform a given operation in the datapath. For low power embedded applications like BSNs, general purpose computation is generally

performed in fairly simple microcontrollers [35] [38]. Sophisticated operations like a fast Fourier transform (FFT) or data processing algorithm will thus require numerous instructions in the simple core.

An alternative to using GPPs for processing is hardware accelerators. Hardware accelerators use a dedicated ASIC circuit to implement a specific function, resulting in up to a 1000x reduction in energy consumption compared to a GPP [7]. These accelerators can vary in function; BSN nodes and aggregators may have multiple accelerators. Some examples of hardware accelerators are multipliers, fast Fourier transforms, or FIR filters. These operations take several instructions over multiple clock cycles to complete on a GPP, consuming a large amount of energy and time. An ASIC accelerator is able to complete the operation likely in less clock cycles and energy. However, using hardware accelerators incurs the overhead of increased area and additional leakage paths.

# Chapter 3

## Enabling Energy Harvesting in BSN Nodes

<sup>1</sup> Body sensor networks (BSNs) show promise in the healthcare domain by enabling continuous monitoring and recording patients' biosignal data. This technology helps medical personnel diagnose, prevent, and respond to various illnesses such as diabetes, asthma, and cardiac arrhythmias. Though BSNs show great potential, BSNs have many design challenges that may impede their widespread adoption. One of the most critical issues is device lifetime. In many applications, such as long-term monitoring of chronic illnesses, limited lifetimes severely undermine the effectiveness of BSNs. To improve lifetime in BSNs, we must improve the lifetime of its parts: the BSN node and aggregator. This chapter focuses on improving device lifetime within BSN nodes.

Currently, most BSN nodes are battery-powered [4] [13]. Supplying the node with sufficient power and energy over a long lifetime while meeting the form factor constraints ( $<1\text{cm}^3$ ) poses a challenge. A large battery does not allow the node meet its form factor constraint; a small battery requires frequent changing or charging, and reduces the chance the patient

---

<sup>1</sup>This chapter is based on the published papers titled: "A Batteryless 19  $\mu\text{W}$  MICS/ISM-Band Energy Harvesting Body Sensor Node SoC for ExG Applications" [YS2], "A Custom Processor for Node and Power Management of a Battery-less Body Sensor Node in 130nm CMOS" [YS3], and "A Battery-less 19 $\mu\text{W}$  MICS/ISM-Band Energy Harvesting Body Area Sensor Node SoC" [YS6].

will wear the device regularly. Both, however, restrict the device lifetime to a finite period of time.

To eliminate battery changing/charging and enable a possible infinite lifetime in BSN nodes, BSN nodes can operate solely from energy harvesting instead of using a battery. Table 2.1 shows the energy density of some common sources [1]. However, operating solely from energy harvesting introduces new challenges. The full system must consume less power than the amount harvested (approximately  $50\text{-}100\mu\text{W}$ ). High power components such as the transmitter must be heavily duty-cycled to meet the power constraint. The node must cope with time varying harvested energy profiles [39]. To be capable of achieving this low power consumption to run off of an energy harvester, a BSN must use low power digital, analog and radio frequency (RF) circuits with intelligent duty cycling and tight system integration.

This chapter describes a fabricated 130nm state-of-the-art batteryless BSN node powered solely from an energy harvester. Using measurements from this fabricated BSN node, we explore two BSN architectural decisions, utilizing custom versus generic microcontrollers and global versus local decoding for bus architecture, to inform future BSN designs. We extend the state-of-the-art and knowledge in this area in several ways in this chapter. We utilize recent advances in energy harvesting, low voltage boost circuits, dynamic power management, subthreshold processing, biosignal front-ends, and low power RF transmitters to realize an integrated reconfigurable wireless BSN node system on chip (SoC) for electrocardiography (ECG), electromyography (EMG), and electroencephalography (EEG) applications with autonomous power management for completely battery-free operation [2]. This SoC can run indefinitely from energy harvested from body heat while worn and decrease cost through its high integration and targeting a wide range of bio-electric sensing applications. The additional explorations into microcontrollers, bus architecture, and power profiles inform future BSN node design to further improve lifetime.

## 3.1 Related Work

BSNs are being used in a growing list of applications and have become commercially available. These applications include heart monitoring, fall detection, gait analysis, and pulse oximetry. This section details the state-of-the-art commercial-off-the-shelf (COTs) and application specific integrated circuit (ASIC) BSN nodes.

### 3.1.1 COTS

The literature in this area ranges over multiple applications and over multiple power budgets. We look at three state of the art systems.

The first is Technology Enabled Medical Precision Observation (TEMPO) platform [4]. TEMPO is a custom inertial BSN developed at the University of Virginia that provides sensing with six degrees of freedom (three axes of both linear acceleration and rotational rate) and wireless data streaming in the form factor of a wristwatch. The system utilizes two MEMS gyroscopes and one accelerometer. The node uses a Texas Instrument MSP430F1611 microcontroller (MCU) for on-node computation and a Roving Networks Bluetooth module. TEMPO has been used to for gait analysis and fall detection in medical laboratories. The power consumption of this node ranges from 3.92mW to 185mW, depending on which components are turned on.

The second is the Sensing Health with Intelligence, Modularity, Mobility and Experimental Reusability (SHIMMER) platform [13]. SHIMMER comprises of a baseboard which provides computational capabilities, data storage, communications, and a daughterboard socket to expand its functionality. It utilizes a Texas Instrument MSP430 MCU for processing and a microSD card for data storage. The platform uses a Chipcon CC2420 radio transceiver, gigaAnt 2.4 GHz Rufa antenna, and a Roving Networks Bluetooth module for wireless communication. The core functionality of SHIMMER is extended via a range of daughterboards which provide various kinematic, physiological, and ambient sensing capabilities. This platform has been

actively used for kinematic, physiological and ambient sensing applications. It can consume 4.4mW to 93mW when writing to a non-volatile memory or 61mW to 138mW when streaming over Bluetooth.

The third is a COTS EMG monitoring wireless BSN developed by Penders et al. [40]. The node uses a Texas Instruments MSP430 micro-controller for processing, a proprietary single channel chip for the analog front end, an energy harvester with 15 photovoltaic cells, and a Nordic nRF24L01+ radio. The node reduces power by removing its crystal oscillator, heavy duty cycling of components, and reducing its networking capabilities. The node is able to accomplish a 15-fold power reduction from 7.1mW to 450 $\mu$ W. However, this optimized power is still too high to be powered off energy harvesters and the authors claim that their node is still too big and bulky to be deployed. Further power reduction is needed to improve the lifetime of these BSN nodes.

### 3.1.2 ASIC

Literature in ASIC BSN nodes is limited. ASIC BSN chip design is still an emerging field, and there are few complete systems in the literature. One reason for this lack in nodes is that a complete BSN node requires optimized blocks that each requires expertise in multiple areas to develop efficiently [12]. We look at three state-of-the-art systems.

Verma et al [41] present a low-power SoC that performs EEG acquisition and feature extraction for continuous monitoring and detection of seizure onset in epilepsy patients. The SoC has one to 18 EEG channels that are worn to detect seizures. The SoC integrates an instrumentation amplifier, ADC, and digital processor that streams features-vectors to a central device where seizure detection is performed. This chip consumes 3.5 $\mu$ W for sampling and feature extraction of EEG signals. However, this chip is not a full system; it lacks an on-chip radio and supply regulation. Additionally, it does not have energy harvesting capabilities.

Chen et al [42] present an implantable  $0.5 \times 1.5 \times 2\text{mm}^3$  intraocular pressure sensor. This node is designed to provide feedback for glaucoma treatment, incorporates a solar cell, a pressure sensor, and a microbattery with a low-power SoC. The chip SoC consists of a wireless transceiver, a capacitance to digital converter, a DC-DC switched capacitor network, a microcontroller, and on-board memory. The node is capable of achieving an average power consumption of  $5.3\text{nW}$  through heavy duty cycling due to its sampling of pressure every 15 minutes. A transmitter sends  $1\text{b}$   $40\text{mW}$  bursts every  $131\mu\text{s}$ . With less than 10 hours of indoor light a day and measurements less than every 15 min, the node can run perpetually from harvested energy.

Shih et al [43] present a glucose detection SoC with a wireless transmitter that is integrated with a contact lens for temperature and pressure sensing. The system uses a capacitive MEMS pressure sensor for glaucoma detection. The chip converts both capacitance and temperature to frequency using a time-interleaved relaxation oscillator. The sensor is inductively powered by a reader held near the eye, communicates with a 2.4-GHz radio, and consumes approximately  $2.3\mu\text{W}$ . It utilizes a sub- $\mu\text{W}$  regulator and bandgap reference.

These nodes have provided tangible improvements in the field of BSNs. These works show that energy harvesting could provide a viable power supply for BSN circuits. However, integration of a complete wireless, flexible, easily deployable BSN node on a SoC that supports closed-loop power management and energy harvesting has yet to be demonstrated.

## 3.2 BSN Node

We utilize recent advances in energy harvesting, low voltage boost circuits, dynamic power management, subthreshold processing, biosignal front-ends, and low power RF transmitters to realize an integrated reconfigurable wireless BSN SoC for ECG, EMG, and EEG applications with autonomous power management for completely battery-free operation. This SoC can run

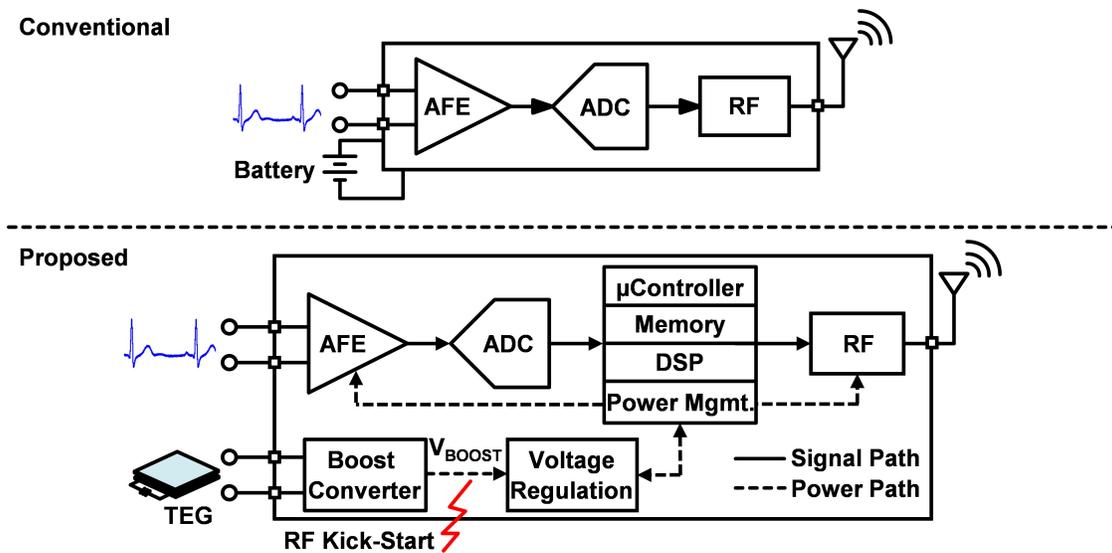


Figure 3.1: High-level diagram of conventional BSN solution (top) and the proposed (bottom) solution with energy harvesting, integrated power management, and ultra-low power flexible DSP architecture [2].

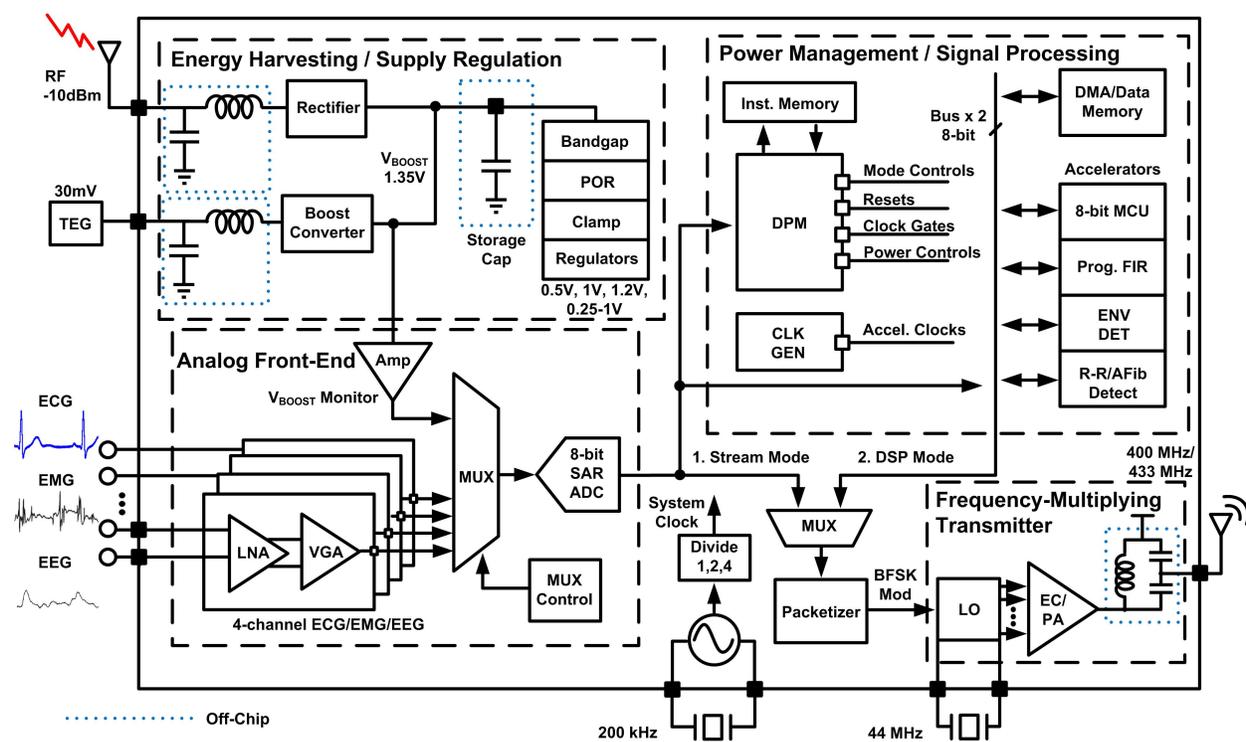


Figure 3.2: System block diagram for the proposed chip comprising the energy harvesting/supply regulation, analog front-end (AFE), subthreshold digital signal processing, and transmitter subsystems [2].

indefinitely from energy harvested from body heat while worn and decrease cost by having high integration and targeting a wide range of bio-electric sensing applications.

### 3.2.1 Architecture Overview

Conventional BSN nodes and wireless sensors use batteries (Figure 3.1), limiting node lifetime and reducing user compliance due to battery changes/charges. We propose a wireless BSN chip powered by energy harvested from human body heat using a thermoelectric generator (TEG), shown in Figure 3.1. This, in conjunction with ultra low power (ULP) circuits, intelligent duty cycling of power-hungry blocks (e.g. the transmitter), and a programmable power management system allows for indefinite operation of the chip. To demonstrate, we present a wireless BSN node that targets ExG (ECG, EEG, and EMG) applications.

To achieve flexible data acquisition and processing while operating the node solely from harvested energy, we propose a system architecture, illustrated in Figure 3.2, which comprises of four subsystems. First, the energy harvesting/supply regulation section boosts an energy harvested supply input voltage as low as 30mV up to a useable, regulated 1.35V on an off chip storage capacitor, provides five regulated voltage supplies to the rest of the chip, and generates a bandgap reference. Second, the four-channel AFE provides biosignal acquisition with a programmable gain and sampling rate, amplifying ExG signals as low as a few  $\mu\text{V}$ s while consuming  $<4\mu\text{W}/\text{channel}$ . A variable gain amplifier (VGA) maximizes the signal at the input to an 8-bit successive-approximation (SAR) analog to digital converter (ADC), thus, reducing the ADC resolution requirement. Third, the acquired data is sent to a subthreshold digital processing subsystem that also performs mode control and power management (including power/clockgating of blocks and fine-grained dynamic voltage scaling (DVS)) based on the available energy on the storage capacitor. The digital section includes a custom microcontroller (the digital power management (DPM)), general purpose microcontroller (MCU), programmable FIR filter, 1.5kB instruction SRAM/ROM, 4kB data memory FIFO, and dedicated accelerators for ECG heart rate (R-R) extraction,

atrial fibrillation (AFib) detection, and EEG band energy calculation. The DPM is responsible for power management, node control, data flow management, and overseeing all processing on-node. Finally, a sub-mW 400/433 MHz MICS/ISM band frequency-multiplying transmitter (TX) that is capable of transmission up to 200 kbps. The TX has low instantaneous power consumption to avoid the need of large filtering capacitors on the supplies and is intelligently duty-cycled to achieve low average power consumption, thus enabling this node to operate exclusively off harvested energy.

### 3.2.2 Energy Harvesting/ Supply Regulation Subsystem

The energy harvesting subsystem is designed to harvest energy from RF, thermoelectric, or solar power sources and to provide regulated voltages to the rest of the chip.

#### Harvesting Energy from a Thermoelectric Generator (TEG)

Thermoelectric generators (TEGs) are constructed of thermopiles in series. This arrangement places a constraint on the maximum voltage achievable from a given temperature difference for a given size. Assuming a temperature gradient of  $1^{\circ}\text{C}$ , a  $11\text{cm}^2$  TEG will generate much less than 1V. In addition, the surrounding air presents a large thermal resistance that dramatically reduces the effective temperature gradient across the thermopiles, further limiting the voltage available at the TEG output. To quantify how much power can be harvested, we placed a COTS TEG of  $44\text{cm}^2$  [44] on different parts of the human body. Figure 3.3 shows that this TEG can harvest approximately  $60\mu\text{W}$  at room temperature and  $200\mu\text{W}$  at  $6^{\circ}\text{C}$ , which would be adequate to power a low power BSN node. Though the selected TEG is too big for required form factor, it serves as a proof of concept for powering a BSN node off a TEG. However, the voltage available at the TEG output is only tens of mV and thus requires a high conversion ratio boost converter to generate a usable supply voltage.

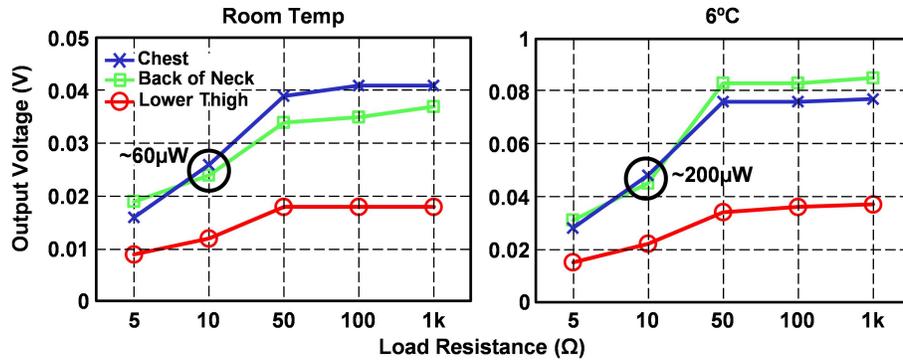


Figure 3.3: System block diagram for the proposed chip comprising the energy harvesting/supply regulation, analog front-end (AFE), subthreshold digital signal processing, and transmitter subsystems [7].

### Boost Converter and RF Kick-start

This work utilizes the boost converter architecture proposed in [45] to use the low voltages available from a body-worn TEG. Due to its increased efficiency at high conversion ratios and minimum usable input voltage (30mV in this work), the converter is well suited for harvesting the TEG energy input. The measured efficiency is 38% on our chip when converting from 30mV to 1.35V.

While the boost converter supports a 30mV power source, the internal oscillator and control logic need 600mV for startup. This requires a one-time pre-charge of the storage capacitor voltage,  $V_{\text{Boost}}$  (output of the boost converter). Previously reported start-up mechanisms use batteries and mechanical switches [45] [46], requiring bulky off-chip components. Instead, we use wireless RF power for the 600mV kick-start. Incident RF power as low as -10dBm is provided wirelessly for 1-2 seconds and rectified through an RF rectifier front-end consisting of a 6-stage charge pump. A shunt regulator clamps  $V_{\text{Boost}}$  to 1.35V to prevent over-voltage during the RF powering. A low-voltage bandgap-based power-on-reset (POR) resets the chip after  $V_{\text{Boost}}$  reaches 1V. Hysteresis in the POR trigger levels allows POR resetting only when  $V_{\text{Boost}}$  drops below a critical voltage where the chip fails to function correctly.  $V_{\text{KILL}}$  is determined by the minimum  $V_{\text{Boost}}$  voltage required to generate correct reference voltages and sustain conversion. Figure 3.4 shows a measurement of the RF kick-start. After the

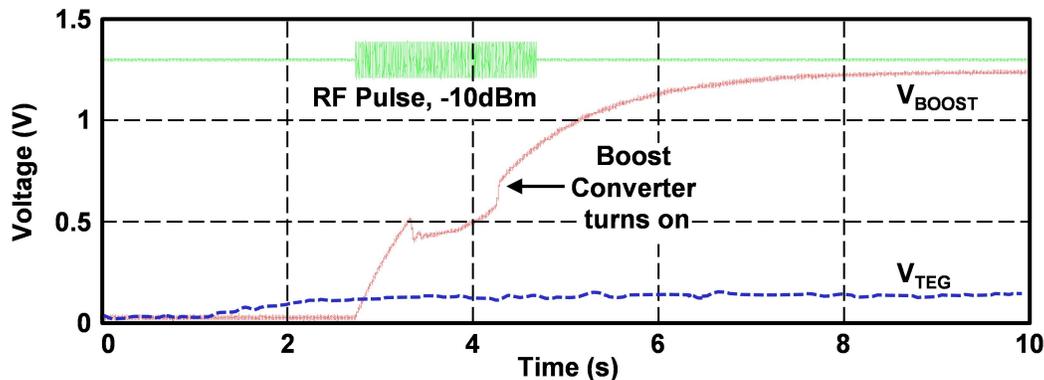


Figure 3.4: Measured startup sequence for the chip. A RF pulse kick-starts  $V_{\text{Boost}}$ , allowing the boost converter to turn on. The chip is able to sustain a usable voltage from harvested energy ( $V_{\text{TEG}}$ ) from then on [2].

voltage at the TEG output settles, a short RF burst wirelessly charges the storage capacitor at  $V_{\text{Boost}}$ . Shortly after the voltage reaches 600mV, the boost converter turns on and charges the capacitor to a regulated voltage. The node can then continue to run indefinitely from the TEG, unless  $V_{\text{Boost}}$  drops below  $V_{\text{KILL}}$  due to a prolonged period of consumption exceeding harvested energy.

### Chip Resuscitation

In the case that  $V_{\text{Boost}}$  decreases below  $V_{\text{KILL}}$ , the chip will blackout and will automatically shut-off. The contents of the instruction memory and data memory will be lost. The chip can be "revived" with the same startup sequence of RF powering and harvesting from the TEG as described previous. A 90 instruction (132 byte total) subthreshold ROM can re-boot the chip and execute a default AFib detection algorithm. The DPM is able to recognize that the chip had died through a latch structure that stores this information.

### Supply Regulation

The low voltage TEG input is boosted and subsequently regulated. Figure 3.5 details the supply regulation circuits. All biases are generated on-chip. On-chip supply regulation is provided by four sub- $\mu\text{W}$  linear regulators: 1.2V (AFE), 0.5V (digital), 1.0V (TX oscillator),

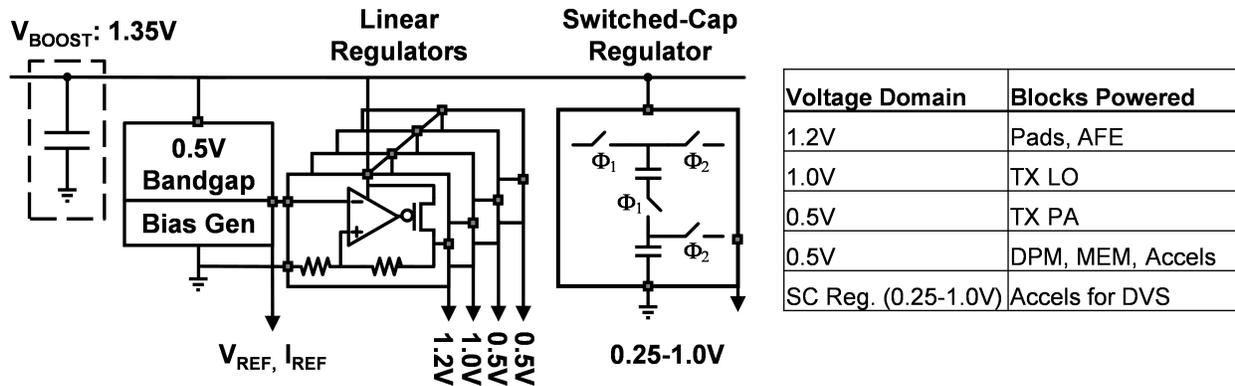


Figure 3.5: Measured results from an on-body thermal-harvesting experiment with 4x4 cm<sup>2</sup> COTS TEG [7].

and 0.5V (TX power amplifier (PA)). A fifth programmable switched-cap DC-DC converter provides an output from 0.25V to 1V in 50mV steps for the digital side. A 3-bit resistor DAC (RDAC) generates a reference for the desired output level based on a 3b control word from the DPM. The arrangement of the capacitors in the array varies according to the desired output range in a manner similar to [47]. An external capacitor ensures that the voltage ripple due to the switches switching is minimized. The subthreshold DSP accelerators can either be connected to the 0.5V supply or variable supply through PMOS headers, enabling DVS for additional power savings.

### 3.2.3 Closed-Loop Power Management

To prevent node blackout, the SoC must be aware of the available energy on the storage capacitance and adjust its mode of operation accordingly. If  $V_{\text{Boost}}$  is decreasing, the SoC must adapt its power consumption to consume less power and energy. When harvested energy is abundant again, the chip should recover itself to a mode of full operation. The always-on DPM (digital power manager) is a custom microcontroller that implements a closed-loop power management scheme. This power management is detailed in Chapter 4. The DPM is also responsible for node control, data flow management, and overseeing all processing on the node, as described in Section 3.2.4.

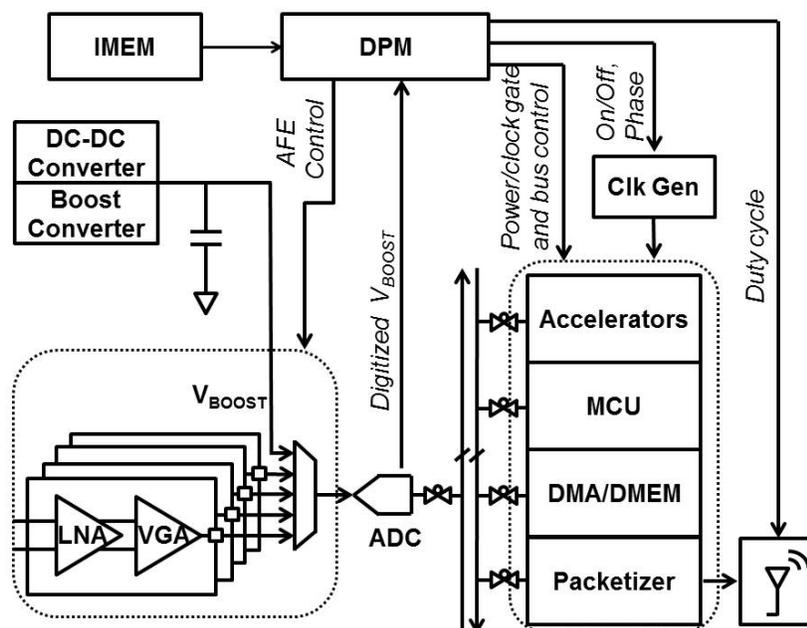


Figure 3.6: The DPM is responsible for power management, node control, data flow management, and overseeing all on-node processing [8].

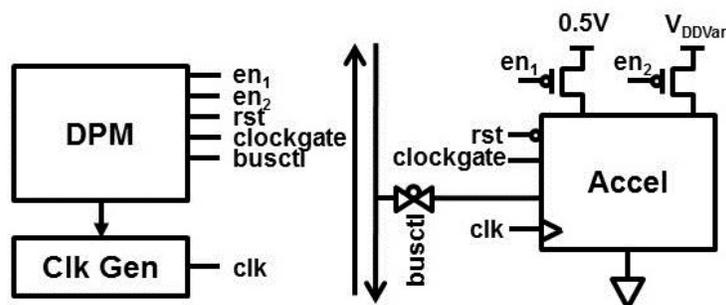


Figure 3.7: Controls from the DPM to a generic accelerator block [8].

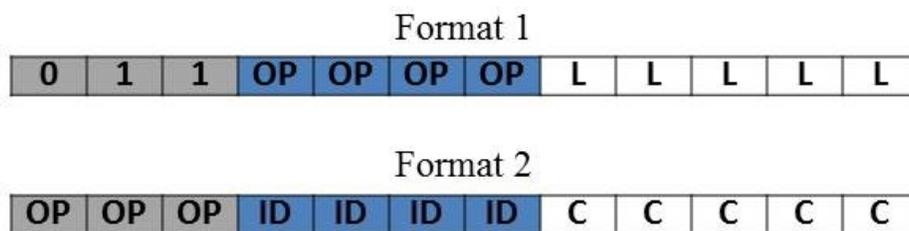


Figure 3.8: DPM instruction formats [8].

### 3.2.4 Flexible Biosignal Datapath

## DPM

The DPM is a light-weight, always-on custom microcontroller that has been designed to meet to the functionality required by this BSN node. In addition to the DPM's power management responsibility that will be described in Chapter 4, the DPM is responsible for node control, data flow management, and overseeing all processing on this batteryless BSN node. Note that the DPM has no computational abilities; this node relies solely on BSN accelerators described in Section 3.2.4 for computation. This is a departure from traditional schemes in which a generic microcontroller is responsible for node control, data flow management, power management, and processing. Entrusting a large amount of responsibilities to the microcontroller may require over-clocking the processor or running at a higher voltage and, thus, consumes more energy.

The DPM controls the data memory (DMEM), input channels of the AFE and ADC, sampling rate, transmission rate, clock frequency creation and distribution to accelerator blocks, bus management for flexible and timing-defined data flow, time delays, and clockgating and DVS voltage of the digital blocks, as summarized in Figure 3.6. Control bits to each block are held in 95 output registers within the DPM. These signals are buffered and routed to their respective blocks, not requiring individual decoders for each block.

**DPM Instruction Set Architecture** The DPM executes arbitrary instructions from the 1.5kB instruction memory. The DPM decodes a 12b instruction word issued by the instruction memory. Instruction words have two formats, as shown in Figure 3.8. The first 3b are an opcode. In format 1, the next four bits serve as a block ID number. Block ID numbers are unique to each block. For example, the ID number 1011 refers to the AFE. The rest of the word is control bits specific to the block, such as selecting voltage and resetting the block. This format allows for easy extension of the ISA to support new blocks. Opcode 011 uses format 2, which effectively becomes an instruction with a 7b opcode. The remaining bits are used as literals to support the instruction.

Table 3.1 shows the DPM’s custom instruction set architecture (ISA), which is designed to facilitate node management.

Table 3.1: DPM Instruction Set Architecture

Code	Description
NOP	No operation
STALL	NOPs until new ADC sample available
TIMER	NOPs for a set number of cycles
EN	Enable/disables, sets voltage of, clockgating, and resets block
DMA	Sets DMA control to read or write
ADCCHAN	Selects ADC channel input
CTRL	Sets flag for intended destination of instruction
SAVEPC	Stores current PC values
RESTOREPC	Restores PC value from the last saved PC value
SETVAR	Sets voltage of variable voltage
BUS1/BUS2	Controls connections to the respective bus
SETCLK	Sets clock to accelerators, DMA
CJMP	Conditional jump
JMP	Absolute jump to literal value

**Boot Sequence** Energy harvesting conditions may change at any time, potentially resulting in node death if the consumed energy exceeds the stored energy faster than harvesting replenishes it. If node death occurs, this BSN node is capable of being revived in the field through a short RF burst. In this case, the volatile instruction memory is invalid and instructions should be run from the ROM. On startup, after the storage capacitor reaches a sufficient voltage, the DPM checks a flag, which is set to 1 when programmed and 0 when rebooted. The DPM sets the program counter (PC) to point to either the instruction ROM or RAM based on the flag through use of the conditional jump (CJMP) instruction.

**Block Control** The DPM is responsible for managing the AFE, MCU, DMEM, accelerator blocks, and transmitter through the EN instruction, allowing for fine-grained power control. The EN command is common in our BSN programs and provides an energy efficient alternative to a two clock cycle generic microcontroller equivalent. The DPM provides control signals to

the block as show in Figure 3.7. The DPM is capable of duty cycling and power gating power hungry blocks such as individual AFE channels, the transmitter, and its crystal oscillator, resulting in large energy savings.

To save dynamic and leakage energy, the DPM can clockgate the whole data memory and individually power gate each bank by asserting bits to the NMOS footer on each bank. Individual accelerators can be clockgated and power gated. Additionally, accelerator blocks are capable of running on a 0.5V supply or a programmable voltage supply. The DPM is capable of setting the voltage of a programmable, variable DC-DC converter ( $VDD_{\text{Var}}$ ) capable of providing 0.3V-1.2V by sending four control bits to the voltage regulator through the SETVAR instruction.  $VDD_{\text{Var}}$  is distributed to each accelerator, allowing blocks to utilize dynamic voltage scaling.

Most accelerators do not require fast clock speeds, such as the 200 kHz system clock, for processing. To reduce energy, the DPM issues control bits through the SETCLK instruction to the clock generator block to turn on the accelerator clock. The clock division is programmed within the scan chains.

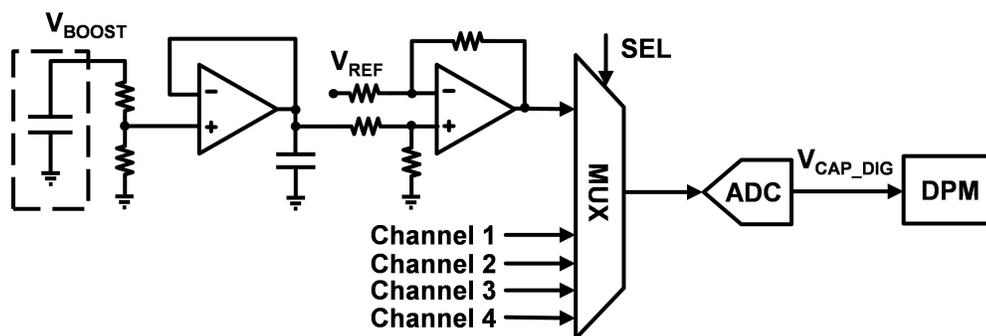
**Flexible Datapath** BUS1 and BUS2 instructions provide flexibility to move data through the ADC, accelerators, the MCU, the DMEM, and the transmitter through executing DPM instructions. The node has two circuit switched buses that connect blocks, shown in Table 3.2, to move data. The DPM issues bits to the transmission gates that connect each block's inputs and outputs to each bus. The DPM ensures that only two blocks are connected on each bus at one time with an exclusive hardware lock.

### Analog Front End (AFE)

The chip has four independently selectable biosignal input channels, each with a fully-differential chopper-stabilized low-noise amplifier (LNA) and variable-gain amplifier (VGA) [48]. We chose 20 kHz as the chopper clock frequency to be significantly higher than the

Table 3.2: Connections for Bus 1 and Bus 2

Available Inputs	Available Output Destinations
MCU	MCU
FIR(x4)	FIR
DMA/DMEM	DMA/DMEM
Envelope Detector	Envelope Detector
ADC	Packetizer
	R-R Extraction/AFib Detection

Figure 3.9: Block diagram of the AFE, including the path for monitoring  $V_{CAP}$  [7].

flicker-noise corner of the OTA for effective flicker-noise reduction. Input chopper switches are placed before the input capacitors. Compared to a topology where the switches come after the input capacitors, our arrangement reduces the amplification of any OTA offsets that might saturate its output. Any mismatch in the input capacitors results in common-mode to differential-mode gain. Since this amplifier is effectively AC-coupled, an off-chip capacitor and resistor are used to block any DC offset voltage at the electrode interface. One drawback of the topology is that the input impedance is relatively low. Periodic steady-state simulations reveal that the input impedance is a few  $M\Omega$ s, sufficient for our ECG, EMG, and EEG applications. A programmable Gm-C filter reduces the switching ripple to below the noise floor. Along with the VGA, our amplifiers provide 7-step digitally-programmable gain (40-78dB) from DC to 320 Hz at  $3\mu\text{W}/\text{channel}$ . A 5-input mux allows the sub- $\mu\text{W}$  8-bit SAR ADC to sample any of the four channels as well as the  $V_{Boost}$  node for monitoring stored energy (Figure 3.9).

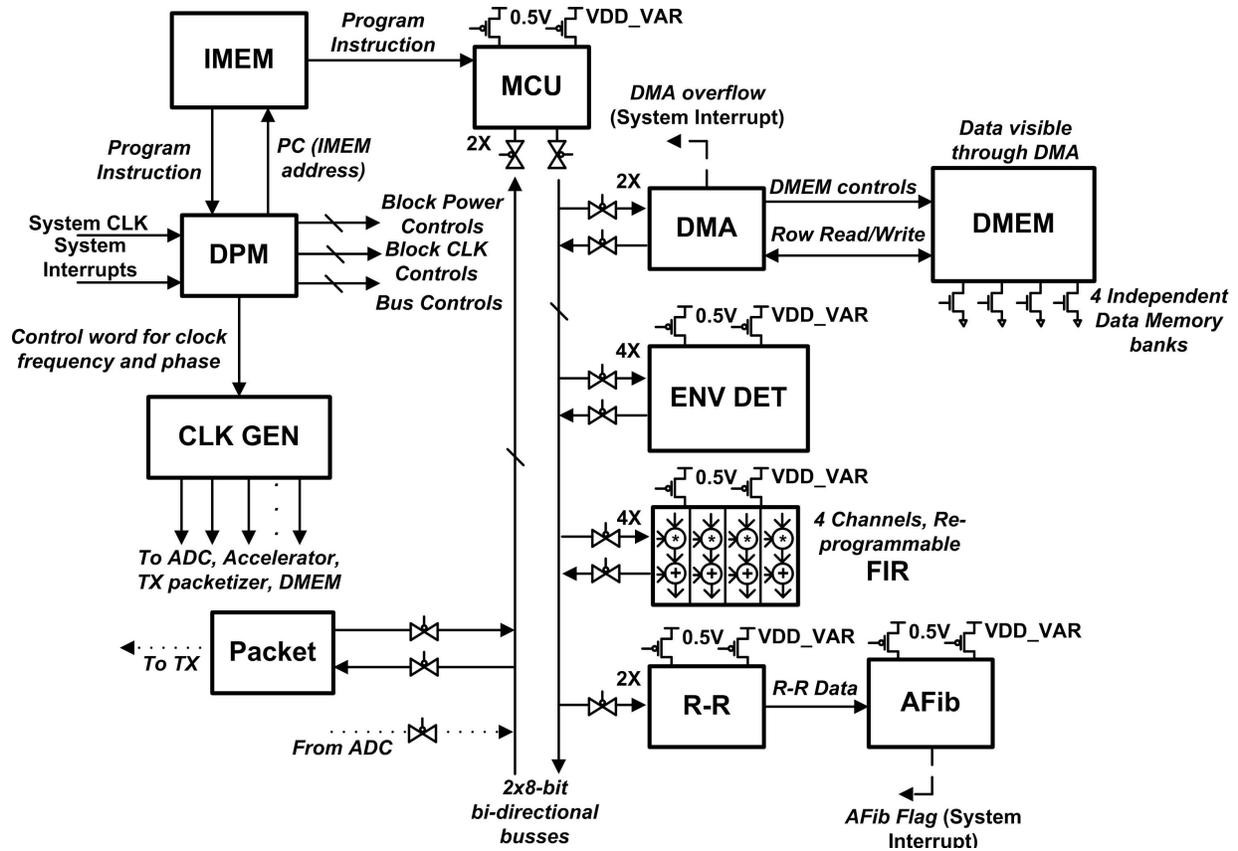


Figure 3.10: Block diagram of subthreshold data processing subsystem [7].

### Subthreshold Digital Signal Processing Subsystem

Figure 3.10 shows the subthreshold DSP subsystem. Since we have clearly defined applications, we implement ASIC accelerators for heart rate extraction (R-R), atrial fibrillation detection (AFib), and energy band extraction/envelope detection (ENV DET) to improve energy efficiency over processing in the MCU. An 8b RISC ISA microcontroller executes generic computations that the accelerators are not capable of executing to provide more flexibility to this node, and a re-programmable FIR does digital filtering. A digital packetizer streams serial data to the TX. Two memory arrays store the program (Instruction Memory, IMEM) and biosignal data (Data Memory, DMEM). A DMA achieves easy FIFO control and low memory latency for the DMEM.

Two 8b switch-box buses connect the inputs and outputs of all the processing accelerators,

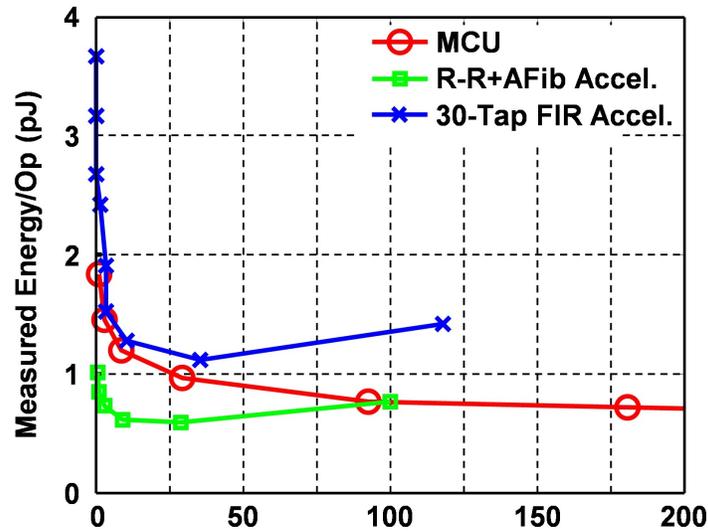


Figure 3.11: Energy-delay curves for MCU, RR+AFib accelerator, and 30-Tap, 1-Channel FIR [2].

MCU, DMA, and packetizer. The DPM issues bits to the transmission gates that connect each block's inputs and outputs to each bus. Each input/output bus port has a 4-bit address that is decoded in the DPM. Having two buses eases data steering and simplifies the control instructions.

To support the power management scheme described in Chapter 4, each processing element has a clock-gate and two PMOS headers [33], one connected to 0.5 V, the other to the variable voltage for DVS. The clock generator block (CLK GEN) distributes a programmable clock signal (frequency and phase) to each of the accelerator blocks. The chip can process data flexibly with the MCU, use energy efficient accelerators, or use a combination of accelerators and the MCU for processing. The chip can also transmit flexibly by either streaming data without processing or storing, storing and bursting data, or doing event-based transmission. The main components work as follows:

1) MCU: The 8b microcontroller is a subthreshold RISC based on the PIC series [49]. The MCU is capable of executing arbitrary programs and functions down to 0.26V, 1.2 kHz. Figure 3.11 shows the energy-delay (E-D) curve for the MCU. The MCU consumes 0.7nW to 1.4 $\mu$ W measured power (0.26-0.55V) and 1.5pJ/op at the default 0.5V, 200 kHz setting. The

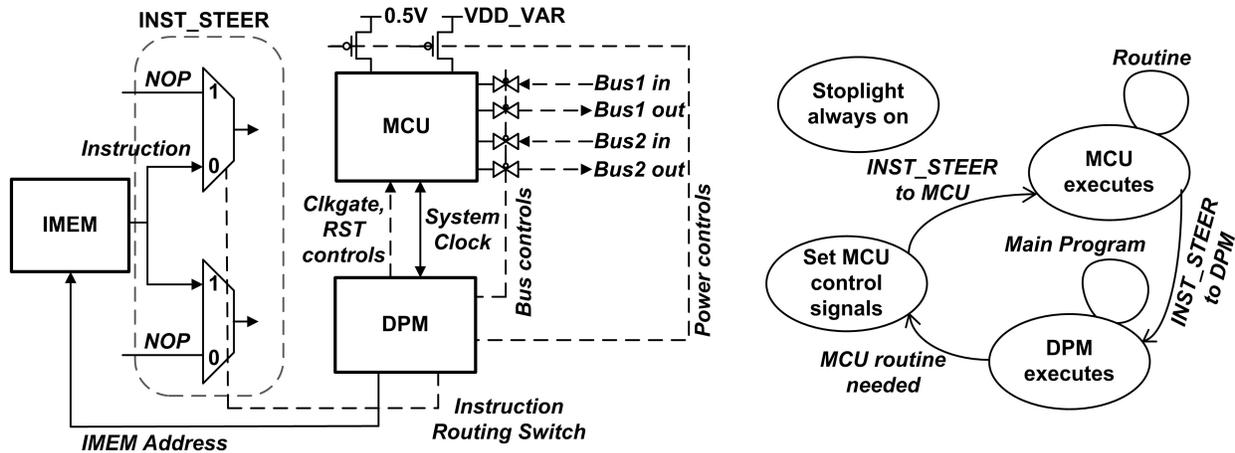


Figure 3.12: Integration of DPM and GPP MCU. DPM and MCU share the same memory, and instructions are steered to the right processor. The MCU can be power-gated when idle. In this way, both efficient control of the node and generic processing are implemented [7].

MCU shares the IMEM with the DPM. Figure 3.12 summarizes the organization of memory, MCU, and DPM, and their capabilities. A multiplexer steers each instruction to either the MCU or DPM (INST\_STEER) based on a special instruction issued by the DPM. When the MCU is executing instructions, the DPM automatically goes into a low power sleep mode. When the DPM is executing instructions, the MCU is either turned off or clock gated to save state. We retain the energy efficiency of the DPM as a chip controller and the generic flexibility of the MCU without requiring an additional memory.

2) Instruction and Data Memories: Because the chip operates in subthreshold in an N-strong technology, the SRAMs use an 8T bitcell and the zero leakage read-buffer from [50]. To eliminate half-select instability during a write, we read and write the full rows of memory. The IMEM interfaces to the DPM; the DPM will read the full row of memory and parse the correct instruction. The DPM also implements a burst mode read, thus reducing the average energy of the IMEM. The DMEM interfaces to the DMA. The DMEM is split into 4 1kB banks that can be individually power gated by NMOS footers being overdriven to 1.2V when active to ensure low levels of ground bounce. Measured results show reliable operation down to 0.3V at 200 kHz with IMEM read energy of 12.1pJ per at 0.5V and leakage energy per cycle at 200 kHz of 6.6pJ.

3) DMA: The DMA is an efficient subthreshold accelerator to interface between the DMEM and the rest of the SoC. It is easily programmed by one instruction of the DPM and effectively treats the DMEM as a FIFO to support efficient streaming. The CLK MUX for the DMA synchronizes the DMA clock rate to the component it interfaces to. The MEM controller uses separate DMEM banks during green and yellow mode for easier data management. To solve the half-select stability issue during writes, we use a row buffer and only write a row when all words are ready. When the difference between the write pointer address and read pointer address is greater than or equal to 4 bytes, the DMA\_flag is raised, which signifies to the DPM that there is a full packet of data. This simple and efficient mechanism of interrupting for transmission limits overflows.

4) Programmable FIR Filter: A four-channel (to support the four-channel AFE), programmable, max-30 tap, and synthesizable filter was designed to operate in the subthreshold regime down to 300mV (measured). The programmable options include coefficient selection, number of taps, and number of filters. When power is critical and data fidelity can be compromised, a half-taps mode allows for a 15-tap filter. The direct-form implementation of an FIR requires as many adders and multipliers as there are taps, costing area and leakage. Due to the small sampling rate for ExG signals, each result can instead be computed serially over multiple faster clock cycles using only one multiplier and one adder. This architecture results in a 30x reduction in area per channel and a measured 1.1pJ per tap at 350mV. For further power reduction, each individual channel can be power and clock-gated. A measured E-D curve is given in Figure 3.11.

5) Envelope Detector: For EEG signals, knowing signal power within a specific frequency band is useful for determining neural activity in the  $\alpha$ ,  $\beta$ ,  $\gamma$ , and low- $\gamma$  frequency bands [51]. The ENV DET circuit computes the average signal power within a specified frequency band. This block receives data directly from the FIR filter and has four input channels corresponding to the channel outputs of the filter. To reduce the computation complexity, intermediate operands were rounded to the nearest power of 4 and the square results come from a lookup

table. The rounding reduces the number of bits required during data transformation as the lower two bits are always 0. The ENV DET consumes 3.5nW (measured) at 0.5V and 200 kHz.

6) R-R Extraction: The heart rate extractor accelerator is a simple version of the Pan-Tomkins algorithm [52]. This R-R algorithm calculates the heart rate by means of time windowing and thresholding, after an initial 4 second time frame where the R-R accelerator gains a baseline DC value for the heart waveform. The time stamp given to two consecutive peaks is the difference in the number of samples between them. For this reason, we can achieve a desired accuracy by changing the sampling rate and using the R-R accelerator in a DVS fashion to accommodate the faster or slower processing rate needed. Once an R-R time has been calculated, a pulse is output, which signifies to the AFib accelerator a new R-R sample is ready.

7) AFib Accelerator: The atrial fibrillation detector is an ASIC accelerator that detects the cardiac arrhythmia using an implementation of the clinically validated algorithm described in [53]. It receives its inputs from the R-R accelerator and outputs an AFib\_flag signal to the DPM signifying the detection of atrial fibrillation. The algorithm uses only 12 R-R intervals for detection [53]. Many variables in the algorithm, such as the margin of error, are programmable to adapt for different patients. The algorithm uses a pattern recognition scheme that quantifies the entropy in these 12 R-R intervals. If the entropy is more than the programmed threshold, then an AFib event is reported.

### **Low Power RF Transmitter**

To allow operation from harvested power, the peak current consumption of the chip must be minimized. We utilize a frequency multiplying transmitter architecture to reduce the synthesizer power by operating the local oscillator at 1/9 the carrier frequency. We use equally spaced edges generated from the cascaded ring oscillators to drive the edge-combiner (EC) embedded PA to perform the frequency multiplication.

The use of frequency multiplication allows harmonic injection-locking from the crystal oscillator. Instead of using a PLL, injection-locking a low-frequency ring oscillator to an on-chip crystal reference eliminates the longer settling times, therefore allowing aggressive duty-cycling of the transmitter to further save power. Directly injection-locking the multi-phase ring oscillator using the single-phase reference introduces significant mismatch. We used cascaded multi-phase injection-locking to correct the phase and amplitude mismatches.

On-chip BFSK modulation is accomplished by pulling the quartz reference clock. By modulating the load capacitor, we can pull the crystal frequency by 200ppm. After 9x multiplication, the resulting frequency deviation is approximately 100 kHz, achieving >100kbps data rate. The TX circuit consumes  $160\mu\text{W}$  when transmitting at its maximum data rate of 200kbps [54]. In biosignal raw data mode, the transmitter operates at a 100% duty-cycle, while the R-R extraction mode, explained in the next section, reduces the duty cycle and average transmitter power consumption to 0.013% and 190nW respectively. The packetizer contains a programmable packet header and CRC to allow compatibility with commercial receivers.

### 3.2.5 System Measurements

An ECG experiment was performed on a healthy human subject. First, the chip was set to ECG raw data mode (consuming  $397\mu\text{W}$  from the 1.35V  $V_{\text{Boost}}$  node) (Figure 3.13). Our chip was paired to an unmodified TI CC1101 receiver and a wireless link was successfully established in the 433 MHz ISM band. The reconstructed ECG (dashed) closely matched the actual ECG. Next, the chip ran an R-R interval extraction algorithm on the MCU and transmitted measured heart-rate every 5s operating from a 30mV supply voltage (Figure 3.14). Every 5s,  $V_{\text{Boost}}$  is sampled to check for sufficient available energy, in which case the crystal oscillator is enabled for 20ms before the TX transmission, which takes  $650\mu\text{s}$  including turn-on time and transmitting a 24-bit packet. The heart-rate extractor algorithm measures the R-R interval with a time resolution of  $(1/128)\text{s}$  (Figure 3.14). In AFib detection mode,

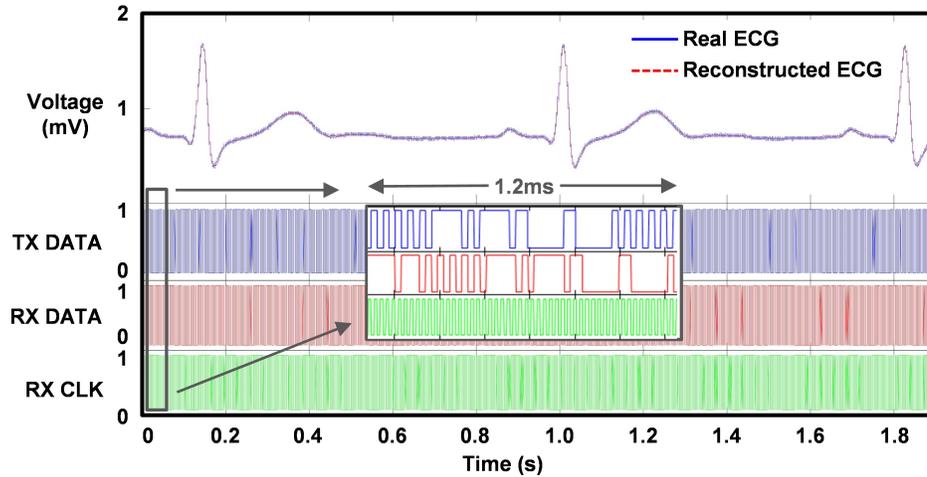


Figure 3.13: Measured system experiment showing correct data acquisition and streaming from the transmitter. Total power of  $397\mu\text{W}$  prevents long use of this mode from harvested power [2].

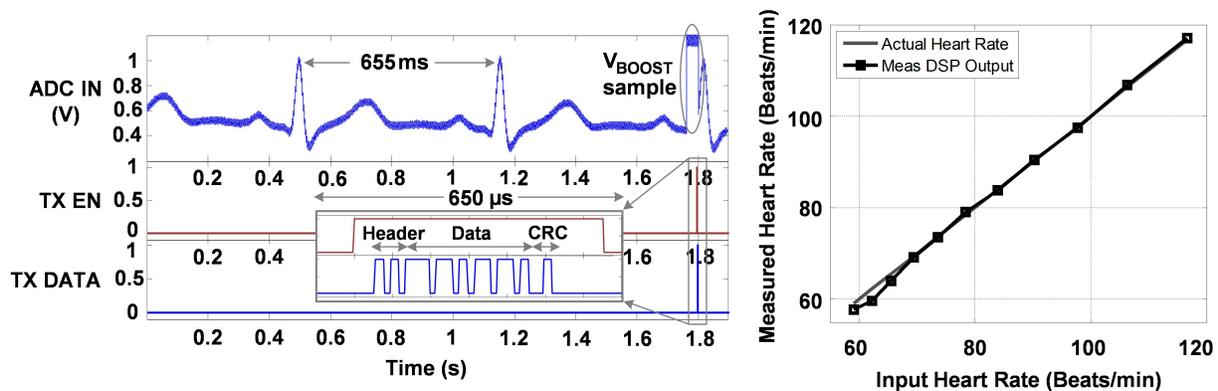


Figure 3.14: Measured system results for an R-R extraction algorithm. Measured system results are for acquiring ECG, extracting R-R intervals, and sending RF updates for R-R every 5s. Total power in this mode is  $19\mu\text{W}$  drawn from a 30mV input. Measured accuracy results for R-R also shown [2].

the R-R and AFib accelerators enable the TX and transmit the last 8 beats of raw ECG (buffered in the DMEM) only when a rare AFib event occurs. Measurement results for the AFib demo are presented in Figure 3.15. The total chip power in both the R-R and AFib modes is  $19\mu\text{W}$ , and the chip is powered exclusively from a 30mV harvested input.

Figure 3.16 presents a current breakdown of the R-R extraction demo. The current is nearly evenly distributed among different components, and selective transmission significantly reduces the average power consumption of the transmitter. Figure 3.17 shows the micrograph of the

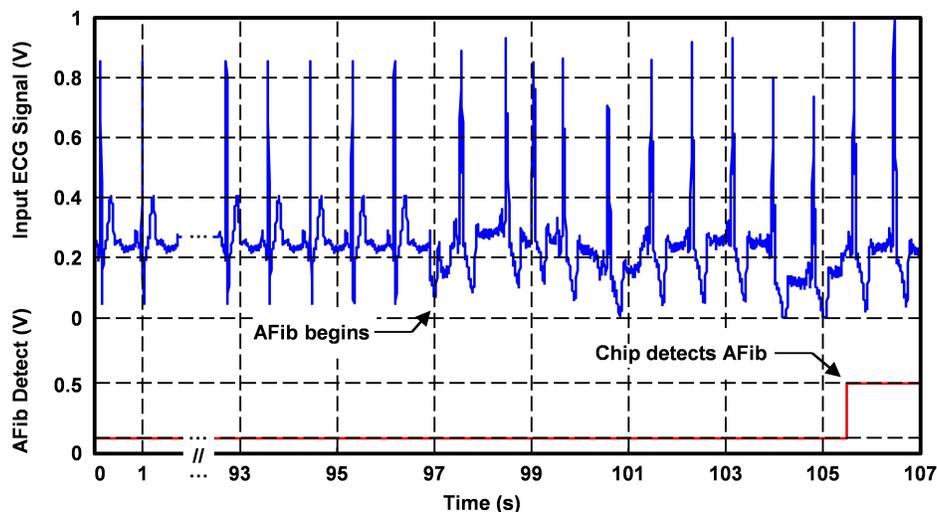


Figure 3.15: Measured system AFib demo experiment using R-R extractor and AFib accelerator. Normal and atrial fibrillation heart waveforms from MIT-BIH database [9]. Last 8 beats of raw ECG are stored in DMEM and streamed over TX if AFib is detected. Total chip power in this mode is  $19\mu\text{W}$  from a  $30\text{mV}$  input [7].

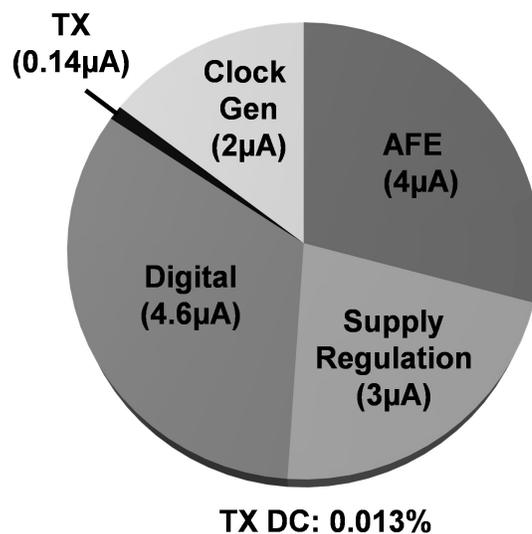


Figure 3.16: Current breakdown for R-R extraction experiment. The current distribution is roughly evenly distributed amongst main contributors, and the originally power-hungry transmitter consumption is now nearly mitigated [7].

2.5mm x 3.3mm batteryless BSN SoC (130nm CMOS), and Table 3.3 gives a performance summary.

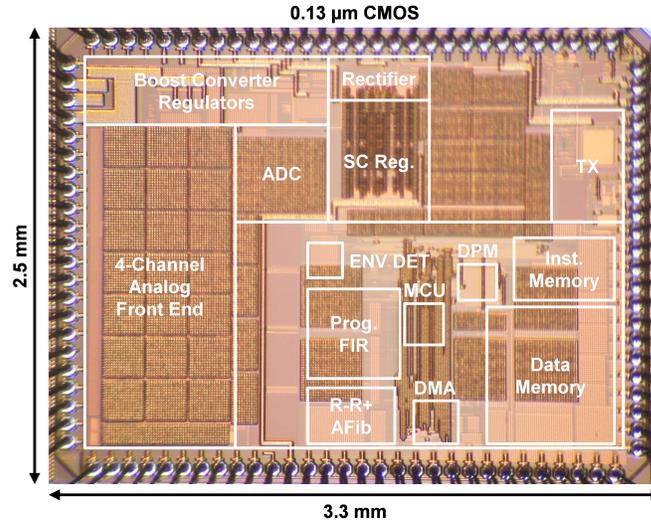


Figure 3.17: Annotated chip die photo [2].

Table 3.3: Measured Performance Summary [2]

Energy Harvesting		Supply Regulation		AFE (1 CH, ADC)		DSP		TX	
$V_{in}$	30mV	$V_{unreq}$	>1.25V	Current	4 $\mu$ A	Op. range	0.3-1.2V	Current	280 $\mu$ A
Kick-Start	RF@ -10dBm	$I_{quiescent}$	3 $\mu$ A	Supply	1.2V	MCU E/op @ 0.5V	1.5pJ	Supply	1V (LO) 0.5V (PA)
$V_{out}$	1.35V	$V_{analog,digital}$	0.5V, 1V, 1.2V	Gain	40-78dB	IMEM E/rd/Inst.	1.0pJ	Data-rate	200kbps
Efficiency	38%	$V_{PA}$	0.3-0.5V	$V_{ni,rms}$	<2 $\mu$ V <sub>rms</sub>	FIR FOM*	0.27	E/b	0.8nJ/b
<b>System Power</b>				Bandwidth	0-320 Hz	AFib E/sample @ 0.5V	6pJ	Output Power	-18.5dBm
Current	14 $\mu$ A (R-R) (0.013% TX duty cycle)	294 $\mu$ A (Stream) (100% TX)		CMRR	>70dB	ENV DET E/sample	0.53pJ	Band	400 MHz MICS 433 MHz ISM
Supply	1.35V					Sub- $V_T$ DVS: 0.3V to 0.6V	2 kHz- 1.7 MHz	Modulation	BFSK
FIR FOM*: Power (nW) / frequency (MHz) / # of taps / input bit length / coefficient bit length									

### 3.3 BSN Architectural Explorations

Our batteryless BSN node is a single point in a vast BSN node design space and cannot meet the requirements of all BSN applications. We can use this BSN node to explore digital tradeoffs in batteryless BSN node architectures that utilize BSN accelerators for processing to inform future revisions with different application requirements.

In our BSN node, the digital section is responsible for power management of all blocks, ensuring the node is active, processing biosignal information, storing information, and interfacing between blocks. Though digital power consumes only 34% of the overall power

(Figure 3.16), improvements in clocking [55] [56], power distribution [57], reduction in SRAM leakage, and the ability to duty cycle high power blocks more efficiently will result in the digital power becoming a more significant portion of the overall node power consumption. Table 3.4 shows the details of our projections of a similar node using the improvements listed. We estimate the digital power to consume 39% of the total power of a future revision of this node.

Table 3.4: Projected BSN Power Using State-of-the-Art Components

	Current Revision		Projected Values	
	Current ( $\mu\text{A}$ )	Percentage	Current ( $\mu\text{A}$ )	Percentage
Digital	4.6	33%	3.2	39%
AFE	4.0	29%	4.0	48%
Supply Reg	3.0	22%	1.0	12%
Clock Gen	2.0	15%	0.01	1%
TX	0.14	1%	0.14	2%
	13.74		8.37	

This section explores two key architectural design decisions in accelerator-based systems: selecting between a generic microcontroller and a custom microcontroller and selecting between a local decoder and global decoder bus scheme. Our aim is to understand the advantages and disadvantages of each to ultimately improve lifetime in batteryless, self-powered BSN nodes. These two components account consume the second largest amount of active power within the digital portion of our node, second only to the power consumed by the memories.

### 3.3.1 Microcontrollers

Table 3.5: MCU vs Accelerators: Energy Efficiency per Sample

	MCU	Accel	Savings
30-Tap FIR	6.3nJ	57.6pJ	110x
Envelope Detection	3.6nJ	530fJ	6800x
R-R Extraction	12pJ	3fJ	4000x

The microcontroller in an accelerator-based system is responsible for node control, data flow management, and overseeing all processing. Relying on the microcontroller to process data is energy inefficient. Processing within the microcontroller is up to 6800x more energy inefficient per sample than the accelerators utilized on this node (Table 3.5) within the fabricated node [2].

With these responsibilities in mind, we explore the benefits of using of a custom microcontroller versus a generic microcontroller, such as the openMSP430 [58] or the PIC processor [49], in terms of power/energy and capabilities. In our node, the DPM serves as the microcontroller and consumes approximately 26% of the digital power and 9% of the overall node power. The leakage of on-chip memories accounts for a large portion of digital power. Reduction of this leakage current will increase the DPM's portion of the digital power, making the selection of the microcontroller more critical in reducing node power and energy consumption to improve node lifetime.

For this case study, we use the DPM, described in Section 3.2.4, as an example of a custom microcontroller and the MCU based on the PIC ISA in the chip, described in Section 3.2.4, as an example of a generic microcontroller. We evaluate each on the following metrics: programmability, capability, and energy/power efficiency. Both energy and power are important in energy harvesting-powered nodes due to the power being sourced from the energy harvester and the energy being sourced from the storage capacitor. Minimizing power will result in more energy being stored on the capacitor or the ability for more functionality on nodes. Better energy efficiency will result in being able to run more operations off the capacitor when the energy harvester is sourcing less power than the node is consumed.

### **Programmability**

While programmability is not specifically correlated to node lifetime, ease of programming is important metric for choosing a microcontroller. The DPM utilizes a custom ISA, based on the minimal functionality required for the node. The ISA is specified in Table 3.1. To

program the DPM, a programmer must write assembly code or binary. A custom assembler and compiler are only available if a custom program is written. The MCU, however, has a tool flow available to translate C into assembly and binary. While this is advantageous for being able to write and synthesize code faster. However, writing high level code poses some risk; you rely on the compiler to translate the high level language into assembly efficiently and assign operations to accelerator blocks. If the compiler is unable to do so, it results in code bloat, requiring more instructions and clock cycles to run the operation. The topic of compilers is outside the scope of this experiment and this dissertation.

### Capability

Custom microcontrollers are coded with the bare minimum functionality in order to reduce energy and power profile. The DPM is capable of power management, node control, data flow management, and overseeing all processing on the node in parallel. However, the DPM is not capable of any data processing. It relies solely on the BSN accelerators to do biosignal processing due to the accelerator's energy efficiency compared to a microcontroller. Through utilizing the MCU's IO ports and ISA, the MCU is capable of the same functionality as the DPM. The MCU utilizes multiple simple instructions to achieve the same DPM instruction as shown Table 3.6. The MCU adds the additional capability of processing, though it is energy inefficient compared to accelerators. This capability can be leveraged to use algorithms that accelerators cannot execute, providing more flexibility to the node.

### Power/Energy Consumption

The DPM provides many delay and energy advantages over the MCU. The DPM measured average energy per operation on the 130nm chip at 0.5V is 2.74pJ. The DPM NOP is 0.68pJ/cycle at 0.5V. Figure 3.18a shows a comparison of measured energy-delay curves for the DPM and the MCU for several instructions. Though the MCU's single cycle energy is 1.45pJ at 0.5V, the MCU requires at least two instructions to run DPM equivalent operations

Table 3.6: MCU Equivalent Instructions

Code	MCU Eq
NOP	1
STALL	1
TIMER	1
EN	2
DMA	2
ADCCHAN	2
CTRL	2
SAVEPC	4
RESTOREPC	4
SETVAR	2
BUS1/BUS2	6
SETCLK	2
CJMP	4
JMP	4

(see Table 3.6). The MCU requires six instructions to produce a DPM equivalent BUS1 instruction. The DPM energy measurement includes the overhead of interfacing to the memory and the power management. Additionally, the MCU's need for multiple instructions could increase the amount of memory needed to run arbitrary instructions and, thus, power/energy consumption of the memories.

However, the MCU shows power advantage over the DPM (Figure 3.18b) for the average operation due to the nature of its simple instructions. At 0.5V, the measured average PIC operation power is 947nA. The DPM NOP and average measured power consumes 78% and 253% that amount, respectively.

### Case Study

We explore two common use case scenarios for our chip: power management and RR extraction & AFib detection (RR/AFib). These case studies are used to compare the code size (number of instruction words required to complete the operation), number of clock cycles need to complete the operation, and microcontroller energy for each scenario for this accelerator-based

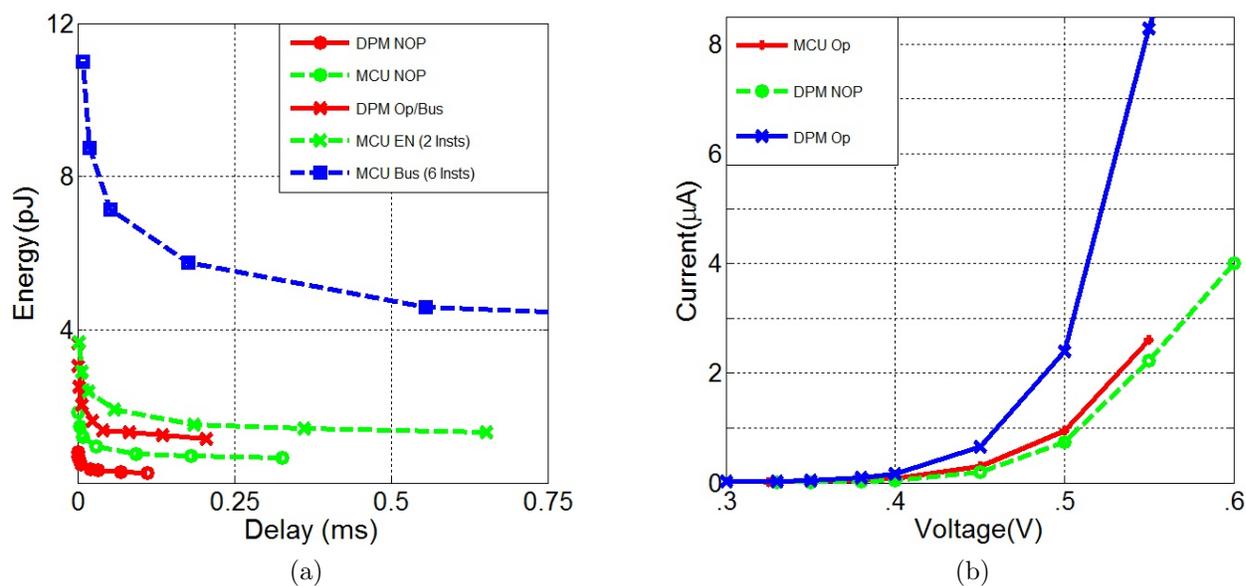


Figure 3.18: a. Measured energy-delay curve for the DPM and MCU for NOPs, ENs, and BUS instructions. b. Measured power consumption of the DPM and MCU

system. This energy is calculated from the measured energy per operation. The comparison is made between the DPM, the implemented MCU, and a more ideal MCU. The on-chip implementation of the MCU does not have the ability to stall efficiently. For a fair comparison, we created an energy efficient stall operation for the MCU. The MCU with this stall capability is called "the ideal MCU". The energy and power for this stall operation were extrapolated from the ratio of DPM NOP to DPM operation. We assume that this stall operation would be issued in a single clock cycle and remain until some condition was achieved, thus reducing code bloating.

In the power management case study, we focus on resolving the power management's operating mode (a state which sets the upper limit of node power consumption). The bus is configured to transfer the ADC value (the digitize voltage of the capacitor which corresponds to the amount of energy on the capacitor) to the respective MCU, compare the value to a maximum of two 8b programmed threshold, select the operating mode, and then write the override signals for each operating mode to a register. These override bits are used to override high power blocks such as the TX, AFE, and DMEM when harvesting conditions are

less than ideal. The power management scheme is explained in greater detail in Chapter 4.

In the RR/AFib case study, we focus on calculating the RR interval and doing AFib detection. We break the use case into three parts: setup, per sample, and radio setup. In the "setup" case, the DMEM is enabled, the clock to the RR/AFib accelerator is set up, the AFE/ADC is configured, the buses are setup, and the DMA is setup. In the "per sample" segment, the microcontroller is woken up, it checks the status of the RR/AFib block to see if there was an AFib event detected and then returns to its stall state. In the "radio setup" case, the microcontroller checks the status of the blocks, does a conditional jump, turns on the radio, configures the DMA to read the data, and sets the bus for transfer from the DMEM to the radio.

Table 3.7 shows the results of this case study.

Table 3.7: Microcontroller Case Studies

		Code Size	Clock Cycles	Energy(pJ)
Power Management	DPM	1	1	2.5
	MCU	24	8-16	11.6-23.2
RR/AFib - Setup	DPM	20	20	50.0
	MCU	40	40	58.2
	Ideal MCU	40	40	58.2
RR/AFib - Per Sample	DPM	4	1000	638.9
	MCU	1000	1000	1455.9
	Ideal MCU	10	1000	378.74
RR/AFib - Radio	DPM	22	22	55.0
	MCU	63	63	91.7
	Ideal MCU	63	63	91.7

## Conclusions

From our exploration, the MCU gives a clear advantage over the DPM in the programmability and capability metrics. The DPM is more energy efficient than the MCU due to its ability to

execute complex instructions in a single clock cycle. The DPM also consumes less current when idling compared to the implemented MCU.

We see through our case studies that the DPM's ISA is advantageous compared to the MCU's ISA in reducing number of clock cycles and code size required. This is due to the DPM's ability to execute complex operation in a single clock cycle. The reduction in the number of clock cycles results in energy savings over the implemented MCU in both cases. Additionally, this reduction can result in a smaller memory size; this memory size reduction is not factored into the calculated energy within the use cases.

In accelerator-based systems, the ability to NOP efficiently is very important consideration for selecting a microcontroller, assuming that the microcontroller has a low utilization. This is particularly evident in the "per sample" case in which the microcontroller idles for the majority of the 1000 clock cycles between samples. We see by implementing an energy efficient NOP in the MCU, it results in a 74% energy savings and a 100x reduction in the number of instructions in the duration of an RR/AFib sample case. The DPM shows up to an 89% energy savings over the MCU due to this ability to NOP. The table shows an additional savings of using the more ideal MCU over the DPM in the "Per Sample" case due to our assumption of the more ideal MCU's NOP energy. This energy is less than the DPM's NOP energy. For the use case, the DPM executes NOPs for 996 cycles and the more ideal MCU executes NOPs for 990 cycles and, thus, leads to this energy savings.

We can generalize these results. Custom microcontrollers run complex instruction set computing (CISC) operations tend to pull more current for a shorter period of time and consume less energy per operation. Generic controllers run reduced instruction set computing (RISC) operations that tend to consume less current but run for several cycles and result in a higher energy per operation. Custom controllers use fewer instructions than a generic controller which can result in using a smaller instruction memory and more NOPs. Therefore, selection of the type of microcontroller for accelerator-based systems will be based on the code size and the utilization of the core (number of NOPS).

Therefore, the custom microcontroller will likely be beneficial in terms of system energy over MCUs in cases where there is low utilization of the microcontroller and the core will be utilizing a lot of NOPs (assuming that NOPs in custom controllers are more energy efficient than generic microcontrollers) or extremely high utilization of the core in which the memory size reduction will overwhelm the results.

### 3.3.2 Bus Architecture

In accelerator-based BSN nodes, the bus is utilized to write and read configuration bits, inputs, and outputs to and from accelerators, the microcontroller, and memories. There can be a lot of addresses on the bus. The Texas Instrument's MSP430F1611 commercial microcontroller [59] utilizes a 16b data bus and up to 9b address bus. This allows for up to 256 peripheral address locations on the bus; the processor utilizes an address bus bit for writing only the high bits or low bits.

Decoders are utilized to ensure that data is sent and received by the correct peripheral address. Global decoding and local decoding, shown in Figure 3.19, are two of the main decoding topologies. Global decoding relies on a single block, such as the microcontroller, to decode the address and then connect the appropriate block or blocks to the bus. In the local decoding scheme, the address is put on the bus by a microcontroller or bus controller and each block is responsible for comparing that address to its own address. If the address matches its own, the decoder connects the register to the bus for reading or writing. Otherwise, the block remains isolated from the bus.

This section compares global decoding and local decoding schemes on an accelerator-based harvesting BSN node to inform future designs. This global decode scheme is used in batteryless BSN node described in Section 3.2. The DPM serves as the global decoder. A bus operation consumes the energy of a single DPM instruction, which corresponds to a power consumption that is approximately 9% of this node and 26% of the digital power when the

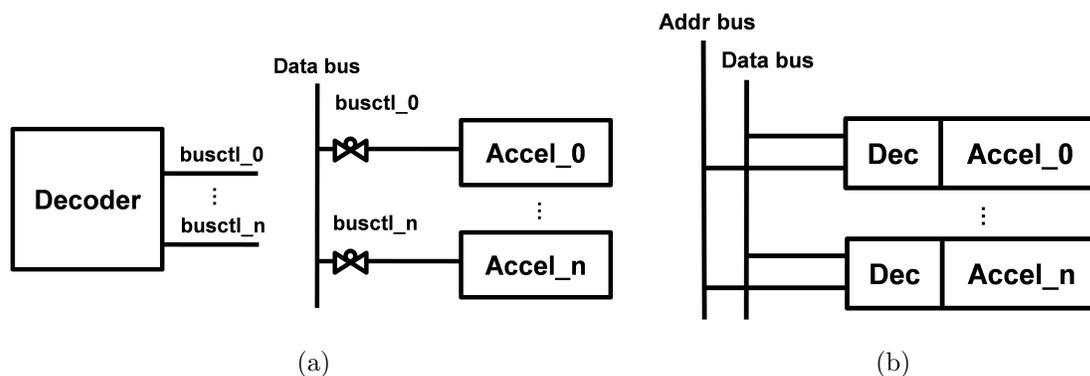


Figure 3.19: a. Structure of global decoding scheme. b. Structure of local decoding scheme.

bus is being used. We use energy and routing complexity as our metrics to compare these two schemes.

## Setup

We wrote behavioral Verilog code for global and local decoders of varying bit width. Global decoders of 2b, 4b, 6b, 8b, 10b, and 12b address bit width and local decoders of 1b, 2b, 4b, 6b, 8b, 10b, and 12b address bit width were written. For this exploration, we assume that each decoder uses an individual register and that no cluster optimizations are used to reduce logic. Additionally, each decoder has a 16b input and output to connect to the bus. Each decoder was synthesized through the Cadence RC compiler and simulations were run in Spectre in a commercial 130nm low power process to measure leakage and active energies.

For this exploration, we swept the number of addresses on the bus from 4 to 320 in a single-clock cycle write architecture. For each number of addresses, the smallest bit-width decoders were selected. We calculated the system energy for the active case (two decoders active and the remainder off) and the standby case (all decoders off). We then use this information to see the average energy as bus utilization changes. This utilization refers to the activity factor of the bus overtime. Lastly, we calculate the number of global wires that need to be routed to see the feasibility of physical implementation for each scheme.

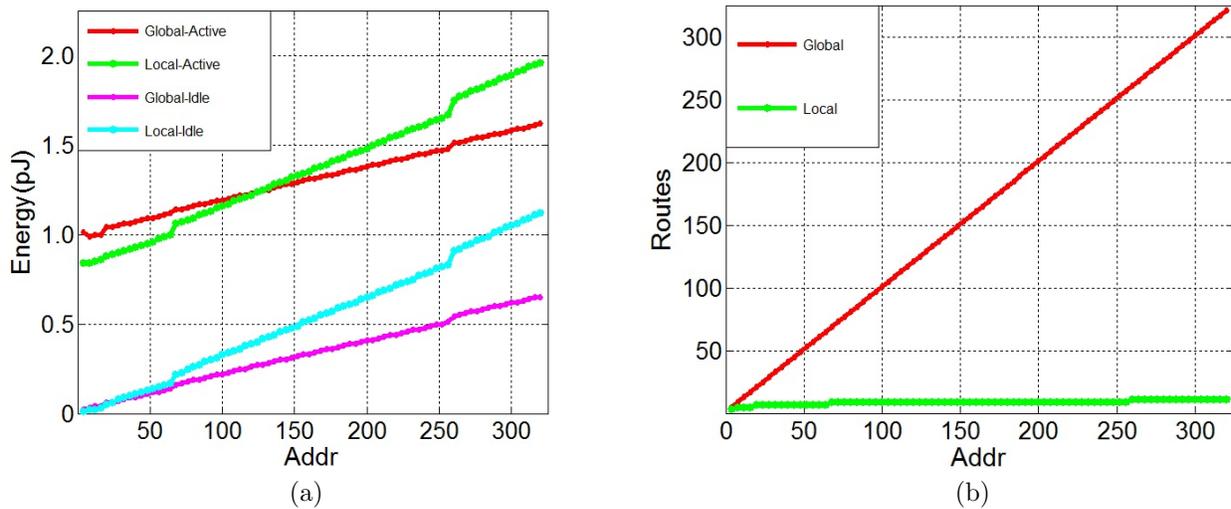


Figure 3.20: a. Simulated active and idle energy at 0.5V of global and local decoding schemes. b. Routing complexity of the global and local decoding schemes.

## Results

Figure 3.20a shows the active and leakage energy consumed at 0.5V. Changes in the decoder bit width account for the abrupt jumps within the graphs. We see that there is a breakeven point in active energy at 120 addresses. The shape of this graph is similar at 0.4V and 0.6V, but the breakeven point changes to 132 and 164 addresses, respectively. The idle energy for local decoding increases as the number of higher power components increases.

Figure 3.20b shows the number of global routes required for each scheme. As the number of addresses increases, the number of routes required for global decoding increases linearly. The local decode scheme’s routing complexity only increases in base-2 logarithmically.

## Conclusions

The routing complexity of the global decoding scheme makes the scheme unusable for a large number of addresses. A large number of addresses requires a lot of wires that originate from the a single point (the global decoder) and get routed through the chip. This causes large amounts of routing congestion near the global decoder and a large number of global

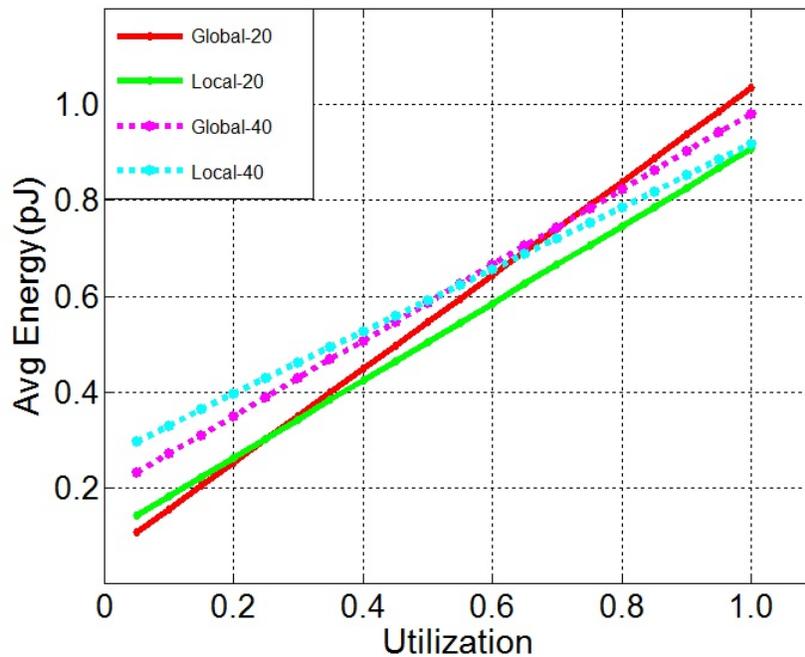


Figure 3.21: Average bus energy of 20 and 40 addresses as a function of utilization.

routes that increase complexity of top level routing within the design. This increased routing complexity leads to more area required, routing density issues, and the need for more routing metal layers.

However, with a manageable amount of routing complexity, selecting the correct scheme depends on the bus utilizations. Figure 3.21 shows the average bus energy as the utilization changes for 20 addresses and 40 address. The break even point between selecting the local and global decoding changes as a function of the number of addresses. As the number of addresses increases and the bus utilization increases, global decoding becomes more energy efficient due to lower active energy of the global decoding scheme and the increase in leakage energy from the local decoding scheme. The determination of the number of addresses that is considered to be too large to utilize the global scheme must be determined by the designer.

Table 3.8: Performance Comparisons with State-of-the-Art BSN Nodes [2]

	<b>This Work</b>	<b>Kim (VLSI '11)</b>	<b>Rai (ISSCC '09)</b>	<b>Verma (JSSC '10)</b>	<b>Yan (JSSC '11)</b>	<b>Chen (ISSCC '10)</b>
<b>Sensors</b>	ECG, EMG, EEG	ECG	Neural, ECG, EMG, EEG	EEG	ECG, TIV	Temp, Pressure
<b>Supply Voltage</b>	30mV, -10dBm	1.2V	1V	1V	1.2V	0.4V/0.5V
<b>E Harvesting</b>	Thermal, RF	×	×	×	×	Solar
<b>Supply Reg.</b>	✓	×	×	×	×	✓
<b>AFE</b>	4-channel	3-channel	1-channel	18-channel	4-channel	N/A
<b>Power Mgmt.</b>	DPM, Clock gating, Power gating	Clock gating	×	×	×	Power gating
<b>Gen. Purp. MCU</b>	1.5 pJ/Instr @ 200kHz (8b RISC ISA)	×	×	×	×	28.9pJ/Instr @ 73kHz (32b CORTEX-M3)
<b>Accelerators</b>	Programmable FIR, AFIB, MCU, Envelope Detector, DMA, Packetizer	ASIC DSP (4x SIMD), FIR, Encryption, DMA	×	ASIC DSP	FIR, Packetizer, Compression	×
<b>Memory</b>	5.5kB (0.3V-0.7V)	42kB (1.2V)	×	×	20kB (1.2V)	5kB (0.4V)
<b>DVS</b>	✓	×	×	×	×	×
<b>Digital Power</b>	2.1 $\mu$ W	~12 $\mu$ W	N/A	2.1 $\mu$ W	500 $\mu$ W	2.1 $\mu$ W (MCU)
<b>TX (datarate)</b>	200kb/s	×	100kb/S	×	1Mbps (on-body link)	×
<b>TX P<sub>DC</sub> (100% on)</b>	160 $\mu$ W	×	400 $\mu$ W	×	2.8mW	×
<b>TX P<sub>OUT</sub></b>	-18.5dBm	×	-16dBm	×	-6dBm	×
<b>TX band</b>	402 / 433 MHz	×	402 / 433 MHz	×	20-40 MHz	×
<b>Total Chip Power</b>	19 $\mu$ W	31.1 $\mu$ W	500 $\mu$ W	77.1 $\mu$ W	2.4mW	7.7 $\mu$ W
<b>Note on Total Power (includes):</b>	1-channel AFE, 8b ADC, DSP (R-R extraction), and TX duty-cycled at 0.013%	AFE, 12b ADC, DSP (heart beat detection)	1-channel AFE, 8b ADC, and streaming TX with 100% duty-cycle	18-channel AFE, 12b ADC, and DSP (EEG feature extraction)	4-channel AFE, 10b ADC, DSP (data compression, FIR), SRAM, TX at 5% duty cycle	Data acquisition, DSP (DFT), storage in SRAM, sample taken every hour
<b>Technology</b>	130nm	180nm	130nm	180nm	180nm	180nm

### 3.4 Summary and Conclusions

This chapter presents the first wireless biosignal acquisition chip BSN node powered solely from energy harvested from body heat and/or RF power with integrated supply regulation, AFE, power management, DSP, and TX. This tight system integration of low power blocks results in low system power that allows this node to be powered by an energy harvesting, potentially extending its lifetime indefinitely. Table 3.8 compares the performance of recent BSN SoCs. This energy harvesting enables a potential infinite lifetime for this BSN node. To the best of the authors' knowledge, this system has lower power, lower minimum input supply voltage, and more complete system integration than other reported wireless BSN SoCs to date.

We use this SoC to inform future energy harvesting-powered BSN nodes. Our exploration of custom controllers illustrates how important idling energy efficiency is and the trade-offs of using using a custom microcontroller. This chapter's decoder exploration shows that the global decoders can provide a lower power solution than the local decode scheme when the number of global routes is not prohibitive.

# Chapter 4

## Power Management for Energy Harvesting-Powered BSN Nodes

<sup>1</sup> Body sensor networks (BSNs) show great promise to provide significant benefits to the healthcare field. This technology helps physicians diagnose, prevent, and respond to various illnesses such as diabetes, asthma, and heart attacks [21] through long term monitoring of patients. Though BSNs show great potential, they have many design challenges that may impede their widespread adoption. One of the most critical issues is device lifetime. Short BSN lifetimes undermine the effectiveness of long term monitoring applications, such as cardiac monitoring and fall detection. This chapter focuses on improving lifetime in BSN nodes. Large batteries are too big for the desired BSN node form factor, and small batteries have short lifetimes that require patients or clinicians to replace or recharge batteries, reducing patient compliance in wearing the node and the overall effectiveness of the BSN.

Energy harvesting mechanisms, such as solar power, thermoelectric generation, and vibrations, provide an alternative power source and are capable of providing an indefinite source of node power. However, there are several challenges with using harvested energy

---

<sup>1</sup>This chapter is based on the published papers titled: "A Batteryless 19  $\mu$ W MICS/ISM-Band Energy Harvesting Body Sensor Node SoC for ExG Applications" [YS2], "A Custom Processor for Node and Power Management of a Battery-less Body Sensor Node in 130nm CMOS" [YS3], and "A Battery-less 19 $\mu$ W MICS/ISM-Band Energy Harvesting Body Area Sensor Node SoC" [YS6].

for long term operation. First energy harvesting sources present a non-ideal supply voltage. Depending on the source, they suffer from widely varying output voltages, large ripple, and extremely low ( $<100\text{mV}$ ) voltage levels. Second, high power operations, such as transmission of data over a wireless radio, can consume 100s of  $\mu\text{Ws}$ , likely exceeding the energy harvested. Finally, energy harvesters' power output is highly environment-dependent [14]. For example, thermoelectric generators' power output changes as the ambient temperature changes.

Though energy harvesting-powered BSN nodes are relatively sparse, many of them [42] [40] have ignored challenges presented by the varying nature of harvested energy, instead focusing on utilizing low power blocks and heavy duty cycling to achieve a low average power consumption. This approach does not ensure a longer lifetime in energy harvesting-powered devices. For example, this low power approach fails to account for a patient wearing a solar-powered BSN node outside in ideal, sunny conditions who then steps into a building. This change of environment causes a substantial reduction in harvested power to which the node does not adapt to. Operating in a static, low power mode that ignores the time-varying nature of power availability to the node will lead to node death, resulting in loss of volatile memory which holds functionality (instruction memory) and important biosignal data (data memory). Therefore, node power consumption must adapt to the varying nature of harvested energy through power management.

Power management systems have been proposed previously in a multitude of battery-powered sensor nodes [60] [61] [62]. However, the metrics for evaluating energy harvesting systems are different from those used for battery powered systems. Harvested energy is distinct from battery energy in two ways. First, it is an inexhaustible supply which, if appropriately used, can allow the system to operate indefinitely, unlike a battery which is a limited resource. Second, there is an uncertainty associated with its availability of energy, compared to the energy stored in the battery which can be known deterministically [25].

Therefore, we require an energy harvesting-specific, closed-loop power management system that is capable of tracking or sampling the health of the node (power consumption of the

node versus energy harvesting power) and adjusting the power consumption immediately to extend lifetime and reduce the probability of lifetime.

This chapter explores several energy harvesting-specific power management decisions such as single-cycle power modification and tracking the health of the node. We subsequently present the first and second revisions of an on-chip power management system for an ultra-low power batteryless energy harvesting body sensor node BSN SoC. We extend the state of the art and knowledge in this area in several ways in this chapter. This chapter presents the first implemented on-chip power management system for a fully power harvesting node. The explorations into single-cycle power modification and methods for checking the amount of energy on the storage capacitor help inform future power management designs to improve lifetime in energy harvesting BSN nodes.

## 4.1 Related Work

The need for power management systems specific to energy harvesting devices has been proposed and implemented in commercial off the shelf (COTS) based wireless sensor nodes. Proposed energy harvesting-specific power management schemes track the amount of power harvested and the amount of energy consumed. In addition, these systems may include some form of prediction to anticipate any changes to the amount of energy harvested. Based on these inputs, the manager will adjust the power consumed by the node or adjust their node to harvest more power. Power managers utilize duty cycling, dynamic voltage scaling, and maximum power point tracking (adjusting the load to maximize the current from the energy harvester) to reduce power [63]. We look at three implemented systems.

Liu et al [64] proposed a power management optimization approach to develop power adaptive solar powered computing systems. Based on the power available and prediction based on previous day's data, the power manager decides how many processing units can

operate. Other processing units are clock gated, allowing this system not to exceed the power harvested.

Raghunathan et al [65] presented a solar energy harvesting module, Heliomote, which utilizes an energy harvesting power management scheme. This system tracks the amount of power harvested by solar cell through a low power battery monitor chip and adapts its load to improve efficiency of the harvesting through maximum power point tracking.

Hsu et al [25] implemented an adaptive duty cycling algorithm that allows energy harvesting sensor nodes to autonomously adjust their duty cycle according to the energy availability in the environment. The goal of their power management scheme is to achieve energy neutral operation (energy consumption should not be more than the energy provided by the environment) through prediction based on the previous day's energy harvesting profile. Further work in [66] [63] has built on this idea to duty cycle by proposing new algorithms to maximize the duty cycling rate.

These power management schemes have the potential to extend lifetime in their respective areas. However, all of these schemes assume a much higher amount of harvested power ( $>1\text{mW}$ ) and assume monitoring schemes that consume more power than the power budget of these energy-constrained BSN nodes.

## 4.2 Power Management Explorations

This section looks at three important energy harvesting-specific power management design decisions: sampling node health, the number of operating modes required, and single-cycle power modification. These explorations inform our power management system designs presented in Sections 4.3 and 4.4.

### 4.2.1 Sampling Node Health

Sampling or tracking BSN node health is required to compare the ratio of node power consumption to power harvested for making decisions. We define the health of the node as the difference between the power harvested and power consumed. The storage capacitor located at the output of the boost converter and the input of the DC-DC converter serves as an indicator of the health of the node. Excess power from the energy harvester is stored in the capacitor. If the power from the energy harvester is less than the amount being consumed, energy will be consumed from the capacitor. We can see how much energy is stored on this capacitor through sampling its voltage ( $V_{Boost}$ ). Stored energy and voltage have simple relationship of:

$$E_{stored} = 0.5 * C_{Boost} * V_{Boost}^2 \quad (4.1)$$

where  $C_{Boost}$  is the capacitance of the storage capacitor. This section investigates two ways of digitizing the voltage on the capacitor on node: using an analog-to-digital converter (ADC) and using a ring oscillator (RO).

#### Structures

ADCs are commonly used to digitize analog voltages biosignals. The BSN node presented in Section 3.2 utilizes an 8b successive-approximation (SAR) ADC to digitize electrocardiography (ECG), electromyography (EMG), and electroencephalography (EEG) signals. Since ADCs consume a significant amount of power ( $>1\mu W$ ), we look to time-multiplex this. However, ADCs have their own input voltage range (in this case, 0 to 600mV), requiring  $V_{Boost}$  to be modified. Our BSN node utilizes an operational amplifier to divide the  $V_{Boost}$  in half and a variable gain amplifier to maximize the signal at the input to the ADC, maximizing our 8b ADC output.

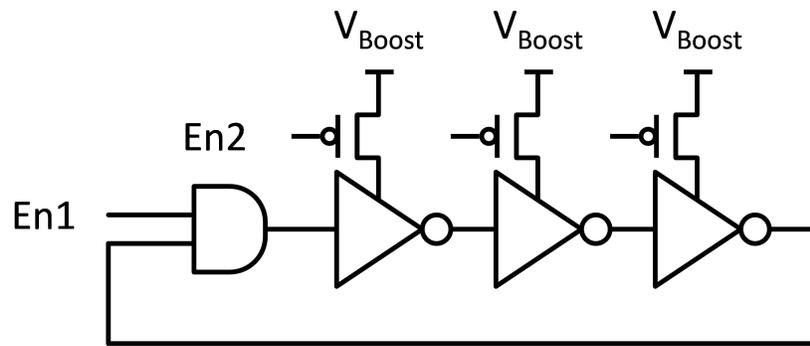


Figure 4.1: RO configuration for digitizing  $V_{\text{Boost}}$ .

A RO is another approach to digitizing values [67]. RO's frequencies vary as a function of voltage. We propose connecting the voltage of the ring oscillator directly to  $V_{\text{Boost}}$ , as shown in Figure 4.1. A power manager is capable of counting the number of RO pulses for a period of time and correlating the number of pulses to  $V_{\text{Boost}}$ .

## Setup

This section explores the energy/power trade offs between using the ADC and RO for various sampling rates. For this exploration, we use the ADC [48] and the amplifiers from the BSN node in Section 3.2 and the RO from [55]. The ADC and amplifiers were designed for ExG operation on BSN nodes, thus making it an acceptable choice for this exploration. The RO was chosen because of its lower power topology and because it is temperature compensated.

For the ADC, we obtain the power/energy of the ADC at different sampling frequencies and the amplifiers used to convert the voltage through Spectre simulation. We then obtain the frequency, and energy/power for the RO and a digital counter through Spectre simulation. We use this information frequency information to inform our window size (the number of system clock cycles in which the DPM counts RO pulses.) A long enough window is required to resolve  $V_{\text{Boost}}$ . With the selected window side, we calculated the RO energy/power for each sampling frequency. We compared energy/power of the ADC and RO power and energy as a function of sampling rate. The required sampling rate changes based on the energy harvesting

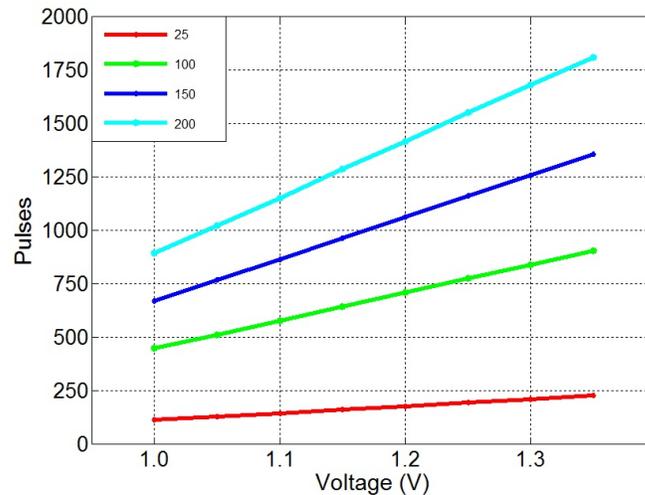


Figure 4.2: Simulated number of RO pulses per window over the range of voltages.

mechanism, the storage capacitor size, and the power consumption of the node. Determining the optimal sampling rate is not the focus of this dissertation.

## Results and Conclusions

Figure 4.2 shows the number of RO pulses as a function of the  $V_{\text{Boost}}$  and window size. From this graph we see that a window size of 200 cycles (at 200 kHz) will be sufficient. This window size achieves a conversion rate of 3 RO pulses per mV. This window size corresponds to a maximum sampling frequency of 1 kHz; slower sampling frequencies will result in the RO idling for the remainder of that period of time.

Figure 4.3a and 4.3b compare the power and energy, respectively. The RO's power and energy is a function of  $V_{\text{Boost}}$ . We plot the power/energy at 1.0V and 1.35V, which we assume to be the maximum and minimum  $V_{\text{Boost}}$ . The RO gives us a substantial energy/power savings of up to 6x over the ADC.

However, many BSN nodes require the ADC to digitize biosignal data. We compare the power/energy of time-multiplexing the ADC against running the ADC and RO in parallel. We look at two sample cases: a BSN application that requires an ADC sampling rate of 200 kHz and one that requires ADC sampling rate of 350 kHz. To time-multiplex, we assume

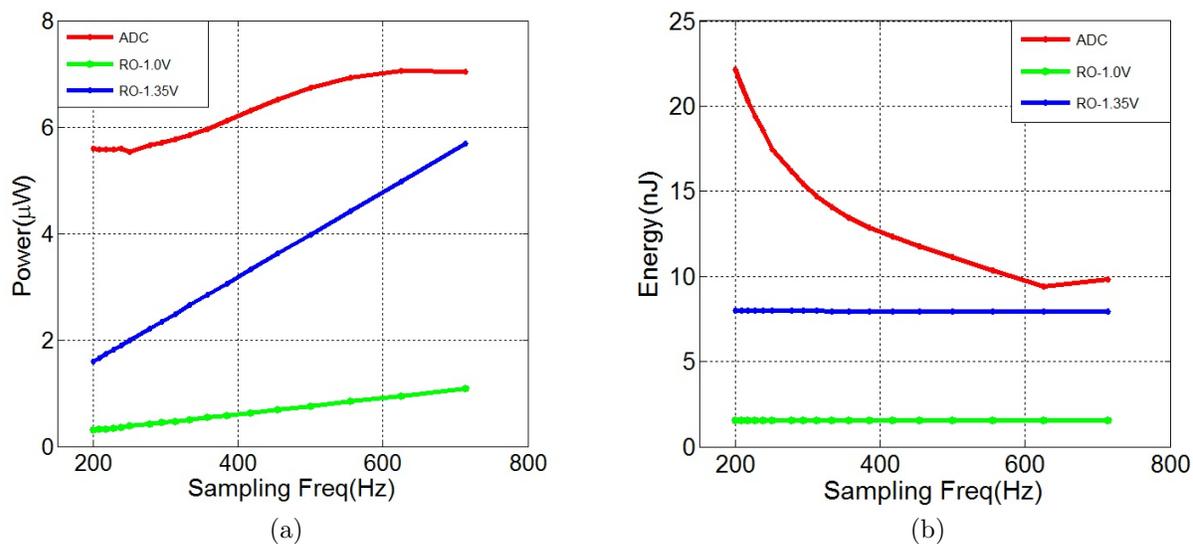


Figure 4.3: a. Simulated average power of RO and ADC. b. Simulated energy of RO and ADC.

that we need to at least double the frequency; this ensures that the the node will not miss a biosignal data. Table 4.1 shows the energy/powers result of this exploration. We see that the addition of a RO to a system that requires an ADC will give minimal to no energy/power benefits.

We can conclude from the experiment that the RO will likely be more energy/power efficient means to sample the health of node than the ADC, but when an ADC is required for the BSN application, time-multiplexing will be more advantageous.

Table 4.1: Energy/Power of RO vs ADC Time-Multiplexing

	Energy(nJ)	Power( $\mu$ W)
ADC/RO @ 200Hz	26.54-30.11	5.90-7.19
ADC @ 400Hz	24.64	6.30
ADC/RO @ 350Hz	14.96-21.40	6.39-8.69
ADC @ 700Hz	19.70	7.03

### 4.2.2 Number of Operating Modes

As the harvested output is decreased, the power manager is required to decrease node power consumption. However, the approach to reducing power depends heavily on the application. Some applications that utilize multiple accelerators and analog blocks will be able to turn off many blocks gradually. For example, the power manager could turn high power blocks off one-by-one to try to preserve functionality. Other applications do not require that. We define each set of operations that can be on to be an operating mode. Operating modes effectively constrain the maximum amount of node power consumption through power gating blocks, clock gating blocks, or other power saving techniques.

This exploration investigates the power and energy overheads as the number of operating modes are swept.

#### Setup

We wrote behavioral Verilog code for a power manager that swept the number of operating modes from three to seven. Each power manager used a simple, 8b thresholding scheme to determine its operating mode. Each operating mode holds 16 control bits for controlling blocks on the chip. When an operating mode is selected, its control bits are selected as the power manager's output bits. Each power manager was synthesized through the Cadence RC compiler. These power managers were verified for functionality in Verilog and Spectre in a commercial 130nm low power process. We used Spectre to measure active and idle energy.

#### Results and Conclusions

Figure 4.4a shows results of our simulations. As the number of operating modes is swept, we see a relatively flat profile for idling energy and the active energy increasing roughly linearly as expected. We pay an additional 15% active energy penalty (compared to a 3 operating mode system) for each operating mode added. However, we can vary the utilization of this block, which is effectively the activity factor, to amortize the additional energy over long

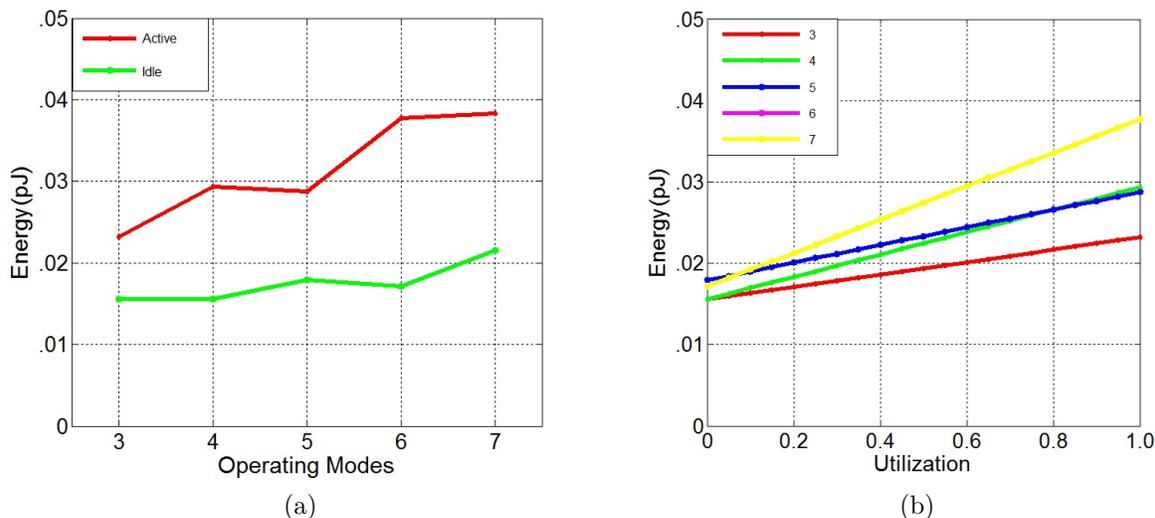


Figure 4.4: a. Simulated active and idle energy at 0.5V of the number of operating modes of a power manager. b. Simulated average energy of power manager based on activity factor for different numbers of operating modes.

periods of time (Figure 4.4b), leading to a flat average energy profile. Thus, we see that the number of operating modes is not a strong knob for influencing the power manager's energy; we can achieve added flexibility within the power manager for a minimal energy overhead.

### 4.2.3 Single-Cycle Modification

Energy harvesters' power output changes very rapidly, varying as environmental conditions change. Power managers must adapt the node power consumption. This section investigates the speed required to adapt this power consumption. Adapting too slowly to changes in harvested energy can lead  $V_{\text{Boost}}$  to drop below the the critical voltage in which essential components cannot operate ( $V_{\text{Kill}}$ ), resulting in node death. Additionally, we look at the feasibility of implementing this power management scheme in a generic microcontroller, seeing if the microcontroller can achieve the required speed to adapt the power consumption.

## Setup

We look at three comparison points: an ASIC power manager that is capable of single-cycle power modification, a generic microcontroller with override structures, and a generic microcontroller without override structures. We define an override structure to be a structure that is capable of masking the current state of block (on or off) with no latency. In this case, we assume that the microcontroller holds a single register that contain override values that are electrically connected to their respective blocks.

For this exploration, we assume an architecture that is a simplified architecture of the BSN node presented in Section 3.2 consisting of a transmitter (TX), an analog front-end (AFE), a digital portion (memories, microcontroller, accelerator, etc), a supply generation portion (boost converter, DC-DC converters), and a crystal oscillator (XTAL). We use the measured power values from Table 3.3 and assume that each block has its own control register that is capable of turning a block off or on. We use the generic microcontroller (MCU), described in Section 3.2.4, based on the PIC architecture for this experiment.

We constrain the power manager to three operating modes (green, yellow, and red modes). The power manager must compare the value of  $V_{\text{Boost}}$  to two thresholds to resolve its operating mode. In green mode, all blocks are allowed to be on. In yellow mode, the TX is turned off, and all other blocks are allowed to be on. In red mode, only critical blocks (XTAL, supply, microcontroller, and instruction memory) are allowed on.

In this exploration, we look at the most critical scenario for this power management scheme. All blocks (TX, AFE, digital, supply, and XTAL) are on and operating, the power manager is in green mode, and the energy harvesting power suddenly stops; the node is powered solely off its storage capacitor. The power manager is required to resolve its operating mode and the power profile to the bare minimum (instruction memory, microcontroller, supply, and XTAL).

## Results and Conclusions

Table 4.2 shows the result of this experiment with the assumption of 200kHz clock rate. The MCU requires up to 16 instructions to resolve the operating mode and an additional two instructions per write operation to the override register or each configuration register. The override structure requires the MCU to write only a single register. The MCU without override requires a write operation for each block's configuration register to ensure it is being turned off. During those clock cycles, high power blocks such as the radio and AFE are on, resulting in wasted energy.

The single-cycle ASIC approach shows a 17x improvement in energy. The override structure accounts for 6 clock cycle and .4% energy savings. These savings will increase if more high powered blocks are required to be turned off. The ability to use a MCU is dependent on the storage capacitor size and  $V_{Kill}$ . Equation 4.2, derived from Equation 4.1, shows amount energy available on the capacitor before it reaches  $V_{Kill}$ , assuming no energy harvested input.

$$E_{kill} = 0.5 * C_{Boost} * (V_{Boost}^2 - V_{Kill}^2) \quad (4.2)$$

This equation indicates that the largest capacitor size would be best. However, large capacitors suffer from greater leakage; additionally, large capacitors take more power to charge, making  $V_{Boost}$  recovery time much longer if it drops. For our node, we selected a 47nF storage capacitor. Assuming the capacitor is initially charged to 1.35V and a  $V_{Kill} = 1.2V$ , the energy consumed in both MCU cases would result in the  $V_{Boost}$  dropping below  $V_{Kill}$ , resulting in node death. This necessitates a departure from using a generic microcontroller for power management and provides the impetus for an ASIC power management blocks.

Table 4.2: Calculated Time and Energy Overheads of Different Power Management Schemes

	Number of Clock Cycles	Energy Consumed
Single-Cycle	1	1.94nJ
MCU- Override Structure	18	34.93nJ
MCU- No Override Structure	24	35.07nJ

### 4.3 Revision 1

The digital power manager (DPM) is the first implemented on-chip power management system for a BSN node solely powered by energy harvesting. It serves as the always-on power manager and custom microcontroller for the BSN node presented in Section 3.2. The node is batteryless, powered entirely from a wearable thermoelectric generator (TEG) with an off-chip storage cap and an on-chip boost converter. The chip supports electrocardiography (ECG), electromyography (EMG), and electroencephalography (EEG) applications and is capable of waveform acquisition, signal analysis and processing, and wireless communication over a MICS radio.

The node is powered by a energy harvesting/supply regulation section that boosts an input as low as 30mV up to a regulated 1.35V supply. However, the amount of energy harvested varies as the environment conditions change. The DPM adjusts the node's power consumption in response to real time measurements of available harvested energy to support batteryless operation.

The DPM's main purpose is to ensure the BSN node remains alive by making intelligent processing and transmitting decisions based on the amount of energy harvested and available on the storage capacitor. The basic premise of the DPM is that it checks the  $V_{\text{Boost}}$ , which corresponds to the energy available on the storage capacitor, and selects and implements an operating mode, which limits the maximum amount of node power consumption for the given amount of energy available through overriding the state of the chip. If conditions are poor, the DPM throttles down processing and/or turns off transmission to save power and preserve

the node; likewise, if conditions are favorable, the DPM can allow more blocks to be utilized.

### 4.3.1 Sampling the Health of the Node

The DPM utilizes the 8b ADC to digitize  $V_{\text{Boost}}$ . This ADC is time-multiplexed with between this power management scheme and converting biosignal data. The 8b digitized  $V_{\text{Boost}}$  is fed directly into the DPM immediately when a new energy value is available.

This DPM functionality is independent of the sampling frequency of the capacitor; the DPM is capable of operation at 200 kHz which is the system clock or as slow as the application/user dictates. This sampling frequency is a function of the capacitor size, energy harvesting mechanism, and, since this ADC is being time-multiplexed, the BSN application itself. The capacitor value is only digitized when the DPM microcontroller ADCCHAN instruction (Table 3.1) is issued, selecting the  $V_{\text{Boost}}$  input to the ADC. This provides the programmer with the flexibility to bypass any power management restriction by not issuing the instruction. This is useful if transmitting or processing the data is more important than keeping the node alive and thus, needing to bypass the power management scheme (i.e. the node detected a heart arrhythmia and needs to transmit the ECG to notify the doctor over the wireless radio, regardless of energy status).

### 4.3.2 Stoplight

The DPM has three hard-coded operating modes (red, yellow, green) that correspond to a traffic stoplight. Table 4.3 shows the three operating modes and each of mode's hardcoded policies. In green mode, all operations are available to be run. In yellow mode, the node reduces its power consumption through duty cycling of the TX. In red mode, the AFE and TX are overridden to the off position to preserve the instruction memory and reduce the likelihood of node death. Loss of instruction memory state forces the node to a default atrial fibrillation detection program. Note that DPM operating modes do not turn blocks on, but,

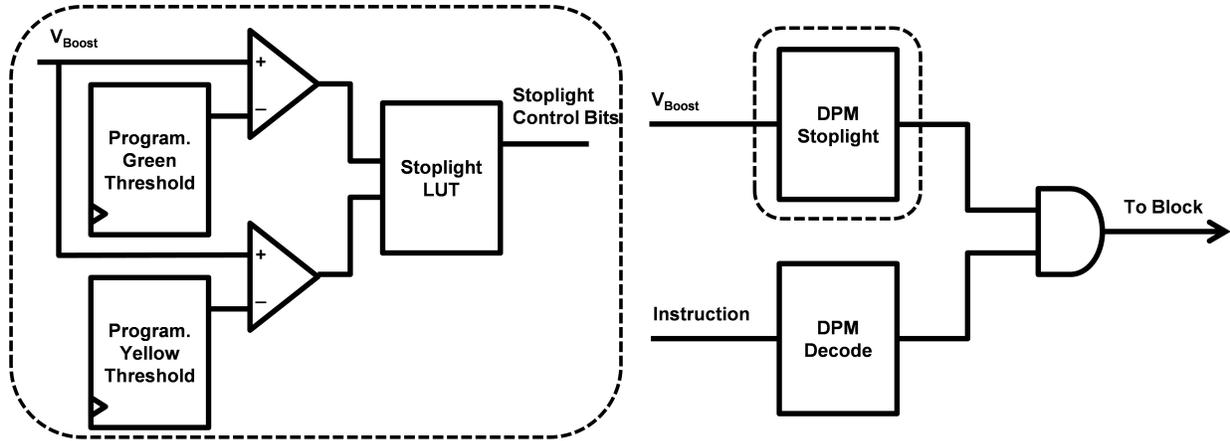


Figure 4.5: Override structure of the DPM stoplight. The stoplight compares the  $V_{\text{Boost}}$  value to the threshold, selects the operating mode, and outputs control bits to the chip [8].

rather, they override the state of the block; the DPM microcontroller EN instruction must be issued to turn on any block.

The DPM stoplight issues bits that can prevent mode-restricted blocks from running operations through overriding the power gate and clocking bits, shown in Figure 4.5. This allows the DPM to modify power consumption within one clock cycle. The energy savings of single-cycle power modification is shown in Section 4.2.3. The DPM’s stoplight has the ability to power gate or clock gate individual node blocks, such as the transmitter, accelerator blocks, and analog front end channels to adjust power consumption.

Table 4.3: DPM Operating Mode

	IMEM	AFE	DMEM	Accel	Transmit
Red	On	Off	Off	Off	Off
Yellow	On	On	On	On	Duty Cycle
Green	On	On	On	On	Off

### 4.3.3 Threshold Values

Threshold values are used to determine the operating mode. In this scheme, the DPM compares the digitized capacitor value to two 8b threshold values (green and yellow threshold)

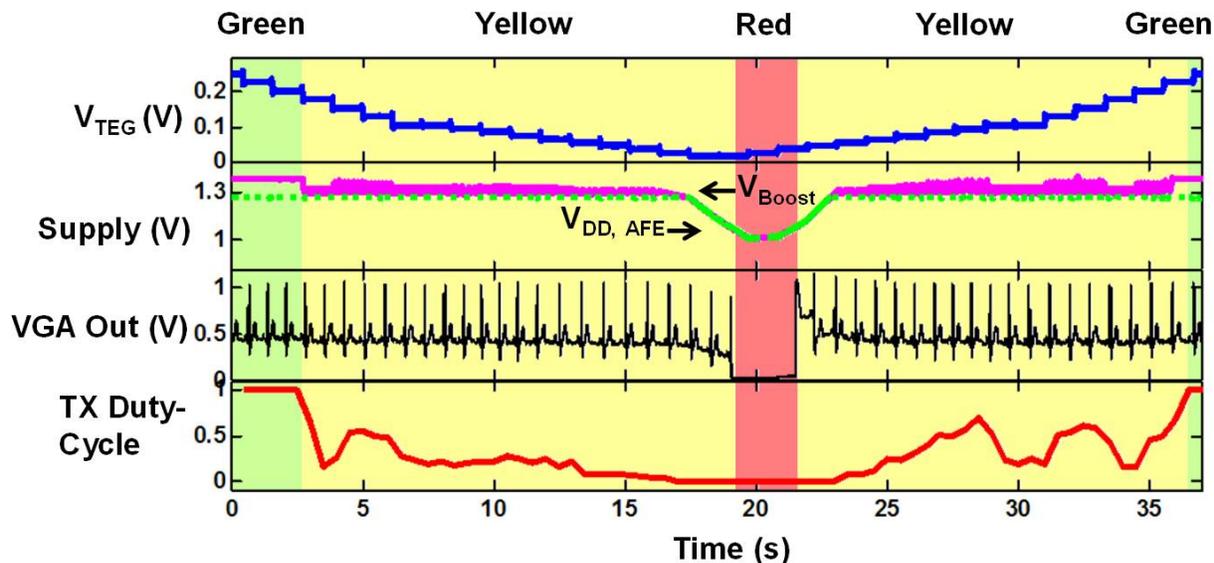


Figure 4.6: Measured DPM closed-loop power management response [2].

and selects the operating mode. The current operating mode updates immediately and updates the override bits that are used in Figure 4.5. The DPM is capable of jumping from the current mode to any mode (i.e. green to red mode) or staying in its current mode at a threshold change. These threshold values are calculated apriori by the user and programmed through scan values. The selected threshold values are based on the application's priorities. Programmers can adapt their priorities on how critical high power operations, such as transmitting and biosignal acquisition, are versus ensuring lifetime through the selection of the threshold. To ensure the node can transmit properly at any voltage, the user should use lower threshold values. Additionally, threshold values can be reprogrammed based on a change in storage capacitor value and to deal with variation in values on a chip by chip basis.

#### 4.3.4 Measured Results

Figure 4.6 shows a measured plot of the DPM stoplight overriding the previous state of the node as the TEG input voltage is varied. The DPM transitions from green mode down to red mode and back to green. As the capacitor voltage varies, the DPM duty cycles the TX

and turns off the AFE and the TX, limiting node power consumption when the simulated energy harvester’s power decreases.

For this revision, the DPM is a combination power manager and microcontroller. The energy-delay curve is shown in Figure 3.18a. The DPM measured energy per operation on the 130nm chip at 0.5V is 2.74pJ. This includes the overheads of fetching and decoding an instruction but does not include the energy of the ADC or AFE. The measured idling energy is 0.68pJ/cycle at 0.5V.

## 4.4 Revision 2

We made significant improvements in the state-of-art in the first revision of the DPM. However, the DPM has several weaknesses. The DPM’s most glaring weakness is that it is too specific to its BSN node, not possessing the scalability or flexibility to be used in future BSN node designs. In Revision 1, the power management is integrated into the custom microcontroller, operating mode control bits are hard-tied to predetermined values, and it is unable to scale to a different number of outputs (blocks to control) based on the node. Last, this node may not allow  $V_{\text{Boost}}$  to fully recover through a condition called “lock up.” Lock up occurs when the DPM oscillates between two operating modes.

We address these weaknesses in this second revision of the DPM and implement a flexible/scalable power manager, capable of being reused in future BSN nodes.

### 4.4.1 Chip Architecture

The second revision of the DPM was implemented on a BSN node in 130nm technology (Figure 4.7). This SoC serves as a platform for future BSNs, allowing for plug-and-play of future low-power accelerators for faster development of BSNs. These future accelerators will utilize different algorithms for existing biosignals or will target the acquisition and processing of new biosignals.

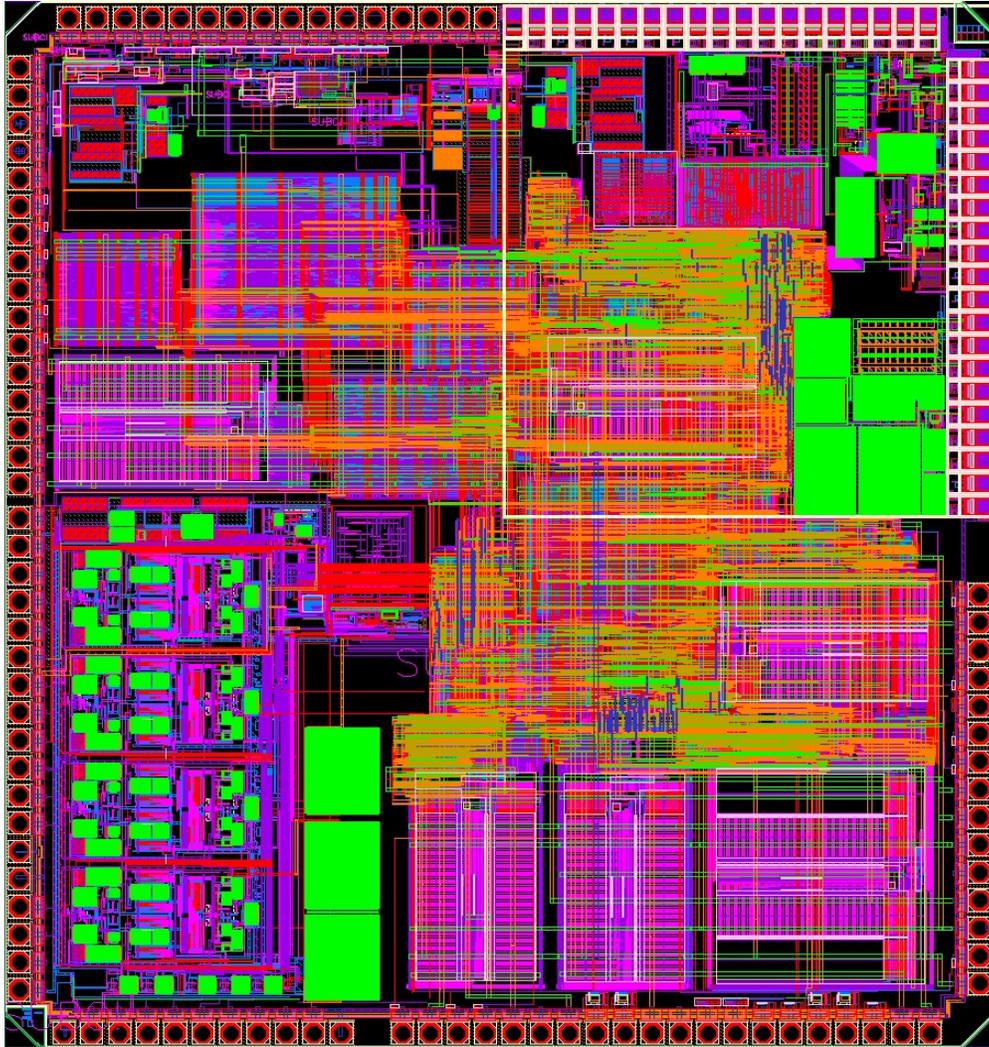


Figure 4.7: Top level layout of the second BSN node.

Figure 4.8 shows the block diagram of this SoC. This chip consists of a four-channel AFE, an integrated power harvesting/supply regulation section boosts, a subthreshold digital section for processing, a wake-up radio, a transmitter and receiver for wireless transmission, and subthreshold memories for data storage. The node supports both 8-bit and 16-bit architectures, allowing future programmers more flexibility to choose their architecture.

The node supports many application modes. This chip is SoC of ECG, EMG, and EEG acquisition, processing, and transmission and is capable of interfacing to standard commercial sensors through I2C and SPI to acquire other biosignals.

The DPM is a critical component of this energy harvesting BSN platform. The DPM is

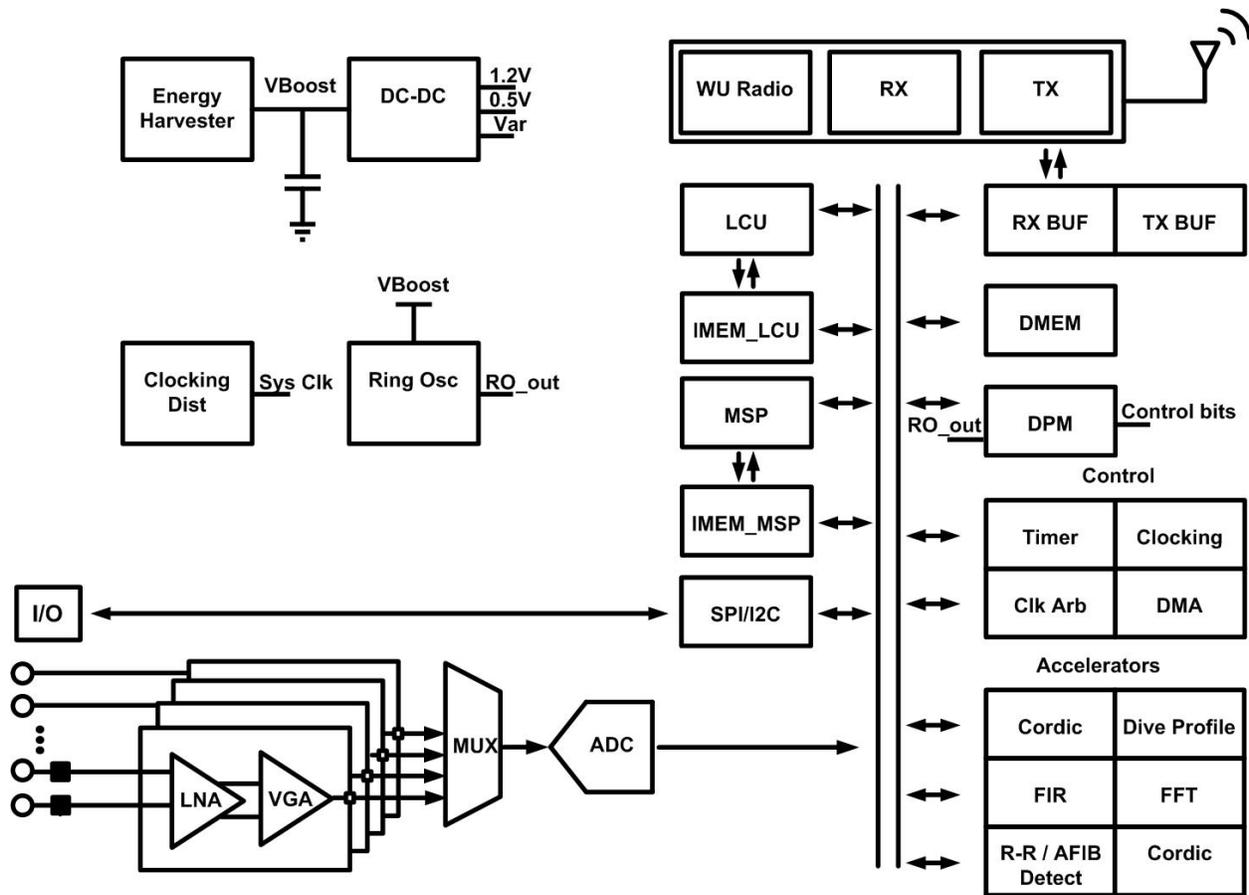


Figure 4.8: Block diagram of the second BSN node.

designed to adjust node power consumption in a programmable method, allowing it to be used in future revisions of BSNs. In this revision, the DPM serves as solely a power manager; the microcontroller functionality was removed from the DPM to reduce the DPM's power consumption to the bare minimum and allow it to be used on future BSN nodes.

This revision keeps the original power management scheme of sampling the energy on the capacitor, comparing to the threshold values, and selecting the correct operating mode to adjust the power consumption while addressing the aforementioned weaknesses of its predecessor. The next subsections detail this revision's programming flexibility in the operating mode, reduction on the reliance on the ADC, and reduction of lock up of the BSN node due to constant switching of operating mode.

### 4.4.2 Programmability/Flexibility

In the previous revision, the operating mode's control bits are hard-tied to the values specified in Table 4.3, preventing the DPM from reflecting different priorities. For example, the user could want to allow transmission in yellow mode since there is critical data to be transmitted or shut off the AFE in yellow mode since node preservation is the priority. For the DPM to be useful in future BSN nodes, it must be programmable to reflect different priorities and scalable/flexible to control different blocks to reduce power and improve lifetime in energy harvesting devices.

In this revision, the user has the ability to program both the thresholds and the operating mode control bits. These 8b thresholds and 16b operating mode control bits are programmable at runtime and during operation. This operating mode control bits are assigned at design time to specific blocks; multiple bits can be assigned to a single block. This enables the DPM to utilize more power modification methods than clock gating and power gating. Multiple control bits can be sent to an arbitrary block to allow for low power modes, frequency scaling, and algorithm changes.

For this node, we assigned the operating mode control bits to turn off and bypass the phase-locked loop (PLL) to reduce the frequency and reduce power, turn off the AFE, put the DC-DC converter into a low power mode, power gate the memories, and power gate and clock gate the accelerators. Additionally, the digital accelerators are divided into two tiers of accelerators to allow for more flexibility to delineate critical blocks from non-critical block.

### 4.4.3 Sampling the Energy

This revision of the DPM supports two methods of sampling  $V_{\text{Boost}}$ : utilizing the ADC and utilizing a RO connected to the  $V_{\text{Boost}}$ . The ADC is used in the original revision as the sole method to digitize the voltage on the capacitor. This approach places additional burden on the ADC which is responsible for digitizing biosignal data. To ensure that no biosignal samples are missed, the clock frequency for the ADC must be increased which results in a

higher power consumption for a power constrained system (Figure 4.3a). Additionally, the ADC may not be able to handle the increase in frequency.

The RO approach allows the node not to rely on the ADC or the AFE and enables the use of the DPM in systems that do not have ADC. For this chip, we selected the RO used in [55]; the ring oscillator utilizes a lower power topology and is temperature compensated. The DPM is responsible for turning on the RO, counting the number of pulses, and turning off the RO. The RO is power gated to save power when  $V_{\text{Boost}}$  is not being measured. The RO window (number of clock cycles that RO pulses gets turned off) and RO frequency (how often  $V_{\text{Boost}}$  is sampled) are programmable.  $V_{\text{Boost}}$  can be digitized in parallel to the ADC digitizing values, leading to a reduction in ADC clock frequency.

A detailed comparison between these two methods is available in Section 4.2.3.

#### 4.4.4 Reduction of Lock up

The previous revision of the DPM resolves its operating modes based on comparison to two 8b thresholds. This structure enables the possibility of lock up. Lock up is when the DPM oscillates between modes. In lock up,  $V_{\text{Boost}}$  to be close to threshold and to jump down to a lower threshold, recover enough to jump to a higher operating mode, and then jump back down to the lower operating mode (i.e. green mode to yellow mode to green mode to yellow mode).

We can demonstrated lock up using a simple model (Figure 4.9a) in which the energy harvester and the load are ideal current sources. Figure 4.9b illustrates the sequence in which the transmission operation puts the node into yellow mode, the node recovers and transitions into green mode, and then the next transmission puts it back into yellow mode.

To address this possibility, this revision has two built in features: asymmetric thresholding and hysteresis. Asymmetric thresholding provides two separate thresholds: one for switching to a higher operating mode (up-switching) and one for switching down to a lower operating mode (down-switching). The difference between the two thresholds is programmable. This

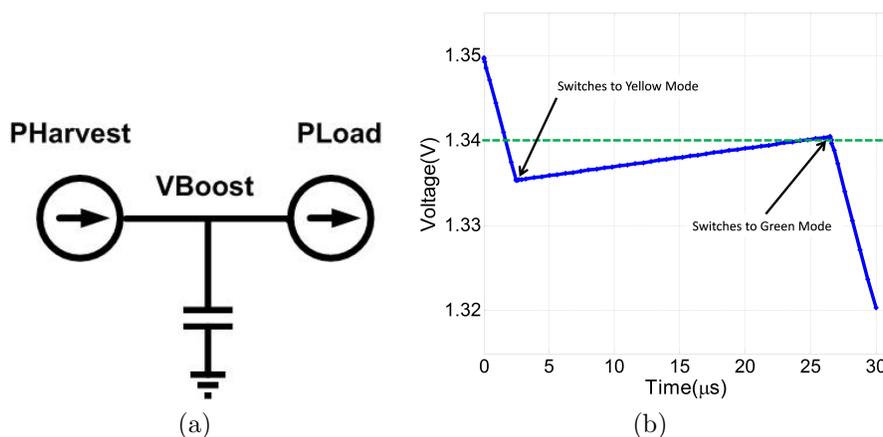


Figure 4.9: a. Simplified model of energy harvester, capacitor, and load. b. Simulated sequence of “lockup.” The DPM switches between green and yellow mode due to transmission.

asymmetric thresholding scheme forces the node to recover more before taking on higher power operations such as transmitting without having to raise the other threshold, thus maximizing the time in the higher mode.

Hysteresis increases the up-switching threshold based on the learned switching pattern. The DPM keeps a history of the previous three operating modes. If the DPM has recognized a switch from a lower operating mode to a high one and back to a lower one (i.e. yellow mode to green mode to yellow), the up-switching asymmetric threshold is incremented by one to prevent lock up.

For flexibility, both modes can be disabled to provide a simple threshold switching. Furthermore, the asymmetric thresholding mode is capable of being used without hysteresis.

#### 4.4.5 Results

The DPM was verified in silicon. Figure 4.10 shows the measured energy of the DPM. The DPM measured energy per operation on the 130nm chip at 0.5V is 2.47pJ (active) and 0.2pJ (idle). The DPM’s RO functionality could not be verified in silicon. Figure 4.3b shows the simulated energy per sample as frequency between the ADC and RO.

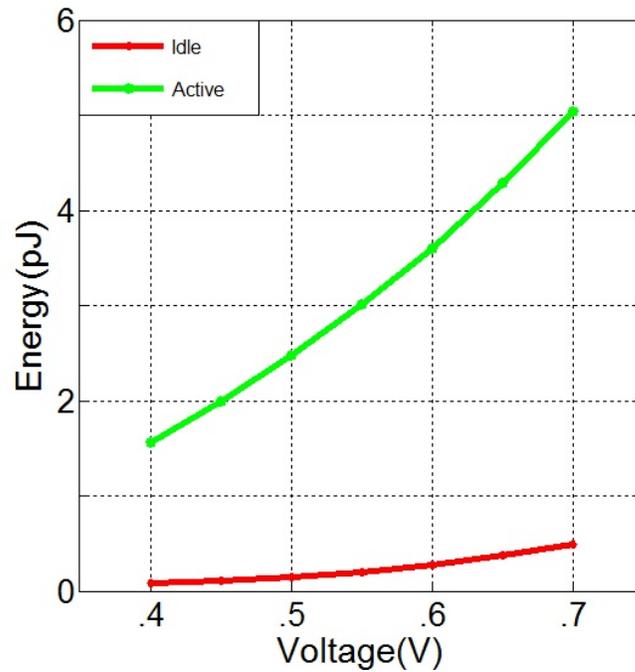


Figure 4.10: Measured DPM revision energy per operation.

## 4.5 Conclusions

This chapter presents groundwork for energy harvesting-specific power management in BSN nodes to adapt to varying amounts of harvested energy to improve lifetime in BSN nodes. We explore the benefits to two methods of sampling the health of the BSN node, increasing the number of operating modes, and single-cycle power modification. We show that ASIC single-cycle power modification provides a 17x energy reduction over a generic microcontrollers. These explorations inform fundamental decisions in power management in the first and second revisions of the DPM as well as future power manager designs.

# Chapter 5

## Improving Battery Lifetime in BSN

### Aggregators

<sup>1</sup> Body sensors networks (BSNs) show great promise for long term, comprehensive, inexpensive, and unobtrusive monitoring of patients and healthy individuals. BSNs could revolutionize healthcare by offering nearly continuous monitoring to provide new levels of medical observation. Additionally, they have the ability to improve the infrastructure of telemedicine, improving access to medical services that would often not be available to patients in rural communities. Though BSNs have tremendous potential for improving healthcare, their practical adoption must overcome technical challenges, such as form factor and device lifetime. BSN nodes will not be used if they require frequent battery changes or charging, uncomfortable, or unsightly [5].

The focus of this dissertation is to improve lifetime in BSN nodes and aggregators to improve the lifetime of the whole BSN. BSN lifetime improvement enables longer term monitoring of first responders in dangerous situations, chronically ill patients for diagnosis and medical treatment, and athletes to ensure safety. This chapter looks specifically to

---

<sup>1</sup>This chapter is based on the published papers titled: "A 32b 90nm Processor Implementing Panoptic DVS Achieving Energy Efficient Operation from Sub-threshold to High Performance" [YS1] and "A 90nm Data Flow Processor Demonstrating Fine Grained DVS for Energy Efficient Operation from 0.25V to 1.2V" [YS7].

improve the lifetime of the BSN aggregator. BSN aggregators play a very important and often under-appreciated role in BSNs. They collect information from the body-distributed BSN nodes and ultimately convey it to other stakeholders such as physicians or first responders. Without these aggregators, BSN nodes would not be able to communicate important patient data with essential stakeholders, thus rendering these BSNs ineffective. Therefore, we must find a way to also improve battery lifetime within BSN aggregators.

Harvesting ambient energy offers an appealing option to improve lifetime. However, aggregators are not capable of operating exclusively from harvested energy due to the inability of current technology to harvest an amount of energy sufficient to accommodate the demands of the aggregator's processing/communication requirement. This requires us to use a battery and find techniques to improve battery life. To extend the lifetime of the aggregator and the BSN as a whole, we can leverage the aggregator's varying workload. This workload varies as function of the amount data that needs to be processed; changes in a BSN node's sampling rate as a result of some trigger (i.e. a cardiac event is detected) or a change in the number of BSN nodes communicating with the aggregator can account for workload variation.

Dynamic Voltage Scaling (DVS) is a common approach to reduce energy consumption to extend battery life in energy constrained systems with varying workloads. Many systems occasionally require high performance, but their varying processing requirements remain below this upper limit for the majority of their lifetimes. This provides an opportunity to reduce the energy of the system when the system is not running at its highest performance, consequently extending the system's lifetime. As the performance requirements decrease, systems without DVS still run at the highest voltage and then idle until the next set of data needs to be processed. In systems that utilize DVS, the supply voltage is throttled down to the lowest voltage possible, reducing the energy consumed quadratically while still meeting its performance requirement.

Traditional DVS implementations suffer from coarse spatial and temporal granularities.

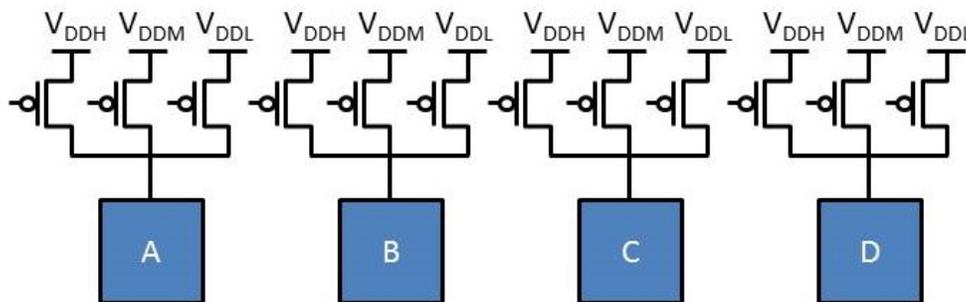


Figure 5.1: Structure of PDVS.

Spatial granularity is the ability to assign different components in a design to different voltages. Most recent DVS implementations are limited to a spatial granularity at the microprocessor core level to entire chip [16] [17] [18]. Spatial granularity is the ability to assign each component to different voltages at one time. Temporal granularity is the ability to assign the same component to multiple voltages over short time periods. Off-chip DC-DC converters traditionally limited temporal granularity as they take tens to hundreds of  $\mu$ seconds to adjust the  $V_{DD}$  [19]. Traditional DVS's coarse spatial and temporal granularity limits the energy efficiency systems like BSN aggregators can achieve.

To improve energy efficiency and lifetime in BSN aggregators, we propose applying a method called Panoptic ("all-inclusive") Dynamic Voltage Scaling (PDVS). PDVS uses multiple PMOS header switches at the component level, shown in Figure 5.1, to provide a local voltage (virtual- $V_{DD}$ ) from a discrete set of chip-wide shared voltages (e.g.,  $V_{DDH}$ ,  $V_{DDM}$ ,  $V_{DDL}$ ) to improve spatial and temporal granularity. This allows for an individual component's virtual- $V_{DD}$  to be set independently of any other component as well as allowing for fast local  $V_{DD}$  switching. The use of voltage dithering [32] (using a division of operations across two voltage/frequency points to approximate an effective intermediate operating point) further enables the approach to closely approximate an ideal energy/performance trade off across a broad range of workloads.

This chapter focuses on improving energy efficiency which leads to an improvement in battery life and therefore, an improvement in energy constrained electronics's lifetime. This

chapter describes a 90nm test chip and simulations used to quantify the savings of PDVS. Using this infrastructure, we explore design decisions for the implementation of PDVS, such as switching methodology and number of resources. In this chapter, we extend the state of the art and knowledge in this area in several ways. We present the first processor that was implemented using the PDVS architecture. To our knowledge, this is the first full processor to implement PDVS, single clock cycle  $V_{DD}$  switching,  $V_{DD}$  dithering, and the ability to switch between high performance DVS operation and a subthreshold mode of operation. Additionally, we quantify the expected energy savings by applying PDVS in resource constrained systems while reducing the number of resources. Lastly, we address  $V_{DD}$  switching methodology to improve the overheads for switching in header-based systems.

## 5.1 Related Work

DVS has become common place in literature and commercial products since being introduced in [68]. DVS is utilized in commercial processors, including the Oracle SPARC T5 processor [29] and Intel Ivy Bridge processor [30]. To extend DVS further, many designers provide multiple frequency and voltage domains to add granularity to DVS. Howard et al [16] utilized eight voltage domains and 24 frequency domains to implement DVS on the core level to manage 32 cores in order to achieve lower energy. This approach has been shown to be popular but requires as many DC-DC converters as there are domains. These DC-DC converters allow each block to achieve a specific voltage but have time, area, and energy overheads.

Other DVS implementations look to limit the number of DC-DC converters. Three implementations show examples of this. Truong et al [17] demonstrated a 167 core implementation in which each core can be connected to one of two voltages through headers and has an independent frequency for each core. Calhoun et al [69] proposed a concept called Ultra-Dynamic Voltage Scaling (UDVS). UDVS uses local voltage dithering, which utilizes two voltages to generate intermediate rates and subthreshold operation to achieve DVS operating voltage

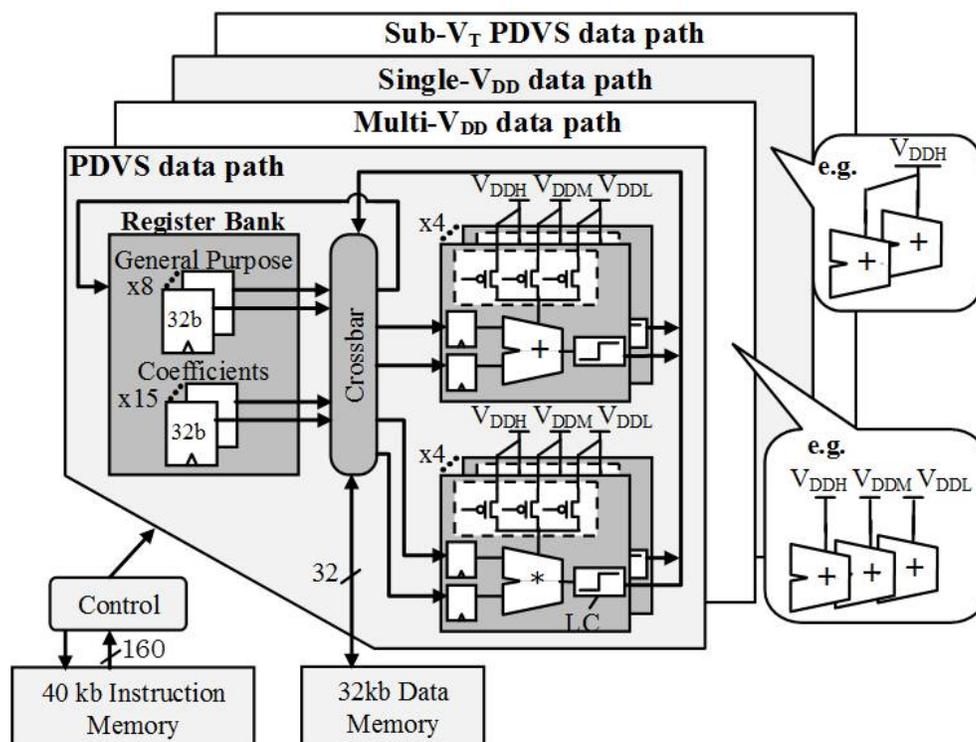


Figure 5.2: Block diagram of the PDVS data flow processor. SRAMs and control serve four data paths for direct comparison of PDVS with  $SV_{DD}$  and  $MV_{DD}$  [10].

range of subthreshold to superthreshold voltages. PDVS [33] [20] builds upon this work and concepts of UDVS. PDVS introduces headers and allows for more flexibility in spatial and temporal granularity to allow parts of the system to lower voltages for smaller periods of time to reduce power. Prior to this work, PDVS has not been demonstrated on a large system, such as a processor.

## 5.2 Test Chip

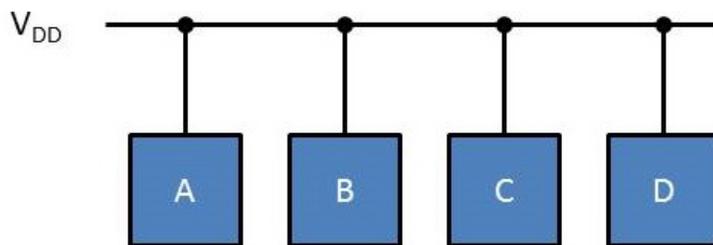
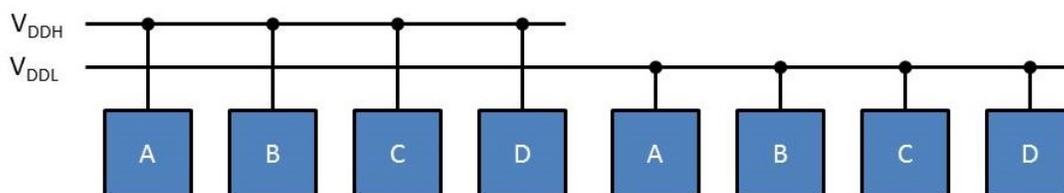
To explore the full benefits of PDVS and compare it to other implementations of DVS, we designed a 32bit data flow processor, shown in Figure 5.2, in a commercial 90nm technology. This processor is capable of executing arbitrary data flow graphs (DFGs) at 1 GHz at 1.2V. We used the PDVS architecture to implement the data path of the processor. The data path consists of four Baugh-Wooley multipliers and four Kogge-Stone adders. Each of these

Table 5.1: PDVS Chip Summary

Feature	This Chip
Process	90nm CMOS Bulk w/ Dual VT
Area	4.3mm x 3.3mm
Transistor Count	Approximately 2 million
$V_{DD}$	250mV to 1.2V
Memory	40kb & 32kb

components uses three PMOS header switches tied to the three  $V_{DD}$ s ( $V_{DDH}$ ,  $V_{DDM}$ ,  $V_{DDL}$ ) that are common throughout the processor. The processor includes a programmable crossbar that feeds input registers of the data path components either directly from the datapath, the register bank, or the memory. To prevent short circuit current from blocks operating below the nominal  $V_{DD}$ , level converters (LCs) are used at the output of each multiplier and adder to up-convert their outputs to the  $V_{DDH}$  level that is used at the register file. These LCs are adapted from [70].

In order to provide a fair hardware comparison to PDVS, we include three additional data paths on the chip that are functionally identical but that use different power management options: single- $V_{DD}$  ( $SV_{DD}$ , Figure 5.3), multi- $V_{DD}$  ( $MV_{DD}$ , Figure 5.4), and a sub- $V_T$  optimized PDVS data path. In the  $SV_{DD}$  data path, the four multipliers and adders all share the same  $V_{DD}$ . In the  $MV_{DD}$  data path, the four multipliers and adders are permanently tied to either  $V_{DDH}$ ,  $V_{DDM}$ , or  $V_{DDL}$ , and operations can be scheduled for execution on any of these components based on the timing requirements. These voltage assignments can be set externally from a voltage source. The processor has a 32kb data memory and a 40kb instruction memory that are shared for all of the data paths. The control word for controlling the data flow (and header control where applicable) of the various data paths is 160b for this test chip.

Figure 5.3: Structure of single- $V_{DD}$ .Figure 5.4: Structure of multi- $V_{DD}$ .

### 5.2.1 Test Setup

In order to test the functionality of our chip and measure the energy consumption of each individual data path, we set up a test platform that generates inputs and compares outputs and allows us to run the same benchmarks on a VHDL model, Spectre netlist, and physical hardware. The benchmarks are developed in VHDL and custom scripts translate them into a Spectre stimulus file and a VHDL state machine for the Spectre simulation and hardware testing, respectively. Only Spectre simulation and the test chip are used to measure energy. We use a custom synthesis script to map the benchmark DFGs to the architecture and use Matlab to create the 160b instruction words. We designed two custom printed circuit boards (PCBs), a test board and a FPGA daughterboard, to test and measure the four different data paths on our test chip. The test board, shown in Figure 5.5a, mates with our chips, to connect voltages to our chip, measure the power consumed for each power domain, and mate to our second PCB, the daughterboard. To provide testing flexibility, the daughterboard contains a Xilinx Spartan-6 XC6SLX150 FPGA, shown in Figure 5.5b. This PCB is used to generate test signals.

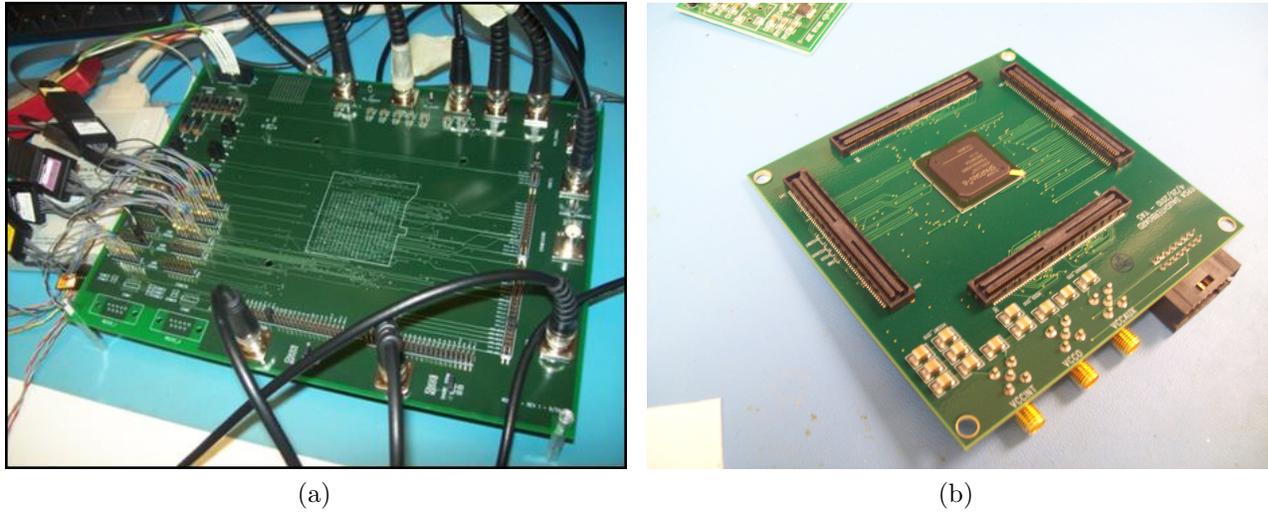


Figure 5.5: Test setup for the PDVS chip. a. Test board. Daughterboard mates with the test board. b. FPGA daughterboard [11].

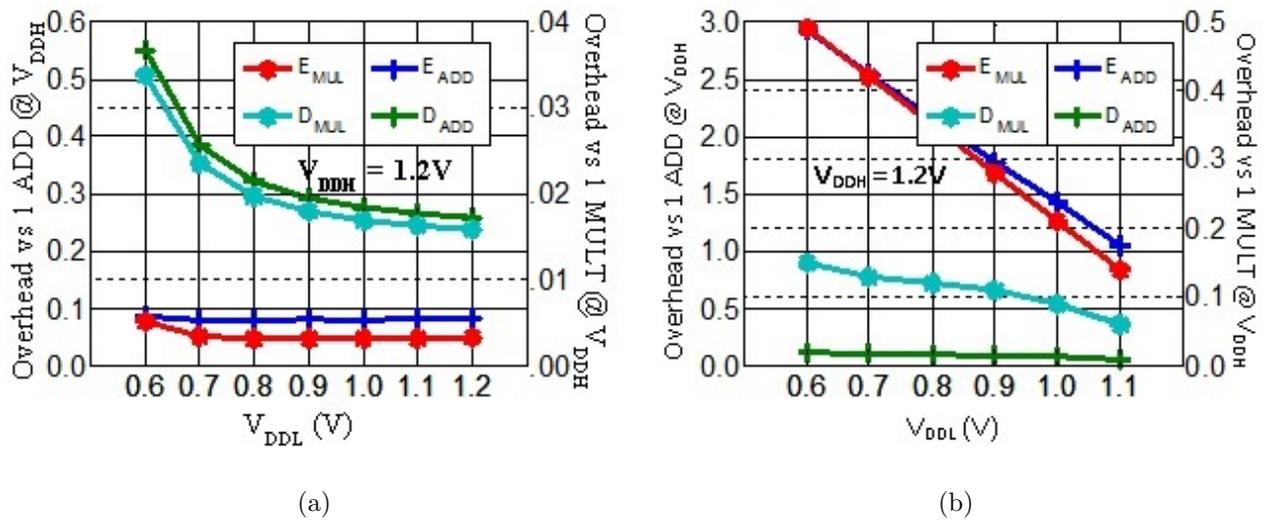


Figure 5.6: a. Simulated level conversion overhead varying  $V_{DDL}$  for both the adder and multiplier. b. Simulated virtual- $V_{DD}$  switching overhead varying  $V_{DDL}$  for both the adder and multiplier [11].

### 5.2.2 PDVS Overheads

There are area, energy and delay overheads for the additional LCs and the headers associated with PDVS compared to  $SV_{DD}$  and  $MV_{DD}$ . The 32b Kogge-Stone adder and Baugh-Wooley multiplier have 2.4% and 1.7% header switch area overhead, and 11.4% and 2.1% level

converter (LC) area overhead, respectively. The LCs have a 32.0% and 2.0% LC simulated delay overhead, and 8.0% and 0.3% LC energy overhead for converting from 0.8V to 1.2V (Figure 5.6a) relative to a single add or multiply operation in  $SV_{DD}$ . The LC overhead is minor compared to the overall timing and energy budget, since the multipliers dominate DFG delay and energy. Additionally, there are energy and delay overheads for recovering the rail when switching from a lower voltage operation to a higher voltage operation. The adder and multiplier have a delay overhead for charging the virtual rail of 10.4% and 12.0% of the normal operation time (Figure 5.6b). There is a 215.3% and 35.0%  $V_{DD}$  switching energy overhead, leading to breakeven times of less than 4 and less than 1 operation for adder and multiplier operations. The energy benefits of PDVS overwhelm these overheads in the benchmark DFGs.

### 5.2.3 Chip Results

Figure 5.7a shows a measured energy per operation of an add and multiply vs.  $V_{DD}$ . Figure 5.7b and 5.7c show the comparison of measured delays and simulated delay of an add and multiply. We were unable to measure faster delays for higher voltages due to limitations of the on-chip voltage controlled oscillator. Simulations and measured delays match closely for lower voltages at which the VCO was able to generate a clean clock. Measured DFG energies, shown in Figure 5.7d, demonstrate PDVS savings across various workloads. For the same area, PDVS gives lower energy for varying workloads than  $MV_{DD}$  by switching components to lower  $V_{DD}$ s when possible. PDVS headers enable  $V_{DD}$  dithering (rapid switching between two  $V_{DD}$  rate pairs) to approximate ideal DVS. Dithering is shown by the line between these points. At a rate of 1, the PDVS and  $MV_{DD}$  curves are slightly lower energy than  $SV_{DD}$  since they remove timing slack in the DFG. This energy profile in Figure 5.7d matches the anticipated savings for PDVS and shows how PDVS with dithering closely approximates the ideal savings achievable if we could provide the perfect voltage for each rate of operation.

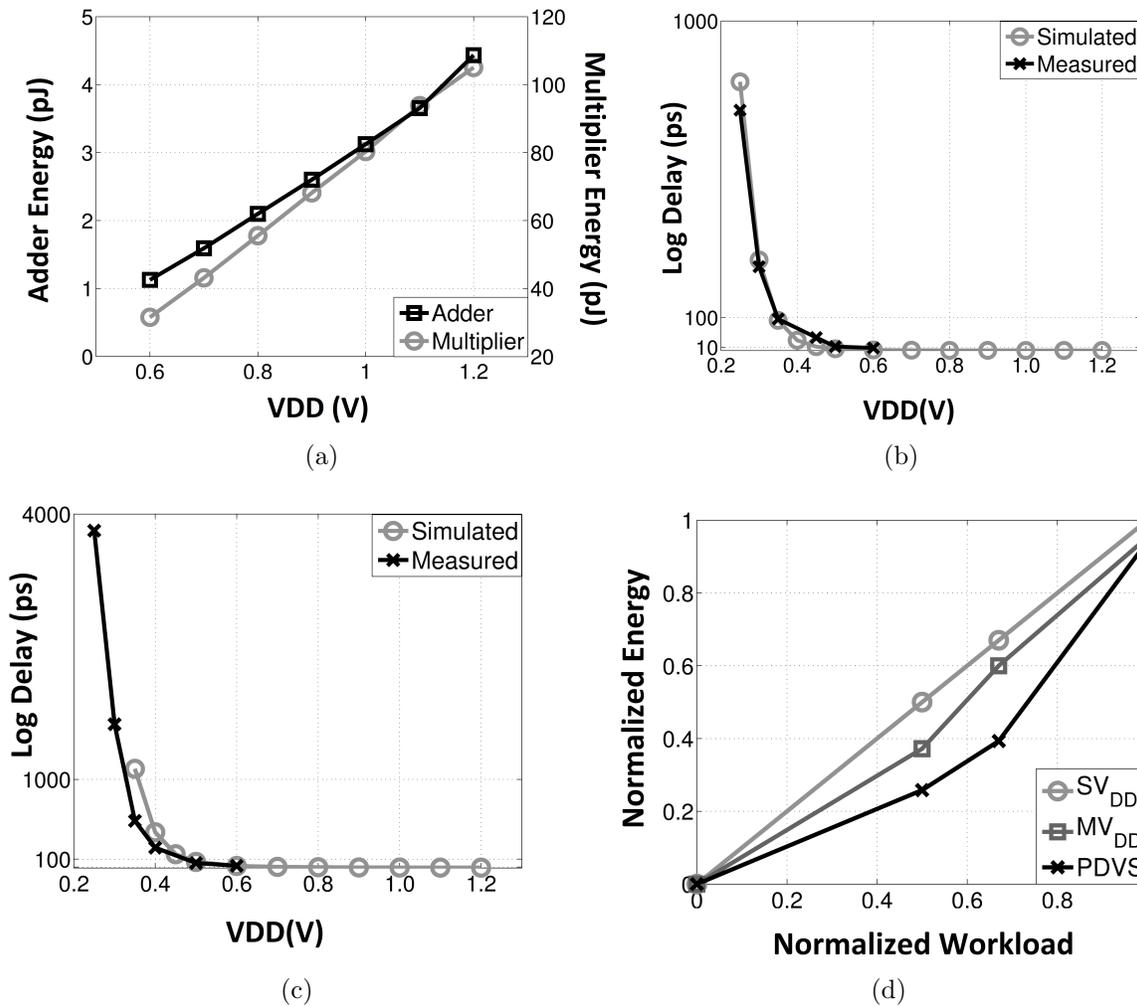


Figure 5.7: a. Measured energy of adder and multiplier vs.  $V_{DD}$  b. Measured vs simulated delay of adder vs  $V_{DD}$  c. Measured vs. simulated delay of multiplier vs  $V_{DD}$  d. Average measured energy (w/ overheads) vs. workload across 4 different DFGs [10].

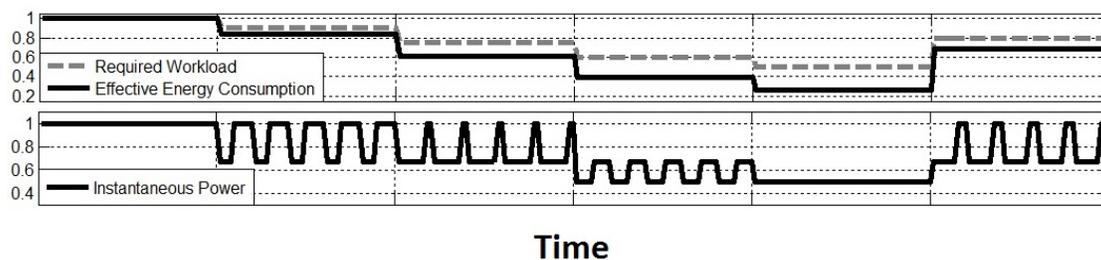


Figure 5.8: Change in average power & instantaneous power as the workload changes over time; power waveform shows dithering between two rates to achieve an intermediate rate, resulting in near optimal average energy savings [10].

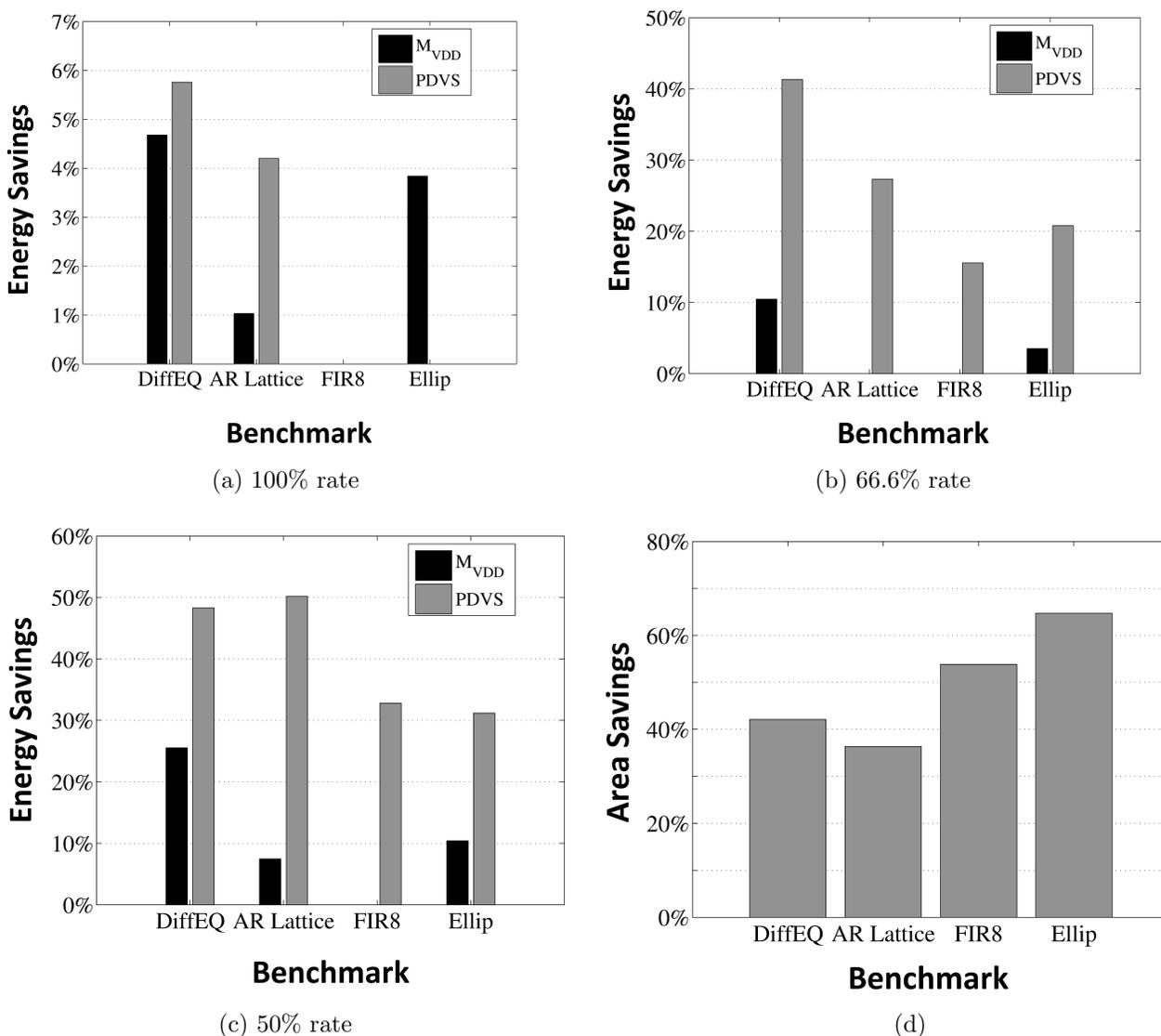


Figure 5.9: a-c. Measured energy benefit (including overhead) of PDVS &  $M_{VDD}$  vs.  $SV_{DD}$  for single function single rate (SFSR) & single function multi rate (SFMR) at 67% and 50% at constant area d. Area benefit of PDVS over  $M_{VDD}$  [10].

Figure 5.8 demonstrates energy savings from dithering for a varying workload, verified by hardware measurements.

Figures 5.9a-c shows results for the benchmark DFGs we ran on all the data paths to demonstrate PDVS's benefits over multiple rates. PDVS has shown more energy savings as the timing constraint of the DFGs is relaxed. The PDVS data path shows up to 50% and 46% measured energy savings over  $SV_{DD}$  and  $M_{VDD}$  for the same area, respectively.  $M_{VDD}$

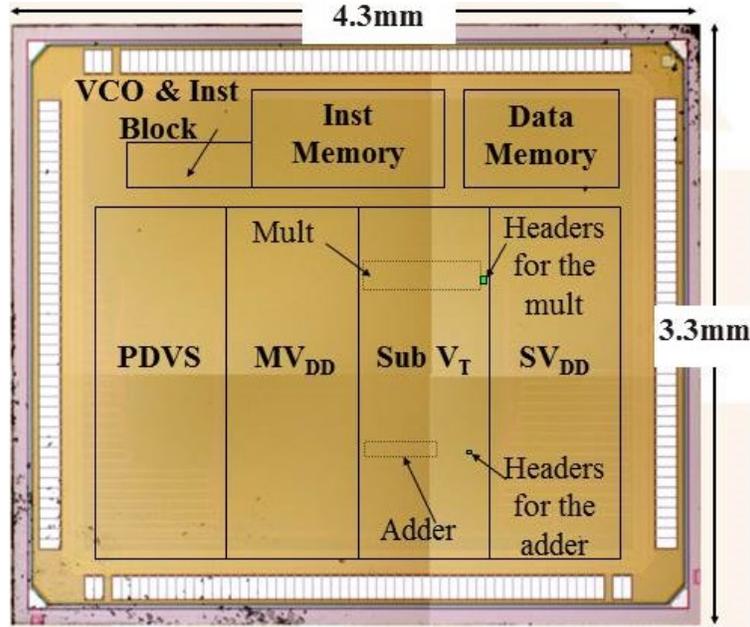


Figure 5.10: Die photo of the 90nm PDVS test chip [10].

can provide the same energy as PDVS for a given DFG by replicating blocks and tying them to different  $V_{DD}$ s. PDVS saves up to 65% area (Figure 5.9d) for the optimal energy schedule by reusing components over  $MV_{DD}$ , since individual components are not statically assigned to a voltage. Table 2 shows the chip summary. Figure 5.10 shows a die photo of the 90nm test chip.

### 5.3 Number of Resources

There are many design "knobs" that affect energy savings DVS within a system. One of the most important knobs is the number of resources. In many systems, this could be the number of processing cores or ALUs. In our system, this is the number of adders and multipliers. This section investigates sweeping the number of resources to get an intuition of PDVS's savings over different system designs. In this section, we sweep the number of resources from 1 to 5 over multiple rates (100%, 150%, 200%, and 300%). We use the 4-tap (FIR4), 8-tap (FIR8), 12-tap (FIR12), and 16-tap (FIR16) FIR filter benchmarks to see how constraining the number of resources available affects the energy over more operations.

### 5.3.1 Test Setup

We start by writing DFGs for each FIR filters, which show the dependencies between different operations. We then assign voltages to components. Voltage rail voltages are constrained to 1.2V, 0.9V, and 0.7V; these voltages were used in the DFGs for Figure 5.9. Voltage selection is not the focus of this experiment. We require that the voltages selected for the components to be capable of reaching the 100% rate in case of any immediate change in the workload. This restriction heavily constrains the voltages chosen for  $MV_{DD}$  since  $MV_{DD}$  voltages are hard-tied. To achieve the 100% rate, the  $MV_{DD}$  system must tie most or all of its components to  $V_{DDH}$  and thus will cannot take advantage of timing slack and run operations at a lower voltage.

We calculate each DFG's energy for each rate while sweeping the resource count. Energy and delay data for each component at each voltage were obtained through Spectre simulation. These calculations account for adder and multiplier active energy and the overheads of LCs, headers, and switching in  $MV_{DD}$  and PDVS; this calculation does not account for any leakage energy or energy from the peripheral circuits (registers, memory, and crossbar).  $SV_{DD}$  and  $MV_{DD}$  energy values were hand calculated. PDVS energy values were calculated by a custom scheduler built by Saad Arrabi. Optimal scheduling in PDVS and  $MV_{DD}$  is an NP-complete problem and is not the focus of this dissertation. We selected the "best slack" algorithm for scheduling. This method looks at the timing slack available and finds the largest slack that is available in the DFG. It selects the lowest possible voltage to take advantage of that timing slack. This process is repeated until all available slack has been analyzed and the lowest energy possible is achieved. We compare the results between  $MV_{DD}$  and PDVS below.

### 5.3.2 Results

The results of this exploration are shown in Figures 5.11, 5.12, and 5.13. All energy values are normalized to the  $SV_{DD}$  energy. Several trends are evident from the data:

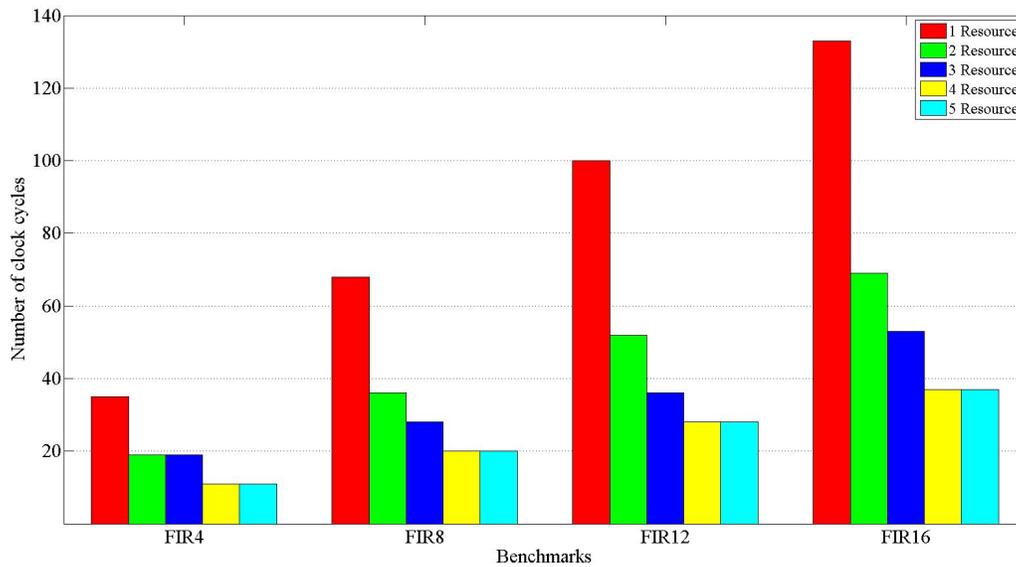
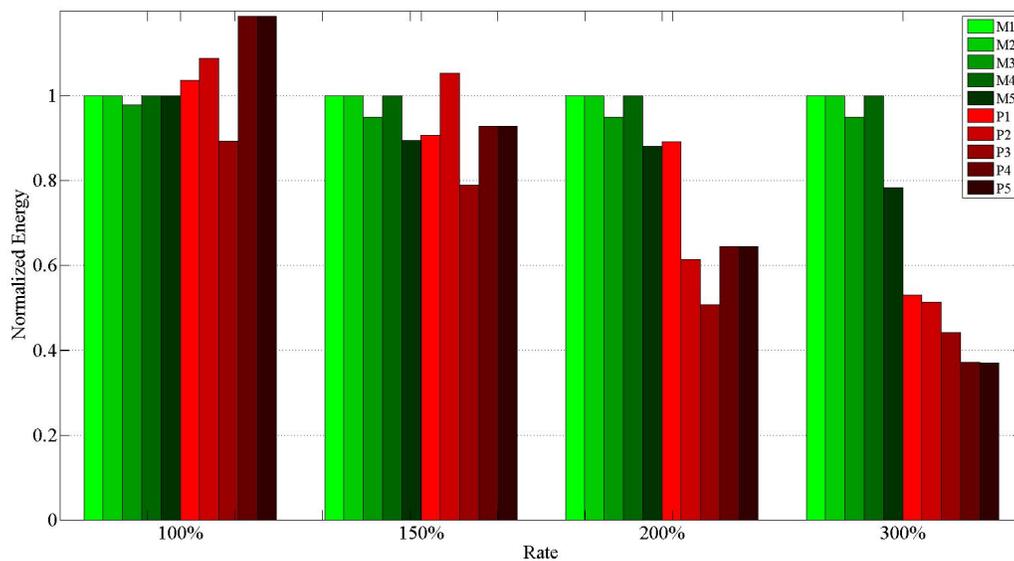
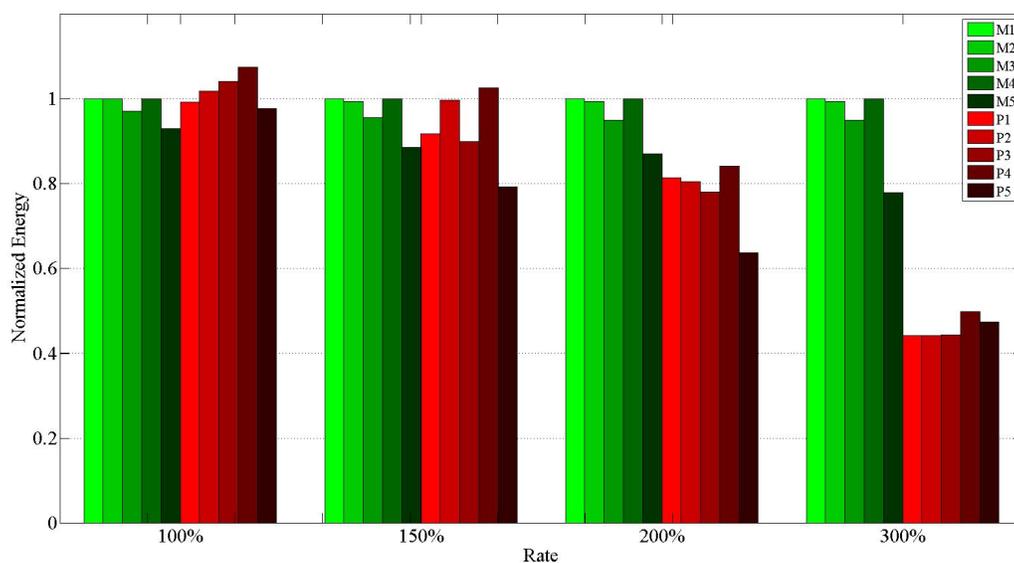


Figure 5.11: Number of clock cycles required for the 100% rate.

- Adding resources does not necessarily decrease energy for the 100% rate for PDVS, but rather improves parallelization and reduces the number of clock cycles required per iteration, as seen in Figure 5.11. However, there is a saturation point in the parallelization. In the selected programs, we show no increase in parallelization between four resources and five resources.
- All of  $MV_{DD}$ 's components are tied to 1.2V when number of taps is divisible by the number of resources. Thus, it achieves a normalized energy of 1 for all rates.
- As the timing requirements are relaxed, PDVS is able to take advantage of the timing slack available and reduce the voltage of the components. The majority of energy savings come from reducing the voltage of the multipliers. FIR filter's energy is dominated by the multiplication energy (91% of  $SV_{DD}$  energy in FIR16).  $MV_{DD}$ 's inability to scale the multipliers voltage in most cases results in no energy savings as the timing slack increases. More multipliers result in the ability to tie more multipliers at a lower voltage to run at lower operations. Additionally as the number of multipliers increase, fewer multipliers have to switch, resulting in a reduction of switching energy.



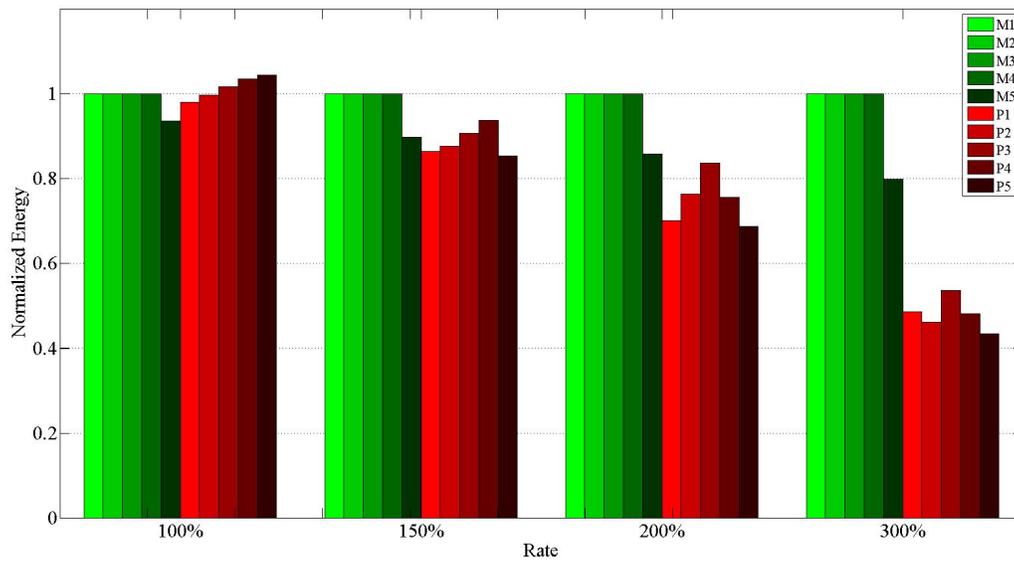
(a) FIR4



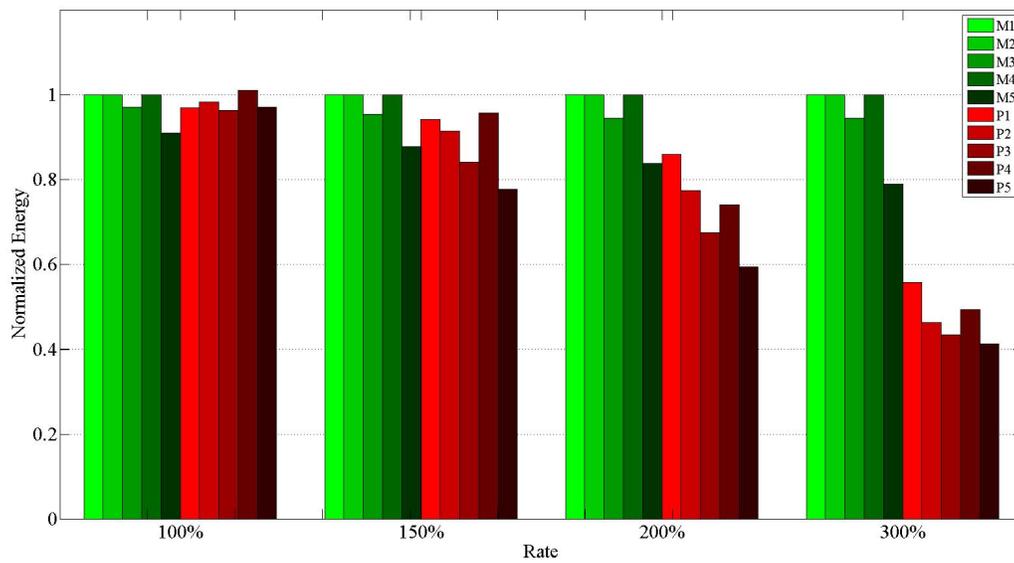
(b) FIR8

Figure 5.12: Simulated active energy of FIR4 and FIR8 benchmarks. Legend: M-  $M_{DD}$ . P- PDVS. Number- number of resources.

- An increase in resources generally correlates to a reduction in active energy within PDVS. This trend is not found in all cases, especially at the 100% and 150% rates. Two reasons for the break in the trend are non-optimal scheduling and the difference in



(a) FIR12



(b) FIR16

Figure 5.13: Simulated active energy of FIR12 and FIR16 benchmarks. Legend: M-  $M_{DD}$ . P- PDVS. Number- number of resources.

the number of clock cycles for the 100% rate between different resource values.

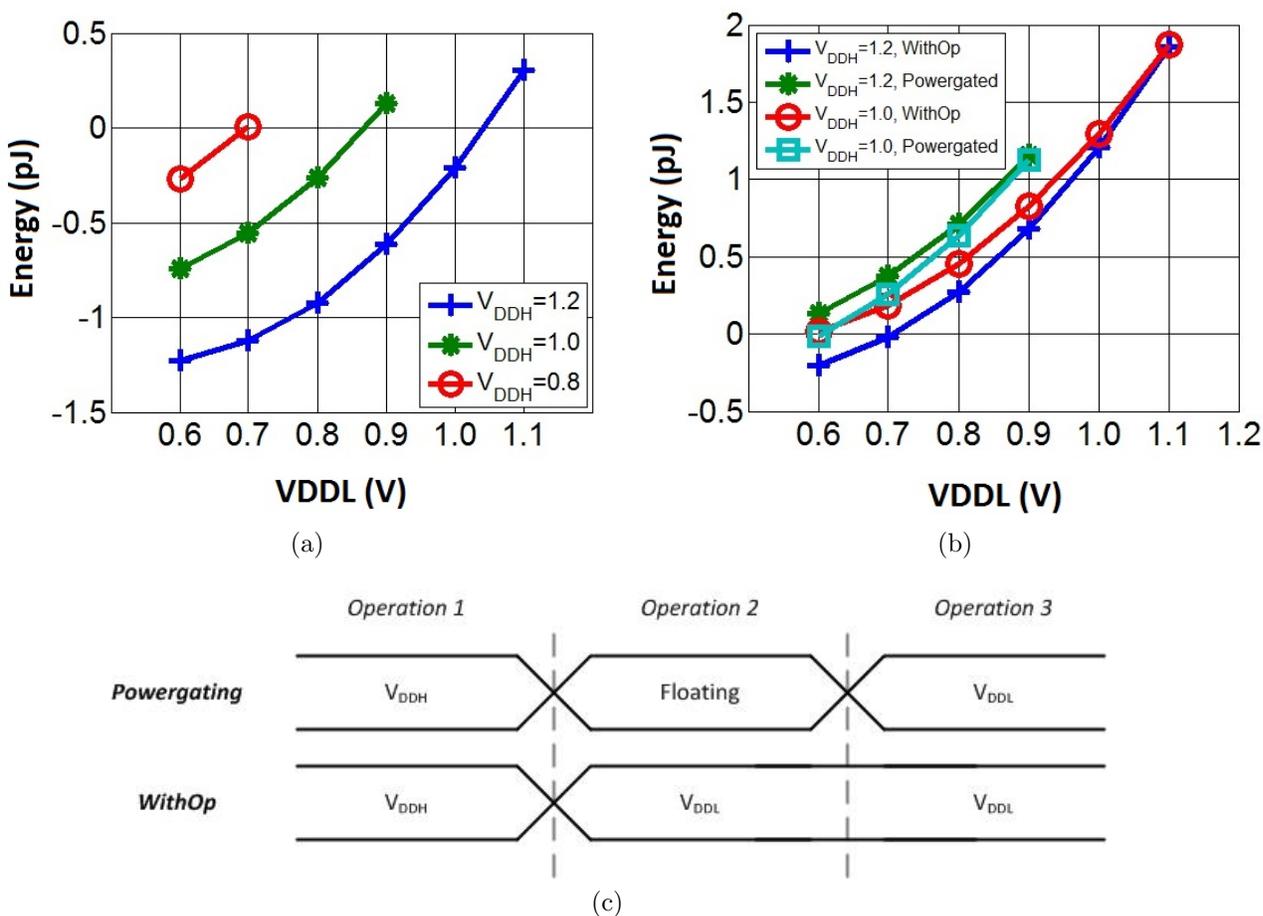


Figure 5.14: a. Simulated energy per operation while switching from  $V_{DDH}$  to  $V_{DDL}$ . b. Simulated energy of running three operations with and without a NOP across various  $V_{DDH}$ s. c. Timing diagram of the switching methodologies [11].

## 5.4 $V_{DD}$ Switching

To improve energy efficiency, PDVS uses headers and three global  $V_{DD}$  rails instead of dedicated block level DC-DC converters. PDVS allows for blocks to change their virtual- $V_{DD}$  through turning on and off headers to achieve lower energy consumption. PDVS's architecture speeds up virtual- $V_{DD}$  switching, allowing PDVS to reduce the voltage for brief changes in workload that cannot be realized in conventional DVS implementations. The delay of switching the virtual- $V_{DD}$  depends on the header size, but for our processor, this time is less than our target clock period.  $V_{DD}$  switching creates noise on the shared supplies, but [16]

shows that this noise can be managed.

PDVS's structure and savings bring about an interesting exploration of the switching control scheme. We separate switching into two cases: switching from a higher voltage operation to a lower voltage operation and switching from a lower voltage operation to a higher voltage operation. For simplicity, we reduce the design into two rails:  $V_{DDH}$  and  $V_{DDL}$ .

When switching from a higher voltage to a lower voltage, if the voltage difference is large enough, the switching will result in a free operation at  $V_{DDL}$ . Figure 5.14a shows the energy per operation consumed across two operations, where the first occurs at a higher voltage and the second occurs at a lower voltage. The negative energy shown in the figure occurs when current enters the source of the  $V_{DDL}$  supply. In our design, this energy would be reused through the power distribution network or would charge the decoupling capacitor at the driving DC-DC converter. We confirmed that this energy can be reused in a simulation. We setup a simulation with two adders and a wire model that connected the voltage rails and saw the energy shared between the two. However, in other designs, this negative energy could be lost, resulting in the operation costing leakage energy.

We investigate two strategies for switching from a higher voltage operation to a lower voltage operation to investigate the conservation of the energy dissipated from the virtual rail as shown in Figure 5.14a: turning off all headers (power gating) and keeping the lower voltage header (WithOp) on during the transition. Figure 5.14b shows the comparison of the two methods over three additions (Power gating:  $V_{DDH}$ , off,  $V_{DDL}$ ; WithOp:  $V_{DDH}$ ,  $V_{DDL}$ ,  $V_{DDL}$ ). For both header connection strategies, we execute operations during all three cycles. WithOp shows an energy savings over the power gated approach by allowing current to flow back into the  $V_{DDL}$  rail instead of severing the connection.

During a switching operation from a low voltage operation to a high voltage operation, an operation needs to be completed and the virtual- $V_{DD}$  rail must be charged to the higher voltage. We investigated two approaches to doing this: allowing the operation to run and rail to recover in the same operation and running a NOP for a single clock cycle to allow time for

the virtual rail to settle to the high voltage and then running the operation. Both methods consume approximately the same amount of energy over the three additions. The NOP approach carries a time penalty for recovery but can reliably execute the second operation in the time constraint.

## 5.5 Summary and Conclusions

This chapter focused on improving lifetime in battery-powered BSN aggregators through PDVS. We explored the benefits of PDVS and the design decision exploration of PDVS of the number of resources and  $V_{DD}$  switching. We fabricated a 90nm PDVS data flow processor that demonstrates single clock cycle  $V_{DD}$ -switching at the component level, integrated  $V_{DD}$  dithering for near optimal energy scalability. We also showed the benefits of energy savings over  $MV_{DD}$  through the use of PDVS headers. Compared to DVS implementations that must change the output voltage of a DC-DC converter, our fine-grained voltage scaling allows the chip to save energy for rapid variations in workload down to the single operation level. We demonstrated measured energy savings in benchmark DFGs of up to 50% and 46% over  $SV_{DD}$  and  $MV_{DD}$  for a minimal area overhead. This processor demonstrates the potential for larger energy efficient systems and SoCs using PDVS. Additionally, we show a comparison of our chip to other state of the art in Table 5.2. We show the increased parallelization and its effect on energy savings. Lastly, we show the ability to reuse energy from  $V_{DD}$  switching from a higher voltage operation to a lower voltage operation. The application of PDVS will result in a longer lifetime compared to  $SV_{DD}$  and  $MV_{DD}$  alternatives in energy constrained, battery powered electronics like BSN aggregators.

Table 5.2: DVS State of the Art Implementation Comparisons [10]

Feature	[16]	[17]	[18]	This Work
$V_{DD}$ Granularity	6 cores	1 core	1 core	Add,Mult
Speed of $V_{DD}$ Change	$>10\mu s$ e.g. [19]	2–5ns	$>10\mu s$ e.g. [19]	1ns
$V_{DD}$ Dithering	No	No	No	Yes
Sub- $V_T$ operation	No	No	No	Yes

# Chapter 6

## Conclusions

Body sensor networks (BSNs) have the ability to revolutionize the medical field through long term monitoring of chronically ill patients, enabling telemedicine to provide new levels of medical care in the rural population, augmenting bodily functions through drug delivery, and supporting the movement of prosthetic limbs [12]. However, there multiple challenges for their widespread adoption: lifetime and form factor. This thesis contributes to improving lifetime while maintaining a desired form factor through three ways: enabling energy harvesting in BSN nodes, implementing power management in energy harvesting BSN nodes, and implementing fine-grained dynamic voltage scaling (DVS) in BSN aggregators. Though this work targets BSNs specifically, the principles of low power node design, tight system integration, energy harvesting-specific power management, and fine-grained dynamic voltage scaling are applicable to wireless sensor networks.

### 6.1 Summary of Contributions

#### Enabling energy harvesting in BSN nodes

- Demonstrated a low power, state-of-the-art BSN node capable of running solely off harvested energy in silicon.

- Explored the benefits and overheads of using a custom microcontroller in place of a generic microcontroller in accelerator-based systems at low utilization.
- Demonstrated the benefits of global decoding scheme when global routing isn't prohibitive.

### **Implementing Power Management in Energy Harvesting BSN nodes**

- Implemented the first energy harvesting-specific power management system in silicon.
- Implemented a flexible, energy efficient power manager capable of being used on multiple energy harvesting BSN nodes in silicon.
- Quantified the benefits of single cycle power modification.
- Explored the benefits of using ring oscillator vs. ADC for sampling node health.
- Demonstrated the number of operating modes is not a strong "knob" for energy/power in power managers.

### **Improving Battery Lifetime in BSN Aggregators**

- Demonstrated a data flow processor using PDVS in silicon
- Demonstrated PDVS's energy savings compared to single- $V_{DD}$  and multi- $V_{DD}$  alternatives in silicon.
- Demonstrated PDVS's energy savings compared to single- $V_{DD}$  and multi- $V_{DD}$  alternatives in simulation in resource constrained system.
- Explored benefits and overheads of different sequences  $V_{DD}$ -switching in header-based DVS schemes.

## 6.2 Team and Individual Contributions

Much of the work done in this dissertation was done in teams with each individual contributing to the project to make it more impactful. The state of the art batteryless BSN node described in Section 3.2 was a design collaboration between the University of Virginia and University of Washington, requiring a lot of coordination for tight system integration and ULP component design. The UVA BSN chip team consisted of Alicia Klinefelter, Jim Boley, Aatmesh Shrivastava, Yanqing Zhang, and me. My individual contributions on this chip are the design of the DPM and with Yanqing Zhang, co-design the chip architecture. The BSN digital architecture and power management explorations are also my individual contributions.

The PDVS data flow processor described in Section 5.2 was designed by a different team. The PDVS student team consisted of Saad Arrabi, Kyle Craig, Sudhanshu Khanna, and me. I was responsible for the design and layout of the 32b Baugh Wooley multiplier, the clocking scheme, the control scheme, and investigating multiple design knobs. The team shared in the chip architecture design and assisted in simulating and testing the chips. The additional explorations of  $V_{DD}$ -switching and sweeping the number of resources are also my individual contributions.

### 6.2.1 Broad Impact of this Work

Energy efficiency is the most critical metric in many modern integrated circuits. The motivations to lower energy ranges over a range of electronic performance such as low, medium, and high performance. This motivation includes enabling battery-less operation in wireless sensor networks (WSNs) and BSNs for potentially an indefinite operation lifetime, maximizing battery life in mobile platforms such as cellular phones, and managing thermals in processor cores.

This dissertation shows improvements in BSN lifetime through improving energy efficiency in BSN nodes and BSN aggregators. The techniques applied in this work can be applied to

other energy-constrained electronics to improve lifetime. While energy efficiency is important in all applications, this section looks at applying the research discussed in this dissertation to low performance and medium performance energy constrained electronics due to their lifetime constraints.

Many low performance electronics such as wireless environmental sensors require long lifetimes, often being deployed in remote places in which its battery cannot be recharged or replaced. Energy harvesting is a viable alternative power supply to batteries due to their low power consumption. Providing a longer or infinite lifetime in these WSNs improves the length of important data acquisition to better understand the environmental conditions and provides an economic benefit by reducing the number of replacement nodes required to be deployed in the field. The discussed techniques of utilizing tight system integration of ultra-low power blocks and on-node processing through data accelerators to reduce on data transmission rate can be applied to these WSNs to extend battery lifetime and enable energy harvesting for a potentially infinite lifetime. Additionally, if WSN are self-powered, the principles of energy harvesting-specific power management applied within the Digital Power Management (DPM) can be applied to WSNs to help it adapt to varying amounts of harvested energy and extend its lifetime.

Many low and medium performance electronics such as WSNs, BSN nodes, cellular phones, and multimedia processors have applications that require high performance. However, their workload requirements remain below this upper limit for the majority of their lifetime. Most of these devices are energy constrained and use a battery as a power source. Improvements in energy efficiency in this area will lead to better a battery lifetime. Utilizing PDVS in systems with variable workloads can improve energy efficiency and, thus, improve lifetime.

### 6.2.2 Conclusions and Open Problems

The work in thesis describes advances in BSN lifetime through improvements in lifetime of both BSN components: the node and aggregator. Hardware/simulation results and

explorations in the areas of BSN architecture, energy harvesting-specific management, and PDVS provide BSN designers with techniques and recommendations to improve the lifetime of the whole BSN, enabling longer term monitoring to improve the effectiveness of BSNs. This section offers the key lessons and conclusions for each of these areas as well as and discusses opportunities for future work.

This work demonstrates the first wireless biosignal acquisition chip BSN node powered solely from energy harvested from body heat and/or RF power with integrated supply regulation, analog front end, power management, digital signal processing, and transmitting. Tight system integration of low power blocks and heavy duty cycling of the radio allows us to reduce the average power consumption to  $19\mu\text{W}$ . Using this node as a platform to explore digital architecture, we see that idling efficiently is an important metric in microcontrollers in accelerator-based systems that should be optimized. In our use case, utilizing a custom microcontroller with efficient NOPs reduces the energy by 74% and reduces the number of instructions by 100x, resulting in needing less memory.

Further reduction in node power consumption can enable new BSN applications and improve robustness to surviving deadzones and reductions in energy harvesting. To reduce power in these node and create a more functional system, tighter system integration and holistic design are key. Off-chip communication and utilizing COTS components (such as radio chips) can dominant these nodes' power budget; tight integration will reduce the need for off-chip communication. A fully integrated solution will remove the need for any COTS components, thus saving a substantial amount of power. Building a low power BSN node requires a holistic approach. The traditional design paradigm of hardware, RF, and software engineers working separately will no longer be adequate in this power-constrained design space. Designers need to analyze issues at every layer of the hierarchy and iterate to see how issues and requirements at each design layer abstraction effects each other.

On the circuit/microarchitectural side, we can reduce power through leveraging the system requirements. BSNs that target electrocardiography (ECG), pulse oximetry, and

blood pressure have slow sampling requirements, providing many opportunities to save energy. Simple knobs have yet to be optimized; these include reduction of clock frequency, dynamic voltage scaling (DVS) and leakage reduction techniques such undersizing powergate headers/footers and placing blocks in a retentive node. Additionally, high powered blocks such as radios and the analog front end need to be evaluated differently on the system level. High power blocks that can duty-cycle for longer periods of time may be more advantageous than lower power blocks for this space.

Several open problems remain in the field of ASIC BSN SoCs. First, there are no system-level models available to do holistic design. These systems must account for interactions between the lower level hardware up to software to account for the impact of decisions on all layers. Second, there is no current methodology to choose what functionality should be assigned to ASIC accelerators. ASIC accelerators are important in reducing system power and can provide a 6800x reduction in energy. The trade-off between energy efficiency and area/cost has yet to be study. The problem becomes more difficult in BSN platform designs where the platform has multiple applications. Fourth, DVS is a common scheme to reduce energy. The fabricated node utilizes a fine-grained DVS scheme. An analysis of the benefits and overheads of fine-grained DVS and global DVS would help inform future BSN node design.

Low node power consumption in energy harvesting-powered BSN nodes does not ensure a long node lifetime; harvested energy varies as a result of the node's environment. Failing to adapt to this varying energy will result in node death, rendering the BSN useless and losing important biosignal data. This work presents the first and second revisions of an on-chip power management system for an ultra-low power batteryless energy harvesting body sensor node BSN SoC. Through this work's power management design space exploration, we see that single cycle-power reduction is capable of an 17x energy reduction over the use of a generic microcontroller in a transition from all blocks on to all blocks off. Furthermore, the use of generic microcontrollers as power managers will result in a lot of wasted energy compared to

single-cycle power modification schemes, possibly dropping the storage capacitor's voltage to an unsustainable level that results in node death. Therefore, an ASIC dedicated power manager capable of single-cycle power modification will result in the highest energy savings and the longest node preservation. Through another exploration, we see that the ring oscillator provides 6x power reduction to sampling node health than an ADC; however, if a BSN node requires an ADC for its application, timing-multiplexing the ADC is more energy efficient than running the ADC and ring oscillator in parallel.

Power management in ultra low power energy harvesting-powered nodes is a new field. To the best of our knowledge, no other such systems have been implemented, providing many directions to go in. This work's approach to power management is to keep the scheme simple. This research has many interesting open questions. First, there is no form of prediction in this power manager. Prediction hardware in BSN COTS nodes and wireless sensor nodes is computationally expensive. Finding an optimal balance between effective prediction and low power will improve robustness and improve lifetime in BSN nodes. Second, designing a methodology to select the optimal sampling rate of node health would result in a better node lifetime. This sampling rate is a function of node power consumption, storage capacitor size, and energy harvesting mechanism. Third, we use the voltage on the storage capacitor to sample node health. Looking at other indicators of node health such as input voltage from the energy harvesting might prove useful for power management and help extend lifetime.

This work presents a fabricated 90nm PDVS data flow processor that demonstrates single clock cycle  $V_{DD}$ -switching at the component level, integrated  $V_{DD}$  dithering for near optimal energy scalability. We demonstrate measured energy savings in benchmark DFGs of up to 50% and 46% over  $SV_{DD}$  and  $MV_{DD}$  for a minimal area overhead. This processor demonstrates the potential for larger energy efficient systems and SoCs using PDVS. This node serves as a platform for more PDVS explorations. Through our explorations, we see that FIR filters energy is dominated by the number of multiplications (91% of  $SV_{DD}$  energy in FIR16).  $MV_{DD}$ 's inability to scale the multipliers voltage in most cases results in no energy savings

as the timing slack increases; PDVS is able leverage its temporal granularity to connect the multipliers the lowest voltage possible to save energy.

PDVS has shown a lot of promise to improve energy efficiency. There are many open questions with PDVS. First, the implemented processor uses microcode to assign the values to headers. Designing and implementing a closed-loop, dynamic scheduler that adjusts header control based on the desired rate would enable processors to use PDVS. The method for implementing this scheduler is still an open ended question. Second, PDVS was applied to a data flow architecture. Exploring the benefits and overheads from implementing PDVS on a traditional, pipelined architecture would allow us to better understand PDVS's potential benefits. Third, no header sizing methodology exists for this scheme. Undersizing a header would result in the virtual-voltage drooping during operation and increased voltage switching time. Larger header sizes would result increased leakage. A methodology that accounts for these two considerations is required to inform future designs.

# Appendix A

## Acronyms

**ADC** analog-to-digital converter

**AFE** analog front end

**Afib** atrial fibrillation

**ASIC** application specific integrated circuit

**BSN** body sensor network

**CISC** complex instruction set computing

**COTS** commercial off-the-shelf

**DAC** digital-to-analog converter

**DFG** data flow graph

**DMEM** data memory

**DPM** digital power management

**DVS** dynamic voltage scaling

**ECG** electrocardiography

**EEG** electroencephalography

**EMG** electromyography

**ENV DET** envelope detector

**FFT** fast Fourier transform

**FIFO** first in, first out

**FIR** finite impulse response

**FPGA** field-programmable gate array

**GPP** general purpose processor

**IMEM** instruction memory

**ISA** instruction set architecture

**LC** level converter

**MCU** microcontroller

**MICS** medical implant communication service

**PDVS** panoptic dynamic voltage scaling

**PLL** phase-locked loop

**POR** power-on reset

**RF** radio frequency

**RISC** reduced instruction set computing

**RO** ring oscillator

**ROM** read-only memory

**SAR** successive-approximation-register

**SoC** system on a chip

**SRAM** static random-access memory

**Sub- $V_T$**  subthreshold

**TEG** thermoelectric generator

**TX** transmitter

**ULP** ultra low power

**$V_{\text{Boost}}$**  storage capacitor voltage

**VCO** voltage-controlled oscillator

**VGA** variable gain amplifier

**$V_{\text{GS}}$**  gate to source voltage

**$V_{\text{Kill}}$**  critical voltage

# Appendix B

## Publications

- YS1** K. Craig, S. Arrabi, Y. Shakhsheer, S. Khanna, et al., “A 32b 90nm Processor Implementing Panoptic DVS Achieving Energy Efficient Operation from Sub-threshold to High Performance.” *Journal of Solid State Circuits*, Accepted for Publication.
- YS2** Y. Zhang, F. Zhang, Y. Shakhsheer, J. D. Silver, A. Klinefelter, et al., “A Batteryless 19 uW MICS/ISM-Band Energy Harvesting Body Sensor Node SoC for ExG Applications,” *Journal of Solid State Circuits*, vol. 48, issue 1, pp. 199-213, January 2013.
- YS3** Y. Shakhsheer, Y. Zhang, B. Otis, and B. H. Calhoun, “A Custom Processor for Node and Power Management of a Battery-less Body Sensor Node in 130nm CMOS,” *Custom Integrated Circuits Conference*, September 2012.
- YS4** K. Craig, Y. Shakhsheer, and B. H. Calhoun, “Optimal Power Switch Design for Dynamic Voltage Scaling from High Performance to Subthreshold Operation,” *International Symposium on Low Power Electronics and Design*, September 2012.
- YS5** K. Craig, Y. Shakhsheer, S. Khanna, S. Arrabi, J. Lach, B. H. Calhoun, and S. Kosonocky, “A Programmable Resistive Power Grid for Post-Fabrication Flexibility and Energy Tradeoffs,” *International Symposium on Low Power Electronics and Design*, September 2012.

- YS6** F. Zhang, Y. Zhang, J. Silver, Y. Shakhsheer, M. Nagaraju, A. Klinefelter, J. Pandey, J. Boley, E. Carlson, A. Shrivastava, B. Otis, and B. H. Calhoun, “A Battery-less 19W MICS/ISM-Band Energy Harvesting Body Area Sensor Node SoC,” International Solid-State Circuits Conference, February 2012.
- YS7** Y. Shakhsheer, S. Khanna, K. Craig, S. Arrabi, J. Lach, and B. H. Calhoun, “A 90nm Data Flow Processor Demonstrating Fine Grained DVS for Energy Efficient Operation from 0.25V to 1.2V,” Custom Integrated Circuits Conference, September 2011.
- YS8** K. Craig, Y. Shakhsheer, S. Khanna, and B. H. Calhoun, “Optimal Power Switch Design for Panoptic Dynamic Voltage Scaling Enabling Sub-threshold Operation,” Subthreshold Microelectronics Conference, September 2011.
- YS9** Y. Zhang., Y. Shakhsheer, A. T. Barth, H. P. C. Jr., S. A. Ridenour, M. A. Hanson, J. Lach, and B. H. Calhoun, “Energy Efficient Design for Body Sensor Nodes,” Journal of Low Power Electronics and Applications, April 2011.
- YS10** B. H. Calhoun, Y. Zhang, S. Khanna, K. Craig, Y. Shakhsheer, J. Lach, “A Sub-Threshold FPGA: Energy-Efficient Reconfigurable Logic,” GOMACTech, March 2011.
- YS11** W. Eberhardt, Y. Shakhsheer, and B. Calhoun, “A Bio-Inspired Artificial Whisker for Fluid Motion Sensing with Increased Sensitivity and Reliability,” IEEE Sensors, October 2011.
- YS12** S. Khanna, K. Craig, Y. Shakhsheer, S. Arrabi, J. Lach, and B. H. Calhoun, “Stepped Supply Voltage Switching for Energy Constrained Systems,” International Symposium on Quality Electronic Design, March 2011.
- YS13** B. H. Calhoun, S. Arrabi, S. Khanna, Y. Shakhsheer, K. Craig, J. Ryan, and J. Lach, “REESES: Rapid Efficient Energy Scalable ElectronicS,” GOMACTech, March 2010.

**YS14** J .Stocking, W. Eberhardt, Y. Shakhsheer, J. Paulus, M. Appleby, and B. H. Calhoun, “A Capacitance-Based Whisker-like Artificial Sensor for Fluid Motion Sensing,” IEEE Sensors, October 2010.

# Bibliography

- [1] S. Roundy, E.S. Leland, J. Baker, E. Carleton, E. Reilly, E. Lai, B. Otis, J.M. Rabaey, P.K. Wright, and V. Sundararajan. Improving power output for vibration-based energy scavengers. *Pervasive Computing, IEEE*, 4(1):28–36, 2005.
- [2] Fan Zhang, Yanqing Zhang, J. Silver, Y. Shakhshsheer, M. Nagaraju, A. Klinefelter, J. Pandey, J. Boley, E. Carlson, A. Shrivastava, B. Otis, and B. Calhoun. A batteryless 19 uw mics/ism-band energy harvesting body area sensor node soc. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 298–300, 2012.
- [3] M.A. Hanson, H.C. Powell, A.T. Barth, K. Ringgenberg, B.H. Calhoun, J.H. Aylor, and J. Lach. Body area sensor networks: Challenges and opportunities. *Computer*, 42(1):58–65, 2009.
- [4] Adam T. Barth, Mark A. Hanson, Harry C. Powell Jr., and John Lach. Tempo 3.1: A body area sensor network platform for continuous movement assessment. In *Proceedings of the 2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks, BSN '09*, pages 71–76, Washington, DC, USA, 2009. IEEE Computer Society.
- [5] Yanqing Zhang, Yousef Shakhshsheer, Adam T. Barth, Harry C. Powell Jr., Samuel A. Ridenour, Mark A. Hanson, John Lach, and Benton H. Calhoun. Energy efficient design for body sensor nodes. *Journal of Low Power Electronics and Applications*, 04/2011 2011.
- [6] G. Chen, S. Hanson, D Blaauw, and D Sylvester. Circuit design advances for wireless sensing applications. *Proceedings of the IEEE*, 98(11):1808–1827, 2010.
- [7] Yanqing Zhang, Fan Zhang, Y. Shakhshsheer, J.D. Silver, A. Klinefelter, M. Nagaraju, J. Boley, J. Pandey, A. Shrivastava, E.J. Carlson, A. Wood, B.H. Calhoun, and B.P. Otis. A batteryless 19 w mics/ism-band energy harvesting body sensor node soc for exg applications. *Solid-State Circuits, IEEE Journal of*, 48(1):199–213, 2013.
- [8] Y. Shakhshsheer, Y. Zhang, B. Otis, and B.H. Calhoun. A custom processor for node and power management of a battery-less body sensor node in 130nm cmos. In *Custom Integrated Circuits Conference (CICC), 2012 IEEE*, pages 1–4, 2012.
- [9] Harvard-MIT. Mit-bih database, 2005.

- [10] Y. Shakhsheer, S. Khanna, K. Craig, S. Arrabi, J. Lach, and B.H. Calhoun. A 90nm data flow processor demonstrating fine grained dvs for energy efficient operation from 0.25v to 1.2v. In *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, pages 1–4, 2011.
- [11] Kyle Craig, Yousef Shakhsheer, Saad Arrabi, Sudhanshu Khanna, John Lach, and Benton H. Calhoun. A 32b 90nm processor implementing panoptic dvs achieving energy efficient operation from sub-threshold to high performance. *Journal of Solid State Circuits*, Submitted.
- [12] B.H. Calhoun, J. Lach, J. Stankovic, D.D. Wentzloff, K. Whitehouse, A.T. Barth, J.K. Brown, Qiang Li, Seunghyun Oh, N.E. Roberts, and Yanqing Zhang. Body sensor networks: A holistic approach from silicon to users. *Proceedings of the IEEE*, 100(1):91–106, 2012.
- [13] A. Burns, B.R. Greene, M.J. McGrath, T.J. O’Shea, B. Kuris, S.M. Ayer, F. Stroi-escu, and V. Cionca. Shimmer - a wireless sensor platform for noninvasive biomedical research. *Sensors Journal, IEEE*, 10(9):1527–1534, 2010.
- [14] S. Rai, J. Holleman, J.N. Pandey, F. Zhang, and B. Otis. A 500 uw neural tag with 2 uvrms afe and frequency-multiplying mics/ism fsk transmitter. In *Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International*, pages 212–213, 213a, 2009.
- [15] N. Oliver and F. Flores-Mangas. Healthgear: a real-time wearable system for monitoring and analyzing physiological signals. In *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on*, pages 4 pp.–64, 2006.
- [16] J. Howard, S. Dighe, Y. Hoskote, S. Vangal, D. Finan, G. Ruhl, D. Jenkins, H. Wilson, N. Borkar, G. Schrom, F. Paillet, S. Jain, T. Jacob, S. Yada, S. Marella, P. Sali-hundam, V. Erraguntla, M. Konow, M. Riepen, G. Droege, J. Lindemann, M. Gries, T. Apel, K. Henriss, T. Lund-Larsen, S. Steibl, S. Borkar, V. De, R. Van der Wijn-gaart, and T. Mattson. A 48-core ia-32 message-passing processor with dvfs in 45nm cmos. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 108–109, 2010.
- [17] D. Truong, W. Cheng, T. Mohsenin, Zhiyi Yu, T. Jacobson, G. Landge, M. Meeuwsen, C. Watnik, P. Mejia, Anh Tran, J. Webb, E. Work, Zhibin Xiao, and B. Baas. A 167-processor 65 nm computational platform with per-processor dynamic supply voltage and dynamic clock frequency scaling. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 22–23, 2008.
- [18] Byeong-Gyu Nam, Jeabin Lee, Kwanho Kim, Seung-Jin Lee, and Hoi-Jun Yoo. A 52.4mw 3d graphics processor with 141mvertices/s vertex shader and 3 power do-mains of dynamic voltage and frequency scaling. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 278–603, 2007.

- [19] Chen Zheng and Dongsheng Ma. A 10mhz 92.1switching converter with adaptively compensated single-bound hysteresis control. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 204–205, 2010.
- [20] M. Putic, Liang Di, B.H. Calhoun, and J. Lach. Panoptic dvs: A fine-grained dynamic voltage scaling framework for energy scalable cmos design. In *Computer Design, 2009. ICCD 2009. IEEE International Conference on*, pages 491–497, 2009.
- [21] G. Z. Yang. *Body Sensor Networks*. Springer-Verlag, 2006.
- [22] Wan-Young Chung, Chiew-Lian Yau, Kwang-Sig Shin, and R. Myllyla. A cell phone based health monitoring system with self analysis processor using wireless sensor network technology. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 3705–3708, 2007.
- [23] Shinan Wang, Weisong Shi, B.B. Arnetz, and C. Wiholm. Spartan: A framework for smart phone assisted real-time health care network design. In *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2010 6th International Conference on*, pages 1–10, 2010.
- [24] Chulsung Park and P.H. Chou. Eco: ultra-wearable and expandable wireless sensor platform. In *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on*, pages 4 pp.–165, 2006.
- [25] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan. Adaptive duty cycling for energy harvesting systems. In *Low Power Electronics and Design, 2006. ISLPED'06. Proceedings of the 2006 International Symposium on*, pages 180–185, 2006.
- [26] A. Chhikara, A.H. McGregor, L. Hadjilucas, F. Bello, and A.S. Rice. Quantitative assessment of the motion of the lumbar spine and pelvis with wearable inertial sensors. In *Body Sensor Networks (BSN), 2010 International Conference on*, pages 9–15, 2010.
- [27] Kiing-Ing Wong. Rapid prototyping of a low-power, wireless, reflectance photoplethysmography system. In *Body Sensor Networks (BSN), 2010 International Conference on*, pages 47–51, 2010.
- [28] M. Kony, M. Walter, T. Schlebusch, and S. Leonhardt. An rfid communication system for medical applications. In *Body Sensor Networks (BSN), 2010 International Conference on*, pages 71–75, 2010.
- [29] J. Hart, S. Butler, Hoyeol Cho, Yuefei Ge, G. Gruber, Dawei Huang, Changku Hwang, D. Jian, T. Johnson, G. Konstadinidis, L. Kwong, R. Masleid, U. Nawathe, A. Ramachandran, Yongning Sheng, J.L. Shin, S. Turullois, Zuxu Qin, and King Yen. 3.6ghz 16-core sparcsoc processor in 28nm. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, pages 48–49, 2013.

- [30] Satish Damaraju, Varghese George, Sanjeev Jahagirdar, Tanveer Khondker, R. Milstrey, Sanjib Sarkar, Scott Siers, I. Stoloro, and Arun Subbiah. A 22nm ia multi-cpu and gpu system-on-chip. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 56–57, 2012.
- [31] S. Jain, S. Khare, S. Yada, V. Ambili, P. Salihundam, S. Ramani, S. Muthukumar, M. Srinivasan, A. Kumar, S.K. Gb, R. Ramanarayanan, V. Erraguntla, J. Howard, S. Vangal, S. Dighe, G. Ruhl, P. Aseron, H. Wilson, N. Borkar, V. De, and S. Borkar. A 280mv-to-1.2v wide-operating-range ia-32 processor in 32nm cmos. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 66–68, 2012.
- [32] V. Gutnik and A.P. Chandrakasan. Embedded power supply for low-power dsp. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 5(4):425–435, 1997.
- [33] Liang Di, M. Putic, J. Lach, and B.H. Calhoun. Power switch characterization for fine-grained dynamic voltage scaling. In *Computer Design, 2008. ICCD 2008. IEEE International Conference on*, pages 605–611, 2008.
- [34] Alice Wang, Anantha Chandrakasan, and Stephen V. Kosonocky. Optimal supply and threshold scaling for subthreshold cmos circuits. In *ISVLSI*, pages 7–14. IEEE Computer Society, 2002.
- [35] S. C. Jocke, J.F. Bolus, S.N. Wooters, A.D. Jurik, A.C. Weaver, T.N. Blalock, and B.H. Calhoun. A 2.6- uw sub-threshold mixed-signal ecg soc. In *VLSI Circuits, 2009 Symposium on*, pages 60–61, 2009.
- [36] Freescale. Data sheet: Mma7331lc, 2010.
- [37] Harry C. Powell, Adam T. Barth, and John Lach. Dynamic voltage-frequency scaling in body area sensor networks using cots components. In *Proceedings of the Fourth International Conference on Body Area Networks, BodyNets '09*, pages 15:1–15:8, ICST, Brussels, Belgium, Belgium, 2009. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [38] Bo Zhai, L. Nazhandali, J. Olson, A. Reeves, M. Minuth, R. Helfand, S. Pant, D Blaauw, and T. Austin. A 2.60pj/inst subthreshold sensor processor for optimal energy efficiency. In *VLSI Circuits, 2006. Digest of Technical Papers. 2006 Symposium on*, pages 154–155, 2006.
- [39] B.H. Calhoun, S. Khanna, Yanqing Zhang, J. Ryan, and B. Otis. System design principles combining sub-threshold circuit and architectures with energy scavenging mechanisms. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 269–272, 2010.
- [40] J. Penders, V. Pop, L. Caballero, J. Van de Molengraft, R. Van Schaijk, R. Vullers, and C. Van Hoof. Power optimization in body sensor networks: The case of an autonomous wireless emg sensor powered by pv-cells. In *Engineering in Medicine and*

- Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 2017–2020, 2010.
- [41] N. Verma, A. Shoeb, J. Bohorquez, J. Dawson, J. Gutttag, and A.P. Chandrakasan. A micro-power eeg acquisition soc with integrated feature extraction processor for a chronic seizure detection system. *Solid-State Circuits, IEEE Journal of*, 45(4):804–816, 2010.
- [42] G. Chen, H. Ghaed, R. Haque, M. Wieckowski, Yejoong Kim, Gyouho Kim, D. Fick, Daeyeon Kim, Mingoo Seok, K. Wise, D Blaauw, and D Sylvester. A cubic-millimeter energy-autonomous wireless intraocular pressure monitor. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, pages 310–312, 2011.
- [43] Yi-Chun Shih, Tueng Shen, and B. Otis. A 2.3uw wireless intraocular pressure/temperature monitor. In *Solid State Circuits Conference (A-SSCC), 2010 IEEE Asian*, pages 1–4, 2010.
- [44] V. Leonov, T. Torfs, P. Fiorini, and C. Van Hoof. Thermoelectric converters of human warmth for self-powered wireless sensor nodes. *Sensors Journal, IEEE*, 7(5):650–657, 2007.
- [45] E.J. Carlson, K. Strunz, and B.P. Otis. A 20 mv input boost converter with efficient digital control for thermoelectric energy harvesting. *Solid-State Circuits, IEEE Journal of*, 45(4):741–750, 2010.
- [46] Y.K. Ramadass and A.P. Chandrakasan. A battery-less thermoelectric energy harvesting interface circuit with 35 mv startup voltage. *Solid-State Circuits, IEEE Journal of*, 46(1):333–341, 2011.
- [47] Y.K. Ramadass and A.P. Chandrakasan. Voltage scalable switched capacitor dc-dc converter for ultra-low-power on-chip applications. In *Power Electronics Specialists Conference, 2007. PESC 2007. IEEE*, pages 2353–2359, 2007.
- [48] Fan Zhang, A. Mishra, A.G. Richardson, and B. Otis. A low-power ecog/eeg processing ic with integrated multiband energy extractor. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 58(9):2069–2082, 2011.
- [49] Microchip. Microchip.
- [50] N. Verma and A.P. Chandrakasan. A 256 kb 65 nm 8t subthreshold sram employing sense-amplifier redundancy. *Solid-State Circuits, IEEE Journal of*, 43(1):141–149, 2008.
- [51] Fan Zhang, A. Mishra, A.G. Richardson, S. Zanos, and B.P. Otis. A low-power multi-band ecog/eeg interface ic. In *Custom Integrated Circuits Conference (CICC), 2010 IEEE*, pages 1–4, 2010.
- [52] Jiapu Pan and Willis J. Tompkins. A real-time qrs detection algorithm. *Biomedical Engineering, IEEE Transactions on*, BME-32(3):230–236, 1985.

- [53] Douglas E Lake and J Randall Moorman.
- [54] J. Pandey and B.P. Otis. A sub-100 uw micsism band transmitter based on injection-locking and frequency multiplication. *Solid-State Circuits, IEEE Journal of*, 46(5):1049–1058, 2011.
- [55] A. Shrivastava and B.H. Calhoun. A 150nw, 5ppm/o c, 100khz on-chip clock source for ultra low power socs. In *Custom Integrated Circuits Conference (CICC), 2012 IEEE*, pages 1–4, 2012.
- [56] Dongmin Yoon, D Sylvester, and D Blaauw. A 5.58nw 32.768khz dll-assisted xo for real-time clocks in wireless sensing applications. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 366–368, 2012.
- [57] K. Kadirvel, Y. Ramadass, U. Lyles, J. Carpenter, V. Ivanov, V. McNeil, A. Chandrakasan, and B. Lum-Shue-Chan. A 330na energy-harvesting charger with battery management for solar and thermoelectric energy harvesting. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, pages 106–108, 2012.
- [58] OpenCores. openmsp430, 2012.
- [59] Texas Instrument. Msp430x1xx family user’s guide (rev. f), 2006.
- [60] K. Lahiri, A. Raghunathan, S. Dey, and D. Panigrahi. Battery-driven system design: a new frontier in low power design. In *Design Automation Conference, 2002. Proceedings of ASP-DAC 2002. 7th Asia and South Pacific and the 15th International Conference on VLSI Design. Proceedings.*, pages 261–267, 2002.
- [61] K. Lahiri, A. Raghunathan, and S. Dey. Efficient power profiling for battery-driven embedded system design. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 23(6):919–932, 2004.
- [62] M. Kulkarni and V.D. Agrawal. Energy source lifetime optimization for a digital system through power management. In *System Theory (SSST), 2011 IEEE 43rd Southeastern Symposium on*, pages 73–78, 2011.
- [63] C. Moser, L. Thiele, D. Brunelli, and L. Benini. Adaptive power management in energy harvesting systems. In *Design, Automation Test in Europe Conference Exhibition, 2007. DATE '07*, pages 1–6, 2007.
- [64] Qiang Liu, T. Mak, Junwen Luo, W. Luk, and A. Yakovlev. Power adaptive computing system design in energy harvesting environment. In *Embedded Computer Systems (SAMOS), 2011 International Conference on*, pages 33–40, 2011.
- [65] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M.B. Srivastava. Design considerations for solar energy harvesting wireless embedded systems. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 457–462, 2005.

- [66] D. Pimentel and P. Musilek. Power management with energy harvesting devices. In *Electrical and Computer Engineering (CCECE), 2010 23rd Canadian Conference on*, pages 1–4, 2010.
- [67] Y.K. Ramadass and A.P. Chandrakasan. Minimum energy tracking loop with embedded dc/dc converter enabling ultra-low-voltage operation down to 250 mv in 65 nm cmos. *Solid-State Circuits, IEEE Journal of*, 43(1):256–265, 2008.
- [68] P. Macken, M. Degrauwe, M. Van Paemel, and H. Oguey. A voltage reduction technique for digital systems. In *Solid-State Circuits Conference, 1990. Digest of Technical Papers. 37th ISSCC., 1990 IEEE International*, pages 238–239, 1990.
- [69] B.H. Calhoun and A.P. Chandrakasan. Ultra-dynamic voltage scaling (udvs) using sub-threshold operation and local voltage dithering. *Solid-State Circuits, IEEE Journal of*, 41(1):238–245, 2006.
- [70] S.N. Wooters, B.H. Calhoun, and T.N. Blalock. An energy-efficient subthreshold level converter in 130-nm cmos. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, 57(4):290–294, 2010.
- [71] Apple. Apple iphone5 specifications, 2013.
- [72] HTC. Htc-one specifications, 2013.
- [73] mophie. mophie, 2013.
- [74] Sheng Hu, Zhenzhou Shao, and Jindong Tan. A real-time cardiac arrhythmia classification system with wearable electrocardiogram. In *Body Sensor Networks (BSN), 2011 International Conference on*, pages 119–124, 2011.
- [75] Lin Zhong, M. Sinclair, and R. Bittner. A phone-centered body sensor network platform cost, energy efficiency and user interface. In *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on*, pages 4 pp.–182, 2006.
- [76] Apple. Nike + ipod.
- [77] E. Farella, A. Pieracci, L. Benini, and A. Acquaviva. A wireless body area sensor network for posture detection. In *Computers and Communications, 2006. ISCC '06. Proceedings. 11th IEEE Symposium on*, pages 454–459, 2006.
- [78] T. Schlebusch, L. Rothlingshofer, Saim Kim, M. Kony, and S. Leonhardt. On the road to a textile integrated bioimpedance early warning system for lung edema. In *Body Sensor Networks (BSN), 2010 International Conference on*, pages 302–307, 2010.
- [79] A. Wang, A.P. Chandrakasan, and S.V. Kosonocky. Optimal supply and threshold scaling for subthreshold cmos circuits. In *VLSI, 2002. Proceedings. IEEE Computer Society Annual Symposium on*, pages 5–9, 2002.

- [80] J. Yoo, Long Yan, Seulki Lee, Yongsang Kim, and Hoi-Jun Yoo. A 5.2 mw self-configured wearable body sensor network controller and a 12 w wirelessly powered sensor for a continuous health monitoring system. *Solid-State Circuits, IEEE Journal of*, 45(1):178–188, 2010.
- [81] Long Yan, Joonsung Bae, Seulki Lee, Taehwan Roh, Kiseok Song, and Hoi-Jun Yoo. A 3.9 mw 25-electrode reconfigured sensor for wearable cardiac monitoring system. *Solid-State Circuits, IEEE Journal of*, 46(1):353–364, 2011.
- [82] Bo Zhai, D Blaauw, D Sylvester, and K. Flautner. Theoretical and practical limits of dynamic voltage scaling. In *Design Automation Conference, 2004. Proceedings. 41st*, pages 868–873, 2004.
- [83] H. Ghasemzadeh and R. Jafari. Data aggregation in body sensor networks: A power optimization technique for collaborative signal processing. In *Sensor Mesh and Ad Hoc Communications and Networks (SECON), 2010 7th Annual IEEE Communications Society Conference on*, pages 1–9, 2010.
- [84] Wonyoung Kim, D.M. Brooks, and Gu-Yeon Wei. A fully-integrated 3-level dc/dc converter for nanosecond-scale dvs with fast shunt regulation. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, pages 268–270, 2011.
- [85] Hui Zhang, V. Prabhu, Varghese George, M. Wan, M. Benes, A. Abnous, and J.M. Rabaey. A 1-v heterogeneous reconfigurable dsp ic for wireless baseband digital signal processing. *Solid-State Circuits, IEEE Journal of*, 35(11):1697–1704, 2000.
- [86] San-Jeow Cheng, Yuan Gao, Wei-Da Toh, Yuanjin Zheng, Minkyu Je, and Chun-Huat Heng. A 110pj/b multichannel fsk/gmsk/qpsk/p/4-dqpsk transmitter with phase-interpolated dual-injection dll-based synthesizer employing hybrid fir. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, pages 450–451, 2013.