Variation Aware Energy-Efficient Methodologies for Homogeneous Many-Core Designs

Meeta Srivastav

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computer Engineering

Leyla Nazhandali, Chair
Patrick Schaumont
Mary Jane Irwin
Paul Plassmann
Ali Butt

December 3, 2014
Blacksburg, Virginia

Keywords: Low Power, Near-threshold, Energy Efficient di/dt, Process, homogeneous many-core, Variation

Copyright 2014, Meeta Srivastav
Variation Aware Energy-Efficient Methodologies for Homogeneous Many-Core Designs

Meeta Srivastav

ABSTRACT

Earlier designs were driven by the goal of achieving higher performance, but lately, energy efficiency has emerged as an even more important design principle. Strong demand from the consumer electronics drives research in the low power and energy-efficient methodologies. Moreover, with exponential increase in the number of transistors on a chip and with further technology scaling, variability in the design is now of greater concern. Variations can make the design unreliable or the design may suffer from sub-optimal performance. Through the work in this thesis, we present a multi-dimensional investigation into the design of variation aware energy-efficient systems. Our overarching methodology is to use system-level decisions to mitigate undesired effects originating from device-level and circuit-level issues.

We first look into the impact of process variation (PV) on energy efficient, scalable throughput many-core DSP systems. In our proposed methodology, we leverage the benefits of aggressive voltage scaling (VS) for obtaining energy efficiency while compensating for the loss in performance by exploiting parallelism present in various DSP designs. We demonstrate this proposed methodology consumes 8% - 77% less power as compared to simple dynamic VS over different workload environments. Later, we show judicious system-level decisions, namely, number of cores, and their operating voltage can greatly mitigate the effects of PV and consequently, improve the energy efficiency of the design. We also present our analysis discussing the impact of aging on the proposed methodology.

To validate our proposed system-level approach, design details of a prototype chip fabricated in the 90nm technology node and its findings are also presented. The chip consists of 8 homogeneous FIR cores, which are capable of running from near-threshold to nominal voltages. In the 20-chip population, we observe 7% variation in the speed at nominal voltage (0.9V) and 26% at near threshold voltage (0.55V) among all the cores. We also observe 54% variation in power consumption characteristics of the cores. The chip measurement results show that our proposed methodology of judiciously selecting the cores and their operating voltage can result in 6.27% - 28.15% more energy savings for various workload environments, as compared to globally voltage scaled systems. Furthermore, we present the impact of temperature variations on the energy-efficiency of the above systems.

We also study the problem of voltage variations in the integrated circuits. We first present the characteristics of a dynamic voltage noise as measured on a 28nm FPGA. We propose a fully digital on-chip sensor that can detect the fast voltage transients and alert the system of voltage emergency. A traditional approach to mitigate this problem is to use safety guardbands. We demonstrate that our proposed sensor system will be 6% - 27.5% more power efficient than the traditional approach.
Dedication

This dissertation is dedicated to my husband, Anil Nadig, who has been a constant source of support and encouragement during the challenges of graduate school and life. I am truly thankful to him for never leaving my side.

To my loving parents, Surendramohan Srivastava and Mridula Srivastava, for their endless love, support and encouragement.

To my caring parents-in-law, Late Srinivasmurthy Nadig and Padma Nadig, who have always given me complete support throughout the process of my doctorate studies.
Acknowledgments

Foremost, I would like to express my sincere gratitude to Dr. Leyla Nazhandali, my advisor for her countless number of hours she spent in guiding, and mentoring. I really appreciate the patience, and the confidence she has shown in me during the entire period of my doctoral program.

Besides my advisor, I would like to thank the rest of my committee: Prof. Patrick Schaumont, Prof. Mary Jane Irwin, Prof. Paul Plassmann, and Prof. Ali Butt, for their encouragement, insightful comments, and the expertise they shared during our discussions.

I would like to thank my fellow labmates in Security and Power Aware Computing lab: Dinesh, Lalleh, Sinan, Mike, Yongbo, Rashmi, and Mehrdad for creating a stimulating environment, and sometimes spending sleepless nights during the deadline period. Their enthusiasm made working in the lab as well as completion of this research an enjoyable experience.

I would like to thank all my roommates and friends including Atashi, Tina, Mittu, Priyanka, Mohini, Suvarna for the company they gave me outside my research. I would like to thank my brother Nitin and his family as well as my sister Neeta and her family for their support and encouragement.

I would also like to thank Manisha, Ritesh and people in the writing center for spending long hours in thoroughly reviewing my thesis.
## Contents

1 Introduction .................................................. 1
  1.1 Organization and Key Contributions ............................ 6

2 Process Variation Aware Energy-Efficient Designs .................. 9
  2.1 Motivation and Introduction .................................. 10
    2.1.1 Roadmap for the Chapter ................................ 12
  2.2 Background ................................................. 13
  2.3 Related Literature ........................................... 15
  2.4 Design Architecture and Environment ............................ 17
    2.4.1 Static Environment ...................................... 19
    2.4.2 Dynamic Environment .................................... 22
  2.5 Design Parameter Selection ..................................... 23
    2.5.1 Choosing $N_{opt}$ and $V_{dd opt}$ in the Static Environment .... 24
    2.5.2 Choosing $N_{opt}$ and $V_{dd opt}$ in the Dynamic Environment .... 25
  2.6 Experimental Setup ............................................ 28
  2.7 Results ....................................................... 30
    2.7.1 Energy Efficiency of Hybrid Technique .................... 31
    2.7.2 Energy Savings in a Static Environment ................... 31
    2.7.3 Energy Savings in a Dynamic Environment ................. 32
    2.7.4 Technology Trends ....................................... 34
    2.7.5 Battery Lifetime Modeling and Estimation ................ 34


## 2.8 Design Corner Aware (Binning) Methodology
- 2.8.1 Design and Implementation of the Controller
- 2.8.2 Results of Binning

## 2.9 Analyzing the Impact of Aging
- 2.9.1 Background and Related Literature
- 2.9.2 System Overview
- 2.9.3 Experimental Setup
- 2.9.4 Results

## 2.10 Conclusions

## 3 Design Details and Test Results for an ASIC Prototype

### 3.1 Motivation

### 3.2 Related Literature

### 3.3 Design Architecture and Methodology
- 3.3.1 Design Parameter Selection

### 3.4 Prototype Implementation Details
- 3.4.1 Clock Management
- 3.4.2 Controller Design
- 3.4.3 Level Converters (LC)
- 3.4.4 Homogeneous Cores

### 3.5 Testing Setup
- 3.5.1 Functional Testing and Speed Measurements
- 3.5.2 Power Measurements
- 3.5.3 Temperature Measurements
- 3.5.4 Characterization Overhead and Scalability

### 3.6 Results
- 3.6.1 Range of Frequencies Available on Chip
- 3.6.2 Speed and Power Measurements
- 3.6.3 Impact of PV on choosing $[N_{opt} V_{dd_{opt}}]$
## List of Figures

1.1 Energy and performance trade-off for circuits operating at near-threshold voltage range [24]. ................................................................. 2

1.2 Impact of variation on homogeneous many-core design systems. ............. 3

1.3 Different variations according to their time constants [91]. ...................... 3

1.4 Variation mitigation strategies for each level of the design hierarchy. ........ 4

2.1 Impact of static variation at different abstraction level of the design. ........ 13

2.2 Delay COV ($\sigma/\mu$) of FFT processor at 90nm and 45nm technology node. 16

2.3 Architecture of a fixed throughput system. ........................................ 19

2.4 Tradeoff between power-gating and clock-gating schemes for an FIR filter. 20

2.5 Architecture of a dynamically scalable throughput systems. .................. 22

2.6 (A) Choosing the set $[N V_{DD}]$ for a targeted throughput of 330 MHz in the Static Environment. (B) Min Energy point for a PVI and PVA design. .... 24

2.7 An algorithm to find the most energy efficient system design parameters, under a dynamic environment. .............................................. 26

2.8 (A) Selecting the number of cores and operating voltage based on throughput for FFT processor$^a$. (B) Selecting the most energy optimum set of $[N V_{dd}]$ for FFT processor. $^a$For better visibility, the points in the graph which cannot deliver 330 MHz are trimmed to 10MHz frequency ............................... 26

2.9 Comparison of power efficiency obtained by hybrid technique over VS and ACS (Clock Scaling). ......................................................... 30

2.10 Power efficiency of a PVA and PVI design for an FFT processor under a static environment ................................................................. 31

2.11 Power efficiency of a PVA and PVI design for an FFT processor under a dynamic environment. ......................................................... 33
2.12 (A) Battery lifetime achievable by a PVA and PVI design for an FFT DSP system under a dynamic environment. (B) Battery lifetime achievable by a PVA and PVI design for an FFT system under a Static Environment. 35

2.13 Normal distribution of the cores based on their speed of operation. 36

2.14 State machine diagram of the controller. 38

2.15 An algorithm to select the design parameters tuples \([N_{opt}, Vdd_{opt}]\) based on binning methodology. 39

2.16 Power efficiency of binning and slowest methodology for various DSP application systems. 41

2.17 Battery lifetime achievable by binning and slowest methodology on various DSP systems under a dynamic environment. 42

2.18 Impact of aging and PV with voltage scaling on the speed of iDCT design. 43

2.19 Impact of aging on the fastest and slowest core over years. 47

2.20 Deterioration of fast, typical and slow cores in the system. 48

2.21 Distribution of all the cores in the system based on speed using policy A and policy B under static workload environment. 49

2.22 Distribution of all the cores in the system based on speed using policy A and policy B under dynamic workload environment. 51

3.1 Architecture of a dynamically scalable throughput homogeneous many-core systems. 60

3.2 Frequency voltage characteristics of FIR filter with various temperatures based on simulation model. 62

3.3 Top level chip layout with all 16 cores. 64

3.4 (A) Test setup. (B) Temperature variation chamber setup. 66

3.5 (a) Range of on-chip clock frequencies on Chip 0. (b) Histogram showing frequency measured from on-chip SC2 clock for 20 chips. 69

3.6 (a) Histogram showing frequency measured for all cores in 20 chips (160 cores). (b) Histogram showing power measured for all cores in 20 chips (160 cores). 70

3.7 \(N_{opt}\) chosen by different chips [0-19] over a range of desired throughputs [100 MHz - 2.5 GHz]. 72

3.8 \(V_{dd_{opt}}\) chosen by different chips [0-19] over a range of desired throughputs [100 MHz - 2.5 GHz]. 72
3.9 Core numbers $core_{opt}$ chosen among 8 cores by different chips over a wide range of throughputs [100 MHz - 2.5 GHz].

3.10 Energy savings as compared to BASE for low workload (10 MHz - 300 MHz), medium workload (400 MHz - 800 MHz) and high workload (900 MHz - 2.5 GHz) environments.

3.11 Distribution of cores based on total power consumed for Chip 3 measured at 50 MHz frequency and an operating voltage of 0.45 V.

3.12 Total average power consumed at Cold (T0) and Hot (T1) temperature for all cores in (A) Chip 3 and (B) Chip 5, measured at 50MHz frequency and operating voltage of 0.65 V.

3.13 Difference in total energy consumed at T0 and T1 temperature w.r.t room temperature, for Chip 3 and Chip 5 under low workload (10MHz-300MHz), medium workload (400MHz-800MHz) and high workload (900MHz-2.5GHz) environment.

4.1 (A) Voltage noise source in a many-core environment sharing the same PDN. (B) Voltage noise characteristics as measured on 28nm technology node.

4.2 High level diagram of the voltage sensor system.

4.3 (a) Steps to generate voltage transients. (b) Measurement of voltage transient measured using an oscilloscope.

4.4 Architecture of sensing component.

4.5 Rise (left) and Fall (right) delay for different cells in the delay line component for 32nm library.

4.6 Delay sensitivity of SBUF based delay element for different technology nodes.

4.7 Architecture of the digitizing component (TDC).

4.8 Timing diagram of all the input/output signals of the sensor.

4.9 Tradeoff between delay line stages, sensor resolution and length of TDC component.

4.10 Relation between PDN voltage noise and TDC measurements.

4.11 (a) Setup for measuring intra-die voltage variations. (b) Sensor readings from UART. (c) Placement of the two sensors and switching fabric in the FPGA.

4.12 Sensor readings of TDC0 and TDC1.

4.13 Concept of reducing safety timing margins.
4.14 Area and power savings for FFT design. ........................................ 105
4.15 Relation between critical path delay of the designs and magnitude of voltage undershoot. ..................................................... 106
## List of Tables

2.1 Wavelength of light used for the lithography process. ........................................ 15
2.2 \(V_{th}\) and \(L_{gate}\) Variation. [40] ................................................................. 18
2.3 Optimal tuples of \([N_{opt} \, V_{dd\_opt}]\) at different throughputs under static environment, for an FFT system. ................................................................. 24
2.4 Optimum set \([N_{opt} \, V_{dd\_opt}]\) at different throughputs under dynamic environment, for FFT processor. ................................................................. 27
2.5 Four optimum voltages for PVI and PVA design under dynamic environment, for FFT many-core system. ................................................................. 27
2.6 List of library and tools used to conduct experiments. ................................................ 28
2.7 Fast Simulation Framework. .................................................................................. 29
2.8 Fast Simulation Framework. .................................................................................. 29
2.9 Percentage energy savings of PVA design over PVI with PG, under both static and dynamic environment. ................................................................. 32
2.10 Percentage improvement of PVA design over PVI in 90\(nm\) and 45\(nm\) technology. 34
2.11 \textit{Binning} of cores based on their critical path delay. .......................................... 37
2.12 Summary of design implementation and testing overheads of the calibration module. \(^a\)Clock cycles were calculated for DCT application based on 8 x 8 matrix data input and a resolution of 12-bit coefficients for 10 test patterns.
\(^b\) Gate Equivalent (GE) is calculated by dividing the post-layout area by the area of a 2 input NAND cell. ................................................................................................. 37
2.13 \([N_{opt} \, V_{dd\_opt}]\) tuples for different design corners at 500 MHz desired throughput, DCT core. ................................................................. 40
2.14 Energy efficiency of the system with \textit{binning} as compared to the system operating at the \textit{slowest} core speed. ................................................................. 42
2.15 Energy efficient configurations for iDCT design. .................................................... 46
3.1 Set of \([N_{opt} Vdd_{opt}]\) tuples chosen by different chips [0-19] over a range of desired throughputs [100MHz - 2.5GHz]. ........................................... 71

4.1 Impact of voltage noise on timing paths with 15% voltage noise in the supply. 85

4.2 Simulation setup ................................................................. 97

4.3 Implementation overhead of an on-chip sensor for measuring different magnitude of voltage noise. .................................................... 99

4.4 Area and power efficiency of using a VVA design over BASE design. ........ 105
List of Abbreviations

ACS  Active Core Scaling
ASIC  Application Specific Integrated Circuits
BASE  Reference Design
CG  Clock Gating
COV  Coefficient of Variation
CPM  Critical Path Monitor
DCA  Design Corner Aware
DCT  Discrete Cosine Transform
DSP  Digital Signal Processing
DUT  Design Under Test
DVFS  Dynamic Voltage Frequency Scaling
EMS  Emergency Monitoring System
FFT  Fast Fourier Transform
FIR  Finite Impulse Response
FPGA  Field Programmable Gate Array
HCI  Hot Carrier Injection
IC  Integrated Circuit
iDCT  Inverse Discrete Cosine Transform
LC  Level Converters
NA  Numerical Aperture
NBTI  Negative Bias Temperature Instability
NTR  Near Threshold Region
NTV  Near Threshold Voltage
PCB  Printed Circuit Board
PCI  Peripheral Component Interconnect
PDN  Power Distribution Network
PE  Processing Elements
PG  Power Gating
PTM  Predictive Technology Model
PV  Process Variation
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVA</td>
<td>Process Variation Aware</td>
</tr>
<tr>
<td>PVI</td>
<td>Process Variation Ignorant</td>
</tr>
<tr>
<td>PVT</td>
<td>Process Voltage Temperature</td>
</tr>
<tr>
<td>QFP</td>
<td>Quad Flat Package</td>
</tr>
<tr>
<td>RO</td>
<td>Ring Oscillator</td>
</tr>
<tr>
<td>RDF</td>
<td>Random Dopant Fluctuation</td>
</tr>
<tr>
<td>RLE</td>
<td>Run Length Encoder</td>
</tr>
<tr>
<td>SBUF</td>
<td>Standard Buffer</td>
</tr>
<tr>
<td>S-RO</td>
<td>Standard-cell Ring Oscillator</td>
</tr>
<tr>
<td>SM</td>
<td>State Machine</td>
</tr>
<tr>
<td>TDC</td>
<td>Time Division Converter</td>
</tr>
<tr>
<td>VVA</td>
<td>Voltage Variation Aware</td>
</tr>
<tr>
<td>VDDH</td>
<td>High Voltage Domain</td>
</tr>
<tr>
<td>VDDL</td>
<td>Low Voltage Domain</td>
</tr>
<tr>
<td>VHDCI</td>
<td>Very High Density Cable Interface</td>
</tr>
<tr>
<td>VI</td>
<td>Voltage Increase</td>
</tr>
<tr>
<td>VS</td>
<td>Voltage Scaling</td>
</tr>
<tr>
<td>$Bin_{opt}$</td>
<td>Optimal Bin Number</td>
</tr>
<tr>
<td>$Core_{opt}$</td>
<td>Optimal Core Number</td>
</tr>
<tr>
<td>$L_{eff}$</td>
<td>Effective channel length</td>
</tr>
<tr>
<td>$N_{opt}$</td>
<td>Optimal Number of Cores</td>
</tr>
<tr>
<td>$Vdd_{opt}$</td>
<td>Optimal Operating Voltage</td>
</tr>
<tr>
<td>$V_{th}$</td>
<td>Threshold voltage of a transistor</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Semiconductor markets continue to grow with an ever increasing demand to pack more and more functionality and features within a single chip. Further, through Moore’s law\(^1\) and Dennard scaling\(^2\), the semiconductor industry has been able to sustain this rapid pace of improvements in ICs by decreasing the feature sizes. However, starting around 65\(nm\) and below, Dennard scaling has broken down. The primary reason for the breakdown is an increase in the leakage current and hence, power. Increase in power density can cause the chip to heat up, which creates a threat of thermal runaway and will further increase packaging and heat removal costs. Therefore, in order to sustain the growth of the semiconductor industry, reducing the power consumption has become a vital concern.

Energy efficient VLSI circuits are in growing demand in various applications, such as smart cards, wireless sensor networks, and medical implants, where power is supplied by small batteries or limited scavenged energy. In such applications, energy efficiency concerns supersede the traditional emphasis on speed. The main concerns are battery lifetime maximization, robustness, fault tolerance and self-configuration. One very attractive approach for energy-constrained applications to gain energy efficiency is to use aggressive Voltage Scaling (VS).

\(^{1}\)Number of transistors on a chip doubles every 18 months.

\(^{2}\)As the transistor get smaller their power density stays constant.
In a VS technique, the power supply voltage is decreased beyond the nominal supply voltage. As shown in Fig. 1.1, in the super-threshold region, energy is extremely sensitive to supply voltage due to the quadratic dependence of dynamic (active) energy on the supply voltage. Therefore, voltage scaling down to the near-threshold voltages yield 10x energy reduction at the expense of nearly a 10x performance decrease [24]. Interestingly, energy reduces by only 2x when VDD is further scaled down from the near-threshold region to the sub-threshold region, but at the same time delay increases dramatically by 50x-100x.

Previous works have shown that the loss in performance due to aggressive voltage scaling can be regained to some extent by using multiple identical cores (homogeneous) to process the data in parallel [94]. The above approach will not only yield similar energy gains, but will also help in meeting the performance (throughput) requirement. This is especially true for algorithms that can be highly parallelized such as the ones implemented in most DSP applications. One of the main challenges in designing such a homogeneous many-core DSP systems for energy-constrained applications, is in determining the optimal system parameters necessary to meet the throughput requirement along with achieving maximum energy-efficiency. In other words, the challenge is to determine the optimal number of cores, and their optimal operating voltage that can meet the throughput requirement with minimal energy consumption. Through the work in this thesis, we propose a methodology for choosing energy-efficient system parameters under different workload environments. We will then
One of the key design challenges related to homogeneous many-core systems operating at near-threshold voltages that should be addressed is the increase in variability. Variability is defined as any deviation from the specification of a parameter that can cause nondeterministic results. Due to extreme miniaturization in nano-electronics, even small deviations may lead to unusable or suboptimal circuits. Variation can impact both speed and power characteristics of the core. In the presence of variability, cores of a homogeneous system are no longer similar to each other, but instead will differ in terms of their operating speed and power consumption as shown in Fig. 1.2.

Existence of variability in the design is not new, but many physical phenomena that were once minor effects are now creating a major impact on the design characteristics. For example, the same 1\textit{nm} variation in gate length is relatively larger for minimally sized gates in a 22\textit{nm} process than in a 45\textit{nm} one. Additionally, ever increasing complexity of ICs, complicated power management features, and uncertain workload environments increase this deviation.
Figure 1.4: Variation mitigation strategies for each level of the design hierarchy.

from the specification. Variation can be differentiated mainly into four different types, based on the time it is introduced in an IC. These categories are Process (P), Voltage supply noise (V), Temperature (T) and Aging (A) as shown in the Fig. 1.3. They are generally abbreviated as PVT/A. Fig. 1.3 categorizes these variations according to their time constants [91].

Process Variation (PV) is defined as the variation that creates a fixed impact on the device and remains unchanged. It can be thought of as a defect. Once introduced, it will remain for the lifetime of the device. They are introduced in the devices during the IC fabrication process. Variation in the design that are not permanent in nature or vary with time, are classified as dynamic variation. For example, changes in temperature and voltage supply can cause dynamic (momentary) changes in the design characteristics. Additionally, during the lifetime of a device its characteristics will be affected by aging induced wear-out. In the long run, significant degradation of CMOS logic delays can develop due to aging. To summarize, there are different types of variation, both static and dynamic, that can cause a huge impact on the design in terms of power usage, performance, and even the correctness of results.

Impact of variation can be mitigated at various design hierarchy levels as shown in the Fig. 1.4. At the device level, many process variation prevention methods are currently being explored. Continuous improvements in the fabrication process and in the use of materials have been made, to increase device performance, and reduce variability [28,90]. At circuit-level, some of the mitigating techniques include using adaptive feedback schemes, such as
adaptive body biasing [82]. Detection and correction of timing errors due to variations by using circuit structures like RAZOR within the pipeline stages are also being explored [18]. At system or architectural level, a safety margin is often added globally to a design to ensure correct and predictable behavior in the presence of variability. Mitigating techniques at various hierarchy levels can work in conjunction with each other. In this thesis, we present a system-level technique that can help in mitigating undesired effects in a design due to the impact of variation.

Further, the effects of variability in a design are more dominant in ultra-low voltage regions [86]. Therefore, it becomes essential to study and analyze the impact of variability on system parameters, while determining the energy-efficiency of a homogeneous many-core system. Besides, as shown in Fig. 1.2, in the presence of variation, a homogeneous system now behaves like a heterogeneous system. A simple solution to bring homogeneity back in the design is to run all the cores at the speed of the slowest core in the system. This will be the case in a typical global voltage scaled system. However, our proposed approach of gathering the characteristic behavior of each core on the die after fabrication (performed only once) will help increase the energy-efficiency of the system. This is mainly true when the number of optimal cores required is less than the total cores present in the system. Our proposed methodology can analyze the impact of PV on the cores and then judiciously select the energy-efficient cores. In this case, the system can operate at the speed defined by the current set of cores selected rather than the slowest in the system. In this thesis, we will show how the above approach of selectively choosing cores can increase both the energy-efficiency and performance of a system.

As part of these investigations, a detailed study of a Process Variation Aware (PVA) system is performed and compared with the Process Variation Ignorant (PVI) counterpart of the system. Energy-efficiency of both the systems is compared under variable workload conditions. Further, based on chip measurement results, we demonstrate the impact of intra-die and inter-die process variation on a homogeneous 8-cores FIR based system, operating at near-threshold voltages. Later, we present an analysis of temperature and aging variations
on the system parameters and recommended policies.

To continue our work on system variation in this thesis, we will also study the characteristics and impact of supply voltage variation in various designs. We propose a fast all-digital sensor capable of sensing supply voltage variation. With the help of the proposed voltage sensor, we will discuss the design of an emergency monitoring system, which can help in monitoring, measuring, and raising an alert when the design experiences voltage emergencies. Thus, this thesis presents a multi-dimensional investigation into the design of variation-aware energy-efficient many-core homogeneous DSP systems operating at near-threshold voltages.

1.1 Organization and Key Contributions

In Chapter 2, we propose an energy efficient design for a many-core DSP system that caters to a wide range of throughputs and is also process variation aware. We will present in detail the impact of process variation on the energy-efficiency of such systems under different workload environments. We will also discuss energy efficiency of PVA systems as compared to PVI systems.

The material in Chapter 2 has been disseminated via two conference papers [71,74] and one journal paper [75]. The key contributions of Chapter 2 are as follows:

1. We propose a hybrid methodology using Active Core Scaling (ACS) and Voltage Scaling (VS) for building energy-efficient systems.

2. Based on simulation, the proposed hybrid methodology can achieve nearly 8% (min) to 77% (max) of energy savings as compared to VS or ACS alone for different DSP benchmark circuits.

3. We present a methodology to selectively choose cores in the system based on delay variation profile, which can further improve the efficiency by 49.3%.
4. We present the impact of aging on our proposed algorithm for choosing energy-efficient architectures under different workload conditions.

In Chapter 3 we present the architecture as well as test results of a prototype chip, which is designed in 90nm technology node. The prototype is used as a platform to validate the findings presented in Chapter 2. The chip consists of 8 homogeneous FIR cores, which are capable of running from near-threshold to nominal voltages. Based on chip measurement results performed on 20 chips, we demonstrate the impact of intra-die and inter-die variations on optimal system parameters.

The findings of Chapter 3 has been disseminated through one journal [78] and one conference paper [73]. The key contributions of Chapter 3 are as follows:

1. We provide the frequency and power measurements of an 8-core FIR filter, which we use as a vehicle to show the effectiveness of our proposed hybrid method in saving energy.

2. We present an extended analysis of process variation effect on the number of cores and operating voltages for a population of 20 chips over a wide range of throughputs.

3. From the chip measurement results, we observe 6% to 28% of energy savings as compared to systems with Voltage Scaling (VS) capabilities alone.

4. We present an analysis of the impact of ambient temperature changes on energy efficient architectures.

In Chapter 4, we present the impact of voltage variation, which are dynamic in nature, on the design. We propose a fast all-digital sensor capable of sensing voltage transients on the order of nano-seconds speed. We will discuss the design details of our proposed sensor, its characteristics, and its overheads. With the help of the proposed sensor, we will give insights into the presence of intra-die dynamic voltage supply variations. Further, we will discuss
design details of an emergency monitoring system, which can help in monitoring, measuring, and raising an alert when the design experiences a voltage emergency.

The material in Chapter 4 is presented in one journal [77] and one conference [98] paper. The key contributions of Chapter 4 are as follows:

1. We demonstrate an example of a novel demonstration of extreme voltage undershoot and overshoot caused by fabric activity on a 28nm technology node.

2. We design and implement a process variation aware digital sensor capable of quickly characterizing nano-second scale transients that normally go undetected.

3. We analyze tradeoffs between sensor resolution, its speed, and area overhead.

4. We develop an alert system that is capable of detecting short-term voltage fluctuations. Such a system is employed to reduce or remove timing margins in order to gain area and power savings.

Chapter 5 offers conclusions for this work.
Chapter 2

Process Variation Aware
Energy-Efficient Designs

In this chapter, we present a system-level approach to build an energy-efficient design for a homogeneous many-core DSP system that can cater to a wide range of throughputs and is also Process Variation Aware (PVA). Voltage scaling has been a prevalent method of saving energy for energy constrained applications. However, the current technology trends which shrink transistor sizes exacerbate process variation effects in voltage scaled systems. Large variations in transistor parameters result in high variations in performance and power across the chip. These effects, if ignored at the stage of designing, may result in unpredictable behavior when deployed in the field. In our proposed methodology, we leverage benefits of the voltage scaling methodology for obtaining energy efficiency and compensate for the loss in throughput by exploiting parallelism present in the various DSP designs. We study this system architecture in two different workload environments; static and dynamic. Besides, we will show that to achieve the highest level of energy efficiency, the number of cores and their operating voltages vary widely between a Process Variation Ignorant (PVI) versus a PVA design.
2.1 Motivation and Introduction

As described in Chapter 1, performance is lost significantly in near and sub-threshold region while the power and energy is reduced. Using multiple identical (homogeneous) cores running in parallel at lower voltages can regain some of the lost performance while maintaining the energy efficiency of these systems [94]. This is especially true for algorithms that can be highly parallelized, such as the ones implemented in most DSP applications.

In a given DSP application the target throughput and the energy consumption of the system can be optimized by varying the operating voltage and the number of parallel cores. In a dynamic environment, where the target throughput varies over time, the traditional approach is to dynamically adjust the operating voltage to meet the required throughput. However, for small, low power embedded systems with limited resources, having a wide variety of operating voltages accessible to the DSP cores is unrealistic due to extra overhead. Instead, we propose a hybrid method, where dynamic Voltage Scaling (VS) is combined with what we call as Active Core Scaling (ACS), in which the number of active cores is decreased by means of power gating. This hybrid method uses only a few operating voltages, picks the number of active cores, and selects the operating voltage in order to meet the target throughput with minimal energy. We show that this approach, in comparison with the traditional approaches of VS and ACS, uses less resources and consumes less amount of energy. We apply this hybrid methodology over a wide range of desired throughputs from 10 MHz to 1 GHz and achieve nearly 8% (min) to 77% (max) of energy savings as compared to VS and ACS.

Further, at near/sub-threshold region of operation, transistors are more susceptible to process variations. The effects of process variation have been increasing as the process technology moves to smaller and smaller feature sizes. In this chapter, we will analyze the impact of process variation in a many-core homogeneous DSP design system. In the presence of process variation, even if a many-core design is built to be homogeneous, it will end up with cores that differ in terms of performance characteristics. In other words, some cores may end up being faster and some being slower than the expected speed. A simple solution to bring
homogeneity back to the design is to run all the cores at the speed of the slowest core in the design. This means that the target throughput may not be achieved and the operating voltage needs to be increased from the original anticipated voltage. Through the work in this thesis, we will show that such an approach will result in a 6.9% to 51.1% more energy consumption as compared to the approach that takes process variation into account while designing the system.

Different cores on the die have a wide range of speed at an operating voltage close to threshold voltage due to process variation. In this work, we also propose a methodology called binning to gather characteristic behavior of each core on the die after fabrication (performed only once) and select active cores to further reduce energy. With the knowledge about the core speeds, the system will not be forced to operate at a speed of the slowest core and hence, can further increase the energy efficiency of the system. This will not always be the case. For example, for certain workload conditions if the optimum number of cores happens to be the total number of cores present on the die, the system will still be limited to run at a speed determined by the slowest core. Furthermore, we will show this methodology will be 49.3% more energy efficient compared to building the system with no knowledge about the characteristics of each core. Later, we will also discuss the overhead incurred in designing such systems.

We have tested our methodologies on various DSP algorithms used mainly for image compression in wireless sensor network application systems. These algorithms include, FFT (Fast Fourier Transform), FIR (Finite Impulse Response), DCT (Discrete Cosine Transform), DWT (Discrete Wavelet transform), RLE (Run Length Encoding) and iDCT (Inverse-Discrete Cosine transform). Even though the number of processing elements (PE), $N$, can be chosen arbitrarily, the control signals and data flow (bus connections, etc.) becomes increasingly complicated and inefficient, if $N$ is not a power of 2. Hence, in this work, we study many-core DSP designs, where a number of PEs is kept power of 2.

Aging or wear-out is another type of variation that is introduced in the integrated circuits
due to circuit activity, temperature, and supply voltage. Aging can cause slow and gradual degradation of the circuit. In this Chapter, we will address the above issue by providing insights into aging effects on homogeneous many-core DSP system operating in Near-Threshold Region (NTR). In addition, we will also compare different policies on selecting the core, and evaluate impact of aging on these policies.

In summary, the main contributions presented in this Chapter are,

1. We present a hybrid methodology consisting of VS and ACS to increase the energy efficiency of a homogeneous many-core DSP designs.

2. We present a system-level approach of judiciously selecting optimal system parameters, namely, the number of cores and its operating voltage to gain energy efficiency for a scalable throughput system under different workload environment.

3. We present the impact of process variation on the system design parameters as well as the energy efficiency of the above systems at 45nm.

4. To further improve the energy efficiency of the system, we present Binning methodology, which selectively chooses active cores from all the cores present on the die after studying their delay variation profile.

5. We study the impact of aging on the speed of all cores and compare two different policies on the lifetime performance of the system.

### 2.1.1 Roadmap for the Chapter

The rest of this chapter is organized in this order: We first present our proposed methodology to build energy-efficient designs and then demonstrate the impact of PV on such designs (Section 2.2 to Section 2.7). Later, we will present our proposed binning methodology which can help mitigate the impact of PV to gain energy-efficiency (Section 2.8). Further, we will discuss the impact of aging variation on the design and usage policies (Section 2.9).
2.2 Background

In this section, we will present background on static variation. Static variations are introduced into the device during the IC fabrication. During the manufacturing process, imperfections in Random Dopant Fluctuation (RDF) and lithography process causes variation in transistors. As shown in the Fig. 2.1, at device level these variations will cause variation in the threshold ($V_{th}$), variation in the channel length ($L_{eff}$) and variation in the interconnect width. At circuit-level, variation in $V_{th}$ and $L_{eff}$ will cause variation in the gate delay and leakage current of the device. Additionally, variation in interconnect width can cause variation in the wire delay. Further, at system-level, variation in the delay can cause performance degradation or total circuit failure.

Below we provide a short description of two major sources of static variation, namely, Random Dopant Fluctuation (RDF), and sub-wavelength lithography.

1. **Random Dopant Fluctuation (RDF)**

The biggest cause of variation is Random Dopant Fluctuation (RDF), where the dopant atoms that are implanted in the transistor channel are either unequal or unevenly distributed.
In older technologies this was never an issue. However, in lower technologies the size of transistors and the number of dopant atoms are small enough that the density can fluctuate significantly. At 65nm technology node, the number of dopant in a transistor is less than 100 atoms which can increase or decrease the threshold voltage of the device [4]. The location of dopant is also important at these smaller dimensions. For example, [4] measured two devices with 50nm channel length and same 170 number of dopant, but had different threshold voltage of 0.78 V and 0.56 V. A fluctuation of 28% in threshold voltage was observed. Deep-sub-micron technology nodes (<0.5 µm channel lengths) have very small transistor dimensions and therefore, it is very difficult to avoid the dopant fluctuations.

2. Sub-wavelength Photolithography

Photolithography is the process of transferring circuit patterns on a layout to the surface of silicon. It uses light to transfer a geometric pattern from a photo-mask on a silicon substrate [84]. Among the photolithography steps, optical lithography (i.e., exposure and development) is the most important step to create nanoscale circuit patterns. The minimum feature size that a projection system can print is determined by the wavelength of the applied light source (λ) and the Numerical Aperture (NA) of the lens. A general used expression for the minimum feature size (s), (maximum resolution) of the resulting projection is given by equation 2.1.

\[ s = k_1 \times (\lambda/NA) \] (2.1)

Where k1 is a constant in the range of 0.4 - 0.7, defined by the lithography process.

Minimum feature sizes can be decreased by decreasing the wavelength. We can see the trend of light used for a photolithography process for different technology nodes in the table 2.1. Moreover, as seen from the Table 2.1, there has not been much advancement made beyond Argon Fluoride (ArF) light usage, which is currently used for 45nm node and below. The change in gate length is often attributed to line edge roughness (LER) and is largely a result of using 193nm light to draw features, which are nearly an order of magnitude smaller. A lot of new promising technologies are emerging in this area such as immersion lithography [28]
Table 2.1: Wavelength of light used for the lithography process.

<table>
<thead>
<tr>
<th>Technology nodes</th>
<th>Light wavelength</th>
</tr>
</thead>
<tbody>
<tr>
<td>250nm</td>
<td>248nm (KrF)</td>
</tr>
<tr>
<td>180nm</td>
<td>248nm (KrF)</td>
</tr>
<tr>
<td>130nm</td>
<td>193nm (ArF)</td>
</tr>
<tr>
<td>65nm</td>
<td>193nm (ArF)</td>
</tr>
<tr>
<td>45nm / 32nm / 22nm</td>
<td>193nm (ArF)</td>
</tr>
</tbody>
</table>

and multiple patterning [90].

2.3 Related Literature

When the operating voltage is above a certain threshold $V_{th}$, the transistors effectively function like a switch, by responding to the changes that come from their gate voltage. Voltage scaling has been a prevalent method for improving the energy efficiency of a digital circuit. Various research teams in the past have explored the possibility of operating the device in this near/sub-threshold voltage region [14–16,95]. Many prototypes have demonstrated the possibility to extend traditional voltage-scaling limit to below the threshold voltage, i.e., sub-threshold voltage region [10,85]. In this work, we leverage the benefits of this methodology in achieving energy efficiency. At the same time, we broadly exploit the parallelism in the application for compensating in the loss of throughput due to operating at low voltages. More detailed and expanded background on sub-threshold design can be found in [87].

Design variation is defined as the variation in parametric results caused by process and environmental changes (process, voltage, temperature). Impact of variations can either be random or systematic. The main source of random variation is due to RDF during the fabrication process as explained in the previous section [96]. The main source of systematic variation is due to layout dependent effects such as parasitic imperfection. RDF introduces large variation in key transistor parameters such as $V_{th}$ and $L_{gate}$. Extensive study has been
done in the past and is ongoing, to analyze and model process variation (PV) effects, particularly on multi-core designs [9, 11, 41, 69]. The primary challenge with Sub/near threshold operation is increase in delay variability. We measure this in terms of delay Coefficient of Variation (COV) \((\sigma/\mu)\). In Fig. 2.2, we illustrate PV impact on a Butterfly FFT core at 90nm and 45nm technology node. The delay variation in this region is so high that if ignored chip can incur 49% to 87% of loss in throughput respectively. In this chapter, we will illustrate that if this effect is ignored, the cost of remedial approaches is much higher compared to simple judicious decisions made at design time.

[85] presented a 180 mV FFT architecture that runs in the sub-threshold region. The resulting throughput is very low, 164 Hz clock cycle at 180 mV and 10 MHz at 350 mV. Our methodology, on the other hand, exploits the parallelization in the processing to achieve the desired throughput. In addition, the architecture in [85] would have to use dynamic voltage scaling to achieve a higher throughput. Additionally, [45] recommends a technique of applying Dynamic Voltage Frequency Scaling (DVFS) for assembly of general purpose processor. However, this technique requires careful calibration, feedback mechanism to adapt and design of the on-chip DC-DC converter. In this chapter, we propose a hybrid methodology of using dynamic voltage scaling and active unit scaling to cater to different throughputs (workload environments) by avoiding the use of on-chip converters and feedback circuits. [55] presented a sub-threshold general purpose processor for the purposes of wireless sensor networks which
is capable of executing simple instructions. However, with the same battery power, this will run very slow on the intended application of sensor network since it lacks application specific hardware. [35] presented an architecture for parallelizing FFT processing element in sub-threshold region in order to achieve min energy operation. However, this analysis is performed only at 222 MHz (nominal voltage operation).

In the last decade, there has been a lot of research towards exploring individual problems such as near/sub-threshold systems, many-core systems, and process variability. For example, [68] demonstrated a methodology to efficiently selecting cores for operation in a many-core systems. However, in their approach they consider only performance (speed) as their optimization matrix and exclude power from their analysis. [38] also explored variability under voltage/frequency islands. Moreover, this approach explored spatial dependencies between the general purpose processors and memory. [52] presented a power-performance simulator to obtain individual power values that can be assigned to individual program fragments.

The above mentioned work as well as a wealth of other work in this area build a solid background for our contributions. However, they either lack the analysis considering the effects of PV on the design system at sub-threshold region of operation or are incomplete in terms of different workload conditions the system can operate in. Therefore, we first propose an algorithm to select the design parameters under different workload conditions. We extend the analysis to show the impact of process variation effects on the design parameters in terms of energy efficiency. Ultimately, we recommend an approach that studies the delay variation profile at runtime to make wiser decisions in selecting the cores.

### 2.4 Design Architecture and Environment

In this section, we will present details of two different types of design modules, Process Variation Ignorant (PVI) and Process Variation Aware (PVA) design. Later, we will describe
two different design environments in which these modules can be deployed.

1. **Process Variation Ignorant (PVI):**

We define PVI design as a circuit with transistor parameters modeled in a typical condition. The device model used is without considering the impact of process variation.

2. **Process-Variation Aware (PVA) design:**

From circuit-level point of view, there is no difference between a PVI design and PVA design. However, for PVA design, the effect of process variation on the PE core’s speed and power consumption is evaluated and taken into account during the design process. Later, in Section 2.5 we will see, that despite the fact that at the circuit-level, PVI design and PVA design are very similar, there are vast differences in terms of number of cores and operating voltage they pick. In other words, they are different from a system-level perspective.

**Modeling Process Variation**

Detailed description of our methodology is presented later in Section 2.6. Nevertheless, here we provide a brief introduction on our methodology to model process variation at 45nm technology node. Table 2.2 shows variation in $V_{th}$ and $L_{gate}$ at 45nm technology [40].

From the percentage variation listed in the table, we now characterize new device model using Predictive Technology Model (PTM) to study the effects of process variation at 45nm technology node [5, 27]. The architecture file (RTL/Netlist/Spice) is the same for both PVA and PVI designs. However, for PVA design, when we perform Monte Carlo SPICE simulations, we model the delay (statistical 3$\sigma$ distribution) and provide multiple delays that the same critical path may experience in the presence of process variation. We choose

<table>
<thead>
<tr>
<th>Technology</th>
<th>45nm</th>
</tr>
</thead>
<tbody>
<tr>
<td>3$\sigma$ $V_{th}$</td>
<td>10%</td>
</tr>
<tr>
<td>3$\sigma$ $L_{eff}$</td>
<td>40%</td>
</tr>
</tbody>
</table>
30 chip instances and measure the worst case delay from all 30 measurements recorded. We then select the largest delay of the critical path from all these measurements. This is \( (\mu+3\sigma) \) worst case delay of the critical path under the effect of process variation. This delay will be larger than the delay of the critical path in a typical condition (PVI). Further, 30 chips are chosen mainly because the HSPICE simulator engine covers 99.6% of the distribution with 30 instantiations. Later, in Section 2.6, we will present a fast simulation framework to carry out the above analysis in both PVI and PVA designs.

### 2.4.1 Static Environment

We define an environment in which a chip is fabricated to deliver pre-determined desired throughput as static environment. Such a design will have a fixed number of cores and fixed operating voltage. For a fixed target throughput, we choose the optimal number of cores \( N_{opt} \) and operating voltages \( Vdd_{opt} \) for both PVI and PVA designs. The algorithm to choose system parameters is discussed in Section 2.5.1. After fabrication, both designs are subject to process variation, whether it has been taken into account or not. The application distributes the job over the assembly of parallelized processing elements to achieve the desired throughput. However, now the maximum speed will be determined by the slowest core in the system.
Figure 2.4: Tradeoff between power-gating and clock-gating schemes for an FIR filter.

In the field, PVI design due to PV effects on speed will be slower than what was estimated during the design stage. This means that the PVI design may not meet the targeted throughput. The PVA design, on the other hand, will readily meet the target throughput because its design has already considered process variation effect on the speed of its cores.

For the PVI design, two remedial approaches can be considered:

1. The operating voltage of the design can be increased (once at the end of testing and fixed afterwards) until it reaches the targeted throughput with the same number of fabricated cores. We call this method Voltage Increase (VI).

2. The number of active cores is reduced by power gating and the operating voltage of the active cores is increased until it reaches the targeted throughput. We call this method Power Gating (PG).

While the first approach solely depends on the voltage increase of the cores, the second approach provides an extra knob to change the number of cores, while increasing the operating voltage.

CG technique is faster to apply as only the clock is turned off and therefore, the design will still dissipate leakage power during the inactive mode of operation. For a system with a long inactive idle periods, this is not very efficient. Meanwhile, using PG technique, power supply...
to the core will be disconnected using transistor switches such as headers. Significant power savings can be achieved using PG. However, there will be a performance penalty when the cores are active due to leakage through the header switches as well as an energy penalty while Powering Up (PU) and Powering Down (PD) of the cores. The tradeoff between CG and PG for an FIR-filter is shown in Fig. 2.4. By using 32 header switches, the PU and PD time are found to be 14 and 8 clock-cycles, respectively, which are not significant. In the current design targeted towards ultra-low power applications, PG technique is preferred due to sizable leakage savings that can be achieved for long periods of idle time.

Based on [34], for 45nm technology, we model total leakage power and loss in performance due to adding a header (power gating circuit). This methodology uses an exhaustive search algorithm to find an optimum width of the header (power gating transistor) at a given operating voltage. Once it finds the width, depending on the total cells in the design that needs to be power gated, it will estimate the total number of headers needed to perform power gating. We can then calculate the total leakage power due to power gating cells, which is nearly around 16% - 27% of the total leakage power. Given the dominance of leakage power at near/sub-threshold region this amounts to a significant consumption overall. In our experiments, we incorporate this impact while calculating the total power consumption with power gating.

The architecture for the static (non-scalable) design is shown in Fig. 2.3. If VI method is used, the power gating module is not needed. Also, there is only one voltage source that is regulated and fixed after testing and before deployment in the field. Even though these techniques improve the yield by bringing the chip to meet the requirement, it does not mean that the overall energy consumption is the same or better than the PVA design, where the effects of process variation is taken into account from the beginning. Later, in the result Section 2.7.2, we will present the energy consumption differences between PVI and PVA designs for various DSP core systems.
2.4.2 Dynamic Environment

Unlike the static design, dynamic design is built such that it can adapt to a desired throughput in the field. This type of application is highly desirable for environments, where the requirement of the system changes over time (such as an increase or decrease in the resolution).

There are a certain pre-determined number of PE cores available in the system along with four different operating voltages to choose from. In order to scale down the throughput, the traditional approach is to employ Voltage Scaling (VS) and use lower voltage settings. However, in such parallel system, another knob is available and that is to reduce the number of active cores by power gating. We call this method Active Core Scaling (ACS). We propose a hybrid methodology, which combines VS and ACS to achieve maximum energy efficiency.

The architecture of this system is illustrated in Fig. 2.5. It consists of multiple processing element (PE) cores. An event detector or a user will decide on a desired throughput. For example, in case of an acoustic sensor that is employed in a field to detect passing vehicles the resolution of the filter needs to be increased when an interesting event is detected. Based on the desired throughput, the controller will then choose the desired voltage and number of PEs to deliver the throughput. The controller uses a lookup table or dynamically calculates the voltage setting and number of cores based on an algorithm that is described in the next
The controller puts the unneeded cores into sleep, by either clock gating (CG) or power gating (PG). As the four voltages are predetermined, the design methodology offers very easy configuration on the field and alleviate the need to use on-chip DC-DC converter. The challenge in designing such a system, is to select the total number of cores as well as the four fixed voltages that will be available during the deployment. These options will highly depend on the maximum desired throughput and the distribution of the workload amongst different desired throughputs. For the results presented in this study, we have assumed a maximum throughput of 1 GHz and a distributed workload between 10 MHz to 1 GHz. The controller stores the configuration information within an on-chip local memory for various desired throughputs. The same controller is also responsible for clock-gating or power-gating inactive cores.

Even though the dynamic design is more flexible, it will be 6.5% to 26.4% less energy efficient compared to a static design, if it ends up operating at a fixed throughput. The main purpose of this work is not to show the difference between a dynamic and static approach. Rather, we seek to show the differences between a PVI and PVA dynamic design as well as the efficiency of the hybrid throughput scaling approach as opposed to simple voltage scaling.

### 2.5 Design Parameter Selection

In this section, we present an algorithm which chooses minimum energy architecture for different throughputs under two different workload environments; static and dynamic. The algorithm selects system parameters such as number of PEs and their operating voltages. The algorithm is based on exhaustive search between the tuples which will use minimal energy to deliver the targeted throughput. For energy scavenging applications conserving energy is of primary concern and hence in this algorithm we try to optimize for total energy/operation. The cost of the system is mainly dominated by the total number of cores, but in practice we cannot have an unlimited area budget. In all the experiments presented in this chapter
Figure 2.6: (A) Choosing the set \([N V_{DD}]\) for a targeted throughput of 330 MHz in the Static Environment. (B) Min Energy point for a PVI and PVA design.

Table 2.3: Optimal tuples of \([N_{opt} V_{dd_{opt}}]\) at different throughputs under static environment, for an FFT system.

<table>
<thead>
<tr>
<th>Throughput</th>
<th>10 MHz</th>
<th>50 MHz</th>
<th>100 MHz</th>
<th>200 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVI ([N_{opt} V_{dd_{opt}}])</td>
<td>[2 0.28]</td>
<td>[16 0.26]</td>
<td>[8 0.34]</td>
<td>[16 0.34]</td>
</tr>
<tr>
<td>PVA ([N_{opt} V_{dd_{opt}}])</td>
<td>[4 0.37]</td>
<td>[16 0.33]</td>
<td>[32 0.31]</td>
<td>[32 0.36]</td>
</tr>
</tbody>
</table>

we have used 256 cores as an upper bound. In those cases where it favors a larger number of cores, the optimum voltage is adjusted (increased) to make \(N_{opt}\) 256 cores.

2.5.1 Choosing \(N_{opt}\) and \(V_{dd_{opt}}\) in the Static Environment

In this section, we will describe our methodology in choosing \(N_{opt}\) and \(V_{dd_{opt}}\) for PVI and PVA design in a static design environment, i.e. when operating at predetermined fixed throughput.

We first use voltage scaling to obtain the speed of a single core at different operating voltages. As shown in Fig. 2.6 (A) for different number of cores, we now plot throughput achievable by parallelized system, over different operating voltage. For desired throughput (330 MHz in the figure), we then determine the fixed static voltage necessary for different number of cores.
Under static environment, since the throughput remains fixed throughout its operation, getting this fixed voltage on chip will not be a problem. For different cores \([N \, V_{dd}]\), we then choose the most energy efficient set \([N_{opt} \, V_{dd_{opt}}]\) for the given desired throughput.

In Fig. 2.6 (B), we plot the impact of voltage scaling on the energy of a single core FFT system, for both PVI and PVA design. The minimum energy point for a design will happen at a given voltage \(V_{min}\). However, when we consider scalable throughput systems and multiple active units, the \(V_{min}\) instead of a point in the graph, it will be a window. This window is marked for both PVI and PVA designs in Fig. 2.6 (B). PVA designs may operate at a slowest speed obtained in the distribution. Hence, such systems will be slower in operation as compared to the PVI system for each operating voltages. As shown in Fig. 2.6(B), PVA systems will trail behind PVI. We have found that for all throughputs, the optimum voltage will always be in this \(V_{min}\) window. For the PVI design the minimum energy point for a single core is around 0.23 V to 0.35 V. However, the minimum energy point of the PVA design is slightly higher, at around 0.32 V to 0.48 V.

Table 2.3 shows the \([N_{opt} \, V_{dd_{opt}}]\) tuples for a range of desired throughputs. It can be seen that the operating voltage of assembly of parallel units, i.e. \(V_{dd_{opt}}\) for PVI and PVA designs, hover around their respective minimum energy point at the time of design. During the deployment, when the effects of process variation is evident, the operating voltage of the PVI design will have to increase to make up for those effects. If \(N_{opt}\) for PVI and PVA design happens to be the same, there will be no penalty even at the deployment stage. As in that case, we can increase the static fixed voltage to deliver the desired throughput. However, in the case when \(N_{opt}\) for PVI and PVA design differs \((N_{PV-I} < N_{PV-A})\) or \((N_{PV-I} > N_{PV-A})\), there will be penalty in the energy savings. We examine this later in the result section 2.7.2.

### 2.5.2 Choosing \(N_{opt}\) and \(V_{dd_{opt}}\) in the Dynamic Environment

In this section, we describe our methodology in choosing \(N_{opt}\) and \(V_{dd_{opt}}\) for PVI and PVA designs in a dynamic environment.
a) For operating voltage range [0.2* – 0.9] with a step of 50mV, model the speed and throughput achievable \([T_A]\) by single core using HSPICE.
b) For the same operating voltage range, measure the power component \((P)\) using PrimeTime and HSPICE.
c) For Throughput-Desired \([T_D]\) range of [10Mhz – 1GHz] {
   1. For every operating voltage range \(V\), calculate \(N = \frac{T_D}{T_A}\)
   3. Round of \(N\) to closest factor of \([1,2,4,8,16,32,64,128,256]\)
   3. Calculate the total power consumption \(P = P_{measured} \text{ at } V \times N\)
   4. For every operating voltage range \(V\), identify the min power set \(P\) and \(N\). This is \(N_{opt}\) and corresponding \(V_{dd_{opt}}\) for the desired throughput.
   * Calculate the smallest voltage at which design is functionally operating. Choose this as lower boundary.

Figure 2.7: An algorithm to find the most energy efficient system design parameters, under a dynamic environment.

Figure 2.8: (A) Selecting the number of cores and operating voltage based on throughput for FFT processor\(^a\). (B) Selecting the most energy optimum set of \([N \ V_{dd}]\) for FFT processor.
\(^a\)For better visibility, the points in the graph which cannot deliver 330 MHz are trimmed to 10MHz frequency

As described before, dynamic environment offers flexibility to operate and adapt the system for a range of throughputs. In Fig. 2.7, we present the algorithm used in choosing \(N_{opt}\) and \(V_{dd_{opt}}\) i.e. design-time parameter selection. Both static and dynamic designs try to meet desired throughput by varying the number of cores and operating voltages. The difference, however, is that the operating voltages available are restricted. We first make the assumption that any operating voltage between 150 mV (or min voltage for the circuit to function) and 600 mV at 50 mV intervals can be picked for the cores. The minimum energy point for a PVI design for single core is at a lower voltage than PVA design as seen in Fig. 2.6(B). Therefore, we observe that the PVI design has a tendency towards choosing a higher number
Table 2.4: Optimum set \([N_{opt} \ Vdd_{opt}]\) at different throughputs under dynamic environment, for FFT processor.

<table>
<thead>
<tr>
<th>Throughput</th>
<th>100 MHz</th>
<th>200 MHz</th>
<th>300 MHz</th>
<th>400 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVI ([N_{opt} \ Vdd_{opt}])</td>
<td>[8 0.25]</td>
<td>[16 0.30]</td>
<td>[16 0.30]</td>
<td>[128 0.25]</td>
</tr>
<tr>
<td>PVA ([N_{opt} \ Vdd_{opt}])</td>
<td>[16 0.35]</td>
<td>[16 0.40]</td>
<td>[32 0.35]</td>
<td>[8 0.50]</td>
</tr>
</tbody>
</table>

Table 2.5: Four optimum voltages for PVI and PVA design under dynamic environment, for FFT many-core system.

<table>
<thead>
<tr>
<th>Optimum operating voltages chosen</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVI</td>
</tr>
<tr>
<td>PVA</td>
</tr>
</tbody>
</table>

of cores with low operating voltages close to its \(Vdd_{min}\) for a single core. On the other hand, PVA design will be optimum at lower number of cores operating at higher voltage, closer to its \(Vdd_{min}\) energy point. As shown in Fig. 2.8 (A), for a desired range of throughputs and available operating voltages with a step of 50 mV, we choose \(N\) such that, it can deliver desired throughput (330 MHz in the figure). For this set of \([N \ Vdd]\) tuples, we can then determine the most energy efficient set \([N_{opt} \ Vdd_{opt}]\) as shown by the peak spike in the Fig. 2.8 (B).

Further, Table 2.4 shows the \([N_{opt} \ Vdd_{opt}]\) tuples for a range of desired throughputs. From the table, we observe that the optimal number of cores for the PVI system is lower than the PVA design system in order to meet the same targeted throughput. In the cases in which the optimal number of cores remain the same, for example 200 MHz desired throughput, the optimal operating voltage for a PVA design system is higher than the PVI design.

If we consider 50 mV of \(Vdd\) resolution, we observe that for the two different design approaches, i.e. PVI design and PVA design, not more than four choices are picked as options for optimum operating voltage. The four voltages chosen by PVI and PVA design for FFT system is shown in Table 2.5.
Table 2.6: List of library and tools used to conduct experiments.

<table>
<thead>
<tr>
<th>Technology</th>
<th>UMC standard 90nm, PTM Model 45nm</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDL Language and Simulation</td>
<td>Verilog, Synopsys VCS</td>
</tr>
<tr>
<td>Voltage Scaling characteristics</td>
<td>Synopsys HSPICE, NanoSim</td>
</tr>
<tr>
<td>Power Simulation</td>
<td>Synopsys Prime Time PX</td>
</tr>
</tbody>
</table>

In both designs, the system is built considering the maximum number of cores necessary to deliver all throughputs in a given workload environment. When deployed in the field, this method will be highly inefficient due to the large number of inactive cores. Unlike the static approach, in dynamic environment both PVI design and PVA design need to use either clock gating (CG) or power gating (PG) to turn off the cores that are not needed at throughputs which uses cores less than the total available on the die. Later, in the Section 2.7.3, we will examine the energy efficiency of both designs. One should note that the clock gating and the power gating approach incurs some extra area and power overhead due to gating transistors.

2.6 Experimental Setup

In this section, we will describe the methodology used for conducting the experiments. Table 2.6 gives the summary of the library and tools used for all experimentation.

The traditional approach for carrying out a detailed timing analysis is to synthesize the design netlist for a targeted library, and convert this netlist into a SPICE netlist. Later, SPICE simulation is performed in order to get an accurate delay of the circuit by using a specific transistor model. This is true for the PVI and PVA designs. The problem with the above simulation framework is that it can be very time consuming, especially with large circuits and large number of critical paths. In addition, if one needs to perform Monte Carlo simulation to get a distribution of delays in the presence of process variation, this analysis will take even longer. For our study, we have developed a fast simulation framework for doing timing analysis.
Table 2.7: Fast Simulation Framework.

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Stages in Critical Path</th>
<th>Delay Error Percentage</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit 1</td>
<td>40</td>
<td>6.29</td>
<td>240x</td>
</tr>
<tr>
<td>Circuit 2</td>
<td>20</td>
<td>1.20</td>
<td>520x</td>
</tr>
<tr>
<td>Circuit 3</td>
<td>18</td>
<td>3.11</td>
<td>257x</td>
</tr>
<tr>
<td>Circuit 4</td>
<td>15</td>
<td>0.62</td>
<td>314x</td>
</tr>
<tr>
<td>Circuit 5</td>
<td>7</td>
<td>1.26</td>
<td>600x</td>
</tr>
</tbody>
</table>

Table 2.8: Fast Simulation Framework.

<table>
<thead>
<tr>
<th>Module name</th>
<th>Total critical paths</th>
<th>Critical paths considered</th>
<th>Percentage of paths considered</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFT</td>
<td>700</td>
<td>73</td>
<td>10.43%</td>
</tr>
<tr>
<td>FIR</td>
<td>224</td>
<td>43</td>
<td>19.19%</td>
</tr>
<tr>
<td>DCT</td>
<td>1601</td>
<td>192</td>
<td>11.99%</td>
</tr>
<tr>
<td>DWT</td>
<td>72</td>
<td>6</td>
<td>8.33%</td>
</tr>
<tr>
<td>RLE</td>
<td>199</td>
<td>26</td>
<td>13.06%</td>
</tr>
<tr>
<td>iDCT</td>
<td>1607</td>
<td>192</td>
<td>11.94%</td>
</tr>
</tbody>
</table>

The FFT design in our study consists of roughly 14K cells. With a 16-core machine and 64-bit binary, Monte Carlo analysis takes nearly 12 hours to finish. To circumvent this problem we extract the critical paths from the design. With this, now an entirely new circuit is built which consists of critical paths. Justifying input values to stimulate this path are then extracted from Prime Time and applied in HSPICE. To increase the accuracy, we also consider fanouts of the original path, which may / may not be part of the critical path. We incorporate the effect of interconnect loads on these fanout paths. We have summarized results using this methodology in Table 2.7. As shown, for the different circuits considered the maximum error is 6% with an average 380x speedup. Further analysis on all circuits as well as different methodologies compared in this chapter is done by applying this framework.

All the designs are optimally pipelined. Since DSP architecture consist of adders and multipliers, all designs will have many balanced critical paths of equal depth and delay. We
bin the critical paths based on their negative slack and choose the worst bin for the above analysis. Table 2.8 shows total critical paths in all the designs and the total number of paths considered for analysis. There can be an argument made that due to process variation what if the critical paths from the adjacent bin become the most critical path? Therefore, for our analysis we consider more than 10% of all critical paths in the design. With this, the probability that the most critical path after fabrication will still be part of the considered bin is very high.

2.7 Results

This section summarizes the results and our analysis of various experiments. We first present the energy savings by using a hybrid technique over traditional VS and ACS techniques. We present the energy savings of PVA design over PVI in both the design environments, static and dynamic. Later, we present the variation in battery lifetime of the system when operated with 2 AA batteries. This will help us gain insight into the degree of energy efficiency achievable by both designs. The accuracy of all the results presented in this chapter is the

Figure 2.9: Comparison of power efficiency obtained by hybrid technique over VS and ACS (Clock Scaling).
Figure 2.10: Power efficiency of a PVA and PVI design for an FFT processor under a static environment.

accuracy of the numbers reported by the tools, which in this case is 1 ps.

2.7.1 Energy Efficiency of Hybrid Technique

In Fig. 2.9, we plot the total power consumed by using a hybrid technique of VS and ACS at different scalable throughputs. We compare this with the traditional technique of Voltage Scaling and Active Unit Scaling. For low throughput (10 MHz - 100 MHz), hybrid methodology is 8% to 36% better, whereas for high throughput (600 MHz - 1 GHz), hybrid technique proposed will be on an average 75% more efficient than traditional methods.

2.7.2 Energy Savings in a Static Environment

Table 2.9 shows energy savings of PVA design when compared with PVI design using PG (most efficient in PVI), over various DSP application systems. From the table, we observe that PVA design enjoys an average of 10.2% to 26.9% reduction in energy as compared to PVI design. The efficiency is evaluated over a throughput range of 10 MHz to 1 GHz. We evaluate
Table 2.9: Percentage energy savings of PVA design over PVI with PG, under both static and dynamic environment.

<table>
<thead>
<tr>
<th>Cores</th>
<th>Static Environment</th>
<th>Dynamic Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Max</td>
</tr>
<tr>
<td>FFT</td>
<td>17.0</td>
<td>29.0</td>
</tr>
<tr>
<td>FIR</td>
<td>2.5</td>
<td>34.5</td>
</tr>
<tr>
<td>DCT</td>
<td>12.0</td>
<td>20.0</td>
</tr>
<tr>
<td>DWT</td>
<td>3.7</td>
<td>12.5</td>
</tr>
<tr>
<td>RLE</td>
<td>16.0</td>
<td>45.4</td>
</tr>
<tr>
<td>iDCT</td>
<td>10.2</td>
<td>20.0</td>
</tr>
<tr>
<td>Average</td>
<td>10.2</td>
<td>26.9</td>
</tr>
</tbody>
</table>

minimum and maximum savings for every application over this range of fixed throughput. The current numbers reported is considering 100% duty cycle, i.e. considering the system is always active (worst case scenario). However, in wireless sensor node applications, generally they may operate at 15% duty cycle [35]. In such scenarios, energy efficiency gain will be even higher.

In Fig. 2.10, we plot the total power consumption per FFT, at the time of deployment for PVI and PVA design. Here, PVA design is compared with PVI design after applying compensation discussed before, such as VI and PG. We observe that PVA design in this case, is consistently better for all throughputs.

### 2.7.3 Energy Savings in a Dynamic Environment

In a dynamic environment, we analyze the energy efficiency of PVI and PVA designs under different workload conditions. Such as, low workload (10 MHz - 100MHz), medium workload (100 MHz - 600 MHz), high workload (600 MHz - 1 GHz), and uniform workload conditions. From the Table 2.9, PVA designs are 53.3%, 29.7%, 16.7%, and 32.7% more efficient than
Figure 2.11: Power efficiency of a PVA and PVI design for an FFT processor under a dynamic environment.

PVI design under low, medium, high, and uniform workload conditions respectively. As we had restricted the maximum number of cores to 256, for high workload condition, both PVA and PVI design may use maximum cores at their respective $V_{dd_{opt}}$. Therefore, for high workload conditions, energy savings are comparatively less. However, at low workload, PVA design uses a lesser number of cores and higher operating voltages as compared to PVI design, which uses a higher number of cores and lower operating voltages. Hence, under low workload conditions, the energy savings are much higher. On an average overall savings in the dynamic environment is 18% - 51.5%.

In Fig. 2.11, we show the total power consumption per FFT under a dynamic environment. Here, PVA design is compared with PVI design after applying compensation as discussed before, such as CG and PG. PVA with PG for FFT system, has consistent energy gains as compared to PVI with PG (best design in PVI).
Table 2.10: Percentage improvement of PVA design over PVI in 90nm and 45nm technology.

<table>
<thead>
<tr>
<th>Energy savings</th>
<th>Static Environment</th>
<th>Dynamic Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>90nm</td>
<td>10.0% - 22.0%</td>
<td>10.1% - 43.5%</td>
</tr>
<tr>
<td>45nm</td>
<td>17.2% - 29.1%</td>
<td>32.5% - 71.1%</td>
</tr>
</tbody>
</table>

2.7.4 Technology Trends

We initially started our analysis at 90nm technology node. However, as we know the variation effect worsens with shrinking technology nodes, we extend the analysis with 45nm and the results and graphs presented here are for 45nm. In order to understand the gravity of PV effects on shrinking technologies and evaluate our methodology, we present the same analysis for a system with FFT PEs at 90nm. As expected, at 90nm the effects of process variation are not as pronounced as in the 45nm. The number of cores and their operating voltages chosen in PVI and PVA design are largely the same at 90nm and the energy reductions are less compared to 45nm. Table 2.10, gives the percentage improvement of PVA design compared to the PVI design under static and dynamic environment (uniform workload).

2.7.5 Battery Lifetime Modeling and Estimation

We evaluate energy gains discussed before for both designs in terms of battery lifetime for a given application. Battery lifetime in this context, is considered as the number of days an application can run on two heavy duty Alkaline AA batteries. Each Alkaline battery can deliver 1800 mAh - 2600 mAh at 1.5 V ~ 325 mW/day. We have considered worst case scenario, in which each application runs at 100% duty cycle (always on). However, lower duty cycles are conceivable for various scenarios. In Fig. 2.12(A), we plot normalized lifetime achievable under a dynamic environment. We observe PVA will run 70% longer at lower throughput to 50% longer at high throughput under a dynamic environment. In Fig. 2.12(B), we plot normalized battery lifetime achievable under static environment. We observe PVA will run 23% longer at lower throughput to 17.6% longer at higher throughputs.
In all conditions, we observe that the design which considers the effect of process variation during designing (PVA) is more energy efficient and has higher battery lifetime.

### 2.8 Design Corner Aware (Binning) Methodology

In this section, we present a methodology to further improve the total efficiency of the system, under a dynamic environment. We call this methodology *Design Corner Aware (DCA)* or *binning*. The overall concept is similar to speed-binning of the chips after fabrication, in which chips are sorted based on their speed. However, here we expand this methodology of sorting of cores at runtime within a single die and selectively choosing cores based on the throughput requirement and workload environment. In the first part of the section, we present the concept of *binning*; followed by the design and implementation details of the controller. Later, we discuss the overheads in designing this system in terms of area and
Normal Distribution of cores based on Peak speed

Cores in Bin B0 operate at a max speed of \(\frac{1}{(\mu - \sigma)}\)

Cores in Bin B1 operate at a max speed of \(\frac{1}{\mu}\)

Cores in Bin B2 operate at a max speed of \(\frac{1}{(\mu+\sigma)}\)

Cores in Bin B3 operate at a max speed of \(\frac{1}{(\mu+2\sigma)}\)

Cores in Bin B4 operate at a max speed of \(\frac{1}{(\mu+3\sigma)}\)

Figure 2.13: Normal distribution of the cores based on their speed of operation.

additional clock-cycles needed for calibration.

In Section 2.5.2, we explained a hybrid methodology of choosing optimum design parameters, namely, \(N_{opt}\) and \(V_{dd_{opt}}\) for a system at different throughputs. At runtime, we do not have information about the speed characteristics for each core. Keeping that in mind, we had designed the system in a pessimistic way; wherein, modeling of the design parameters were done considering one of the cores can be from the worst corner (slowest). However, if by using some mechanism, we can garner information about the variation in speeds for each core at all four optimum operating voltages, we can selectively choose cores so that it aids in improving the energy efficiency.

Several research groups [21,32,64] have analyzed the delay distribution (or speed) of circuits under the effect of process variation using fabrication results. [21] has shown that on 80-core chip, delay distribution due to process variation follows a normal distribution. Fig. 2.13 shows a normal distribution of critical path delay \(\left(\frac{1}{f_{max}}\right)\) in a DCT core design. In a many-core system, the speed of all the design cores will fall under this distribution curve. For example, earlier in section 2.5.2, we saw that the DCT application will need \(64 \times 64\)
Table 2.11: Binning of cores based on their critical path delay.

<table>
<thead>
<tr>
<th>Bins</th>
<th>Critical Path Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min Range</td>
</tr>
<tr>
<td>B0</td>
<td>$\mu - 3\sigma$</td>
</tr>
<tr>
<td>B1</td>
<td>$\mu - \sigma$</td>
</tr>
<tr>
<td>B2</td>
<td>$\mu$</td>
</tr>
<tr>
<td>B3</td>
<td>$\mu + \sigma$</td>
</tr>
<tr>
<td>B4</td>
<td>$\mu + 2\sigma$</td>
</tr>
</tbody>
</table>

Table 2.12: Summary of design implementation and testing overheads of the calibration module.

- Clock cycles were calculated for DCT application based on 8 x 8 matrix data input and a resolution of 12-bit coefficients for 10 test patterns.
- Gate Equivalent (GE) is calculated by dividing the post-layout area by the area of a 2 input NAND cell.

<table>
<thead>
<tr>
<th>Overhead</th>
<th>Total bins 1</th>
<th>Total bins 3</th>
<th>Total bins 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock cycles&lt;sup&gt;a&lt;/sup&gt;</td>
<td>327680</td>
<td>983040</td>
<td>1638400</td>
</tr>
<tr>
<td>Area (GE)&lt;sup&gt;b&lt;/sup&gt;</td>
<td>103.4</td>
<td>310.3</td>
<td>517.2</td>
</tr>
</tbody>
</table>

cores to cater to all the throughputs from 10 MHz to 1 GHz. If we segregate the cores under different corners such as fast, nominal and slow corner as shown in Table 2.11, there will be approximately 17 cores in Bin B0. For some desired throughput (low), if we need only (2, 4, 8, 16) cores we can activate the cores from this bin and operate the overall system at frequency determined by the bin boundary. In this case, it will be $1/(\mu - \sigma)$ frequency rather than $1/(\mu + 3\sigma)$ (slowest). We call this methodology of selectively choosing the cores as binning. Earlier approach of choosing the design parameters by assuming one of the cores may be from the slowest corner is called slowest in our analysis. Later in the section 2.8.2, we show that with binning the system can achieve 45.1% to 50.1% better energy efficiency as compared to the slowest.
2.8.1 Design and Implementation of the Controller

In this section, we will discuss the details of the controller to implement binning and the overhead it adds in terms of area and clock cycles.

The State Machine ($SM$) of the controller is shown in Fig. 2.14. When the system is powered on, $SM$ goes to a reset state. The system then enters the calibration mode. In this state cores are tested to identify the corner in which they lie and their core IDs are now labeled into different bins. After all cores are calibrated $SM$ then goes into an idle state where it will wait for an event to process. Once an event is detected it will calculate the throughput desired to process this event. $SM$ then goes into the configuration mode. In this state, depending on the desired throughput and cores available under different bins, it will now choose the design parameters; $N_{opt}$ and $Vdd_{opt}$. Further, with the help of the core ID it will identify specific cores that should be activated.

Calibration process involves an overhead in terms of area and clock cycles in order to perform binning. Table. 2.12 shows the overheads for different levels of binning. From the table, by increasing the number of bins i.e. fine-grain binning, the overhead in terms of clock cycles
For desired_throughput {
  1. Get tuples \([N_{opt} \ Vdd_{opt}]\) at all different speed corners.
     Fastest\([N_{opt} \ Vdd_{opt}]\), Fast\([N_{opt} \ Vdd_{opt}]\), Nominal\([N_{opt} \ Vdd_{opt}]\), Slow\([N_{opt} \ Vdd_{opt}]\), Slower\([N_{opt} \ Vdd_{opt}]\), Slowest\([N_{opt} \ Vdd_{opt}]\).
     If Fast\([N_{opt}] <= B0[\ Vdd_{opt}]\) {
         choose chip IDs from bin B0
     } elseif Nominal\([N_{opt}] <= B0[\ Vdd_{opt}] + B1[\ Vdd_{opt}]\) {
         choose chip IDs from bins B0 and B1
     } elseif Slow\([N_{opt}] <= B0[\ Vdd_{opt}] + B1[\ Vdd_{opt}] + B2[\ Vdd_{opt}]\) {
         choose chip IDs from bins B0, B1 and B2
     } elseif Slower\([N_{opt}] <= B0[\ Vdd_{opt}] + B1[\ Vdd_{opt}] + B2[\ Vdd_{opt}] + B3[\ Vdd_{opt}]\) {
         choose chip IDs from bins B0, B1, B2 and B3
     } elseif Slowest\([N_{opt}] <= B0[\ Vdd_{opt}] + B1[\ Vdd_{opt}] + B2[\ Vdd_{opt}] + B3[\ Vdd_{opt}] + B4[\ Vdd_{opt}]\) {
         choose chip IDs from bins B0, B1, B2, B3 and B4
     }
}

Figure 2.15: An algorithm to select the design parameters tuples \([N_{opt} \ Vdd_{opt}]\) based on binning methodology.

and area increases. However, with higher number of bins, it will offer more control on the system speed and hence can achieve higher energy efficiency. Depending on the application, designer can make this trade-off between energy efficiency and overhead. Currently, calibration assumes serial testing of all cores. Further optimization can be achieved by parallelizing the calibration procedure and thereby reducing the clock-cycle overhead.

We also recommend using few backup cores (3% of the total cores in the system). This is helpful for two main reasons. One, backup cores can be used for fault tolerance, i.e. if few cores fail to operate due to manufacturing faults, we can substitute them. Such cores can easily be activated or deactivated only once, using on-chip fuse [36]. Two, 2.1% of the core that fall in the region between \((\mu + 2\sigma \text{ to } \mu + 3\sigma)\), i.e. the slowest bin can now be excluded from the configuration.

Fig. 2.15 illustrates the algorithm used for the configuration module. For a given total number of cores available on the chip, we identify the optimum number of cores \(N_{opt}\) and respective optimum operating voltage \(Vdd_{opt}\) necessary to deliver the desired throughput. Like previous analysis, throughput range of 10 MHz to 1 GHz is considered. \(N_{opt}\) and \(Vdd_{opt}\) will be different if the cores are chosen from different bins. For the desired throughput, if the number of optimum cores \(N_{opt}\) happens to be less than or equal to the cores available in Bin B0, we select the cores based on the core ID stored in B0. However, if adequate number
Table 2.13: \([N_{opt} \ V_{dd\ opt}]\) tuples for different design corners at 500 MHz desired throughput, DCT core.

<table>
<thead>
<tr>
<th>Desired Throughput 500 MHz</th>
<th>Design corners</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Bin B0</td>
</tr>
<tr>
<td>([N_{opt} \ V_{dd\ opt}])</td>
<td>[64 0.30]</td>
</tr>
<tr>
<td>Total Power (W)</td>
<td>9.9e-03</td>
</tr>
<tr>
<td>Energy Efficiency w.r.t Slowest corner</td>
<td>68.4%</td>
</tr>
</tbody>
</table>

of cores are not available, the controller then checks for the next bin. In the worst case scenario, if all cores available on the die are necessary to achieve the desired throughput and if one of the cores happens to be in the slowest corner, the system will still be operating at a slowest bin boundary.

Let us consider an example of the DCT application system. Table 2.13 shows the tuples \([N_{opt} \ V_{dd\ opt}]\) for different bins necessary to achieve 500 MHz throughput. If the cores are selected from the Bin B0, we will need 64 cores operating at 0.3 V. The configuration module will check if 64 cores are available in this bin. If it fails, it will then get the tuples for the next bin B1. For bin B1, we will need 32 cores operating at 0.4 V. If 32 cores are available in (Bin B0 and Bin B1), we can select the required number of cores. The system in this case will be operating at a speed determined by Bin B1 boundary. The algorithm checks iteratively for all bins. From Table 2.13, we can see if the cores are chosen from Bin B0 (fast corner), they will be 68.4% energy efficient as compared to the slowest corner.

2.8.2 Results of Binning

In this section, we will present the results obtained using binning as compared to slowest method.

1. Energy Efficiency with binning

Table 2.14, shows the energy efficiency of binning as compared to the slowest for all
DSP design applications. We analyze them under four different workload environments; low workload [10 MHz - 100 MHz], medium workload [100 MHz - 600 MHz], high workload [600 MHz - 1 GHz], and uniform workload. From the table, binning is 45.1% - 50% better than slowest for all workloads. Further, Fig. 2.16 shows the energy efficiency of three different applications using DCT, FIR, and RLE cores in a uniform workload environment. From the figure, we can see for DCT application, the gains are consistently higher for all throughputs (low to high). However, for FIR application, as the design is small, the speed distribution and variations are low. Hence, the energy gains are fairly small, nearly 14.4% for all throughput. For RLE application, we can see that for lower throughputs, the gains are smaller. This is due to the fact that \( N_{opt} \) \( Vdd_{opt} \) chosen for lower throughputs in both the cases - binning and slowest, are the same. However, the difference between them begins to grow as desired throughput increases and peaks to 56%.

2. Battery lifetime improvement with binning
Table 2.14: Energy efficiency of the system with *binning* as compared to the system operating at the *slowest* core speed.

<table>
<thead>
<tr>
<th>Cores</th>
<th>Low Workload [(10-100)MHz]</th>
<th>Medium Workload [(100-600)MHz]</th>
<th>High Workload [(600-1000)MHz]</th>
<th>Uniform Workload</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFT</td>
<td>21.1</td>
<td>23.1</td>
<td>54.8</td>
<td>64.4</td>
</tr>
<tr>
<td>FIR</td>
<td>14.4</td>
<td>28.7</td>
<td>11.5</td>
<td>14.4</td>
</tr>
<tr>
<td>DCT</td>
<td>66.2</td>
<td>69.7</td>
<td>68.4</td>
<td>63.1</td>
</tr>
<tr>
<td>DWT</td>
<td>54.7</td>
<td>55.6</td>
<td>47.7</td>
<td>47.5</td>
</tr>
<tr>
<td>RLE</td>
<td>50.8</td>
<td>56.6</td>
<td>54.3</td>
<td>45.2</td>
</tr>
<tr>
<td>iDCT</td>
<td>63.4</td>
<td>60.4</td>
<td>63.9</td>
<td>60.9</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>45.1</strong></td>
<td><strong>49.0</strong></td>
<td><strong>50.1</strong></td>
<td><strong>49.3</strong></td>
</tr>
</tbody>
</table>

Figure 2.17: Battery lifetime achievable by *binning* and *slowest* methodology on various DSP systems under a dynamic environment.

We evaluate energy gains discussed above in terms of battery lifetime for a given application. Battery lifetime in this context, is considered as the number of days an application can run at 100% duty cycle on two heavy duty Alkaline AA batteries. In Fig. 2.17, we present these gains for three different applications i.e. DCT, FIR and RLE. We observe DCT system with *binning* will run almost 66.4% longer time under a dynamic environment. Similarly, for FIR and RLE applications, system with *binning*
will run longer by 36.1% and 53.3% respectively, as compared to the slowest. In all applications, we observe that the system with binning will be more energy efficient than with the slowest and therefore will have higher lifetime.

2.9 Analyzing the Impact of Aging

During the normal use, integrated circuits go through what is popularly known as a wear-out or aging [7]. Aging is introduced in the integrated circuits due to circuit activity, temperature, and supply voltage. Aging causes a gradual increase in the threshold voltage ($V_{th}$) of both PMOS and NMOS transistors. At system level, this results in critical paths in the circuit gradually and slowly taking longer time to complete the computation.

For example, in Fig. 2.18, we plot speed deterioration of the iDCT DSP core with respect to the voltage. From the figure, after 0, 1, 2 and 5 years of operation, we can see the deterioration of speed is much more significant (5% - 15%) in the Near-Threshold Region (NTR), compared to nominal voltage. This is because in this region the circuit speed is more dependent on threshold voltage that is highly affected by aging. It is, therefore, important to study the effect of aging in such systems.
2.9.1 Background and Related Literature

In this section, we describe the background related to aging or wear-out. Later, we also discuss related work in this area and present scope of our work. The two main mechanisms that results in aging of transistors are Negative Bias Temperature Instability (NBTI) and Hot Carrier Injection (HCI) [7].

**Negative Bias Temperature Instability (NBTI)** affects in PMOS is a major reliability issue. When an electric field is applied across the gate oxide of a PMOS, traps are generated at the $Si - SiO_2$ interface. [50, 83] explain in detail the degradation process caused by the generation of traps and partial recovery associated with reduction in traps. The effects increases the magnitude of $V_{th}$ of PMOS. Over a long period of time this threshold voltage shift can potentially cause a significant increase in delay of the PMOS devices [46]. With the shrinking technology nodes, NBTI degradation and its impact on circuit reliability and performance have become a key issue. Authors in [17, 47, 58, 67] presented their investigation into analyzing NBTI effects on circuits.

**Hot Carrier Injection (HCI)** is the process in which electrons accelerate under the influence of the electric field of the channel and collide with the gate oxide interface. The collision creates electron-hole pairs. Electrons with high energy (hot) get trapped in the gate oxide layer, causing an increase in $V_{th}$ [80]. HCI induced degradation is largely observed in NMOS transistors due to higher mobility of electrons than in PMOS [88]. Moreover, since hot electrons are generated during circuit activity, the impact of HCI is directly proportional to the switching frequency. Circuits operating at higher supply voltage, elevated temperatures, and with high switching activity are very susceptible to HCI-related aging.

Many researchers looked at the notion of dark silicon, where not all parts of silicon can operate at full speed due to power constraints. An interesting architecture idea such as BubbleWrap [44] was introduced that takes advantage of dark silicon by allowing active cores to run faster and wear out faster, and later replace them with expendable cores. [62] presented cherry-picking concept, which is to select the best subset of cores for an application...
so as to maximize performance within the power budget.

Although all the above mentioned work lay a solid background, however, none of them have looked into analyzing aging impacts on many-core DSP systems especially designed at Near-Threshold-Voltages (NTV) for energy efficiency purposes. In this section, we evaluate the impact of aging along with PV on such systems. Further, we evaluate two different policies in selecting the cores based on aging. Policy A, is to always use the fastest cores available on the die. Policy B, is to attain uniform aging, i.e after a stipulated period of activity, core migration occurs so as to maintain uniform aging across all the cores on the die. Later in Section 2.9.4, we will evaluate the efficiency of these policies on speed degradation due to the aging process.

### 2.9.2 System Overview

Given system consists of pre-determined number of processing elements (PE) cores similar to the Fig. 2.5. In this section, we have considered a system with maximum of 32 cores. The methodology to choose system parameters, i.e. selection of active cores and their operating voltage, is described in detail in Section 2.5.2. This hybrid method picks the number of active cores and selects the operating voltage in order to meet the target throughput with minimal energy. Previously, we showed that the minimal energy for a design is obtained when the operating voltage is close to the threshold voltage of the transistor. Table 2.15 shows some of the energy-efficient configuration consisting of \([N_{opt} \ V_{dd_{opt}}]\) to cater to various throughputs for a 32-core inverse Discrete Cosine Transform (iDCT) design system.

For example, consider at a given time the system needs to cater to 50 MHz throughput. The most energy optimal solution is to use 8 cores operating at 0.3 V. However, since there exist 32 cores on the die, we need to choose which 8 cores to select. Given, it is a homogeneous many-core environment, the choice should not matter as all the cores should have similar speed and power characteristics. Nevertheless, due to PV impact, not all cores will have uniform speed or power characteristics. Some of them will be faster and some of them will
Table 2.15: Energy efficient configurations for iDCT design.

<table>
<thead>
<tr>
<th>Target throughput</th>
<th>Number of cores</th>
<th>Operating voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>25 MHz</td>
<td>4</td>
<td>0.30 V</td>
</tr>
<tr>
<td>50 MHz</td>
<td>8</td>
<td>0.30 V</td>
</tr>
<tr>
<td>75 MHz</td>
<td>32</td>
<td>0.25 V</td>
</tr>
<tr>
<td>200 MHz</td>
<td>16</td>
<td>0.30 V</td>
</tr>
<tr>
<td>500 MHz</td>
<td>32</td>
<td>0.35 V</td>
</tr>
</tbody>
</table>

be slower. Instead, the speed of the cores due to PV within the same die will follow a normal distribution [22].

We evaluate two different policies in selecting the cores based on aging. Policy A, is to always use the fastest cores available on the die. Policy B, is to attain uniform aging, i.e after a stipulated period of activity, core migration occurs so as to maintain uniform aging across all the cores on the die. Nevertheless, such policies are important only when there is a choice to be made. For example, when the number of optimal cores is same as the maximum available on the system, to cater to the throughput, there is no choice to be made. Later in Section 2.9.4, we will evaluate the efficiencies of these policies on speed degradation due to the aging process.

2.9.3 Experimental Setup

To model aging, we use HSPICE MOS Reliability Analysis (MOSRA). MOSRA models, both NBTI and HCI impact on the speed and power of the design. Aging model performs simulation in two phases, which are pre-stress and post-stress. During the pre-stress phase, the degradation in the $V_{th}$ of all the transistors in the chip are measured based on their operating voltage, temperature, activity, etc. In the post-stress phase, the degradation calculated during pre-stress is taken into account to evaluate the effect of aging.
2.9.4 Results

This section summarizes the results and our analysis for various experiments.

1. The impact of aging on the fast and slow cores

The goal of this analysis is to compare the effect of aging on different cores that have been differently affected by PV. Cores that happen to land on fast corners can run a lot faster, while having higher leakage power compared to the cores that happen to land on slow corners of the chip. Fig. 2.19 shows the percentage of speed degradation for fast and slow core throughout a 15 year period. From the graph, we observe that the two cores have a very similar profile when it comes to their speed degradation due to aging.

2. The impact of aging and PV

PV makes a homogeneous many-core system behave like a heterogeneous system. It is common practice to group the cores based on their performance. In this study, we categories the cores into 3 groups: fast, typical and slow. Fig. 2.20 shows the critical path delay of these three groups in a dashed line for 0.3 V. In the same figure, we demonstrate the impact of aging on the speed of the cores which is in the fast group. We observe that after 4 years, once-considered fast cores become slower than typical cores. It will take another 70 years before they become slower than slow cores though.
3. Efficiency of aging policies

In this section, we will evaluate the efficiency of policy A and policy B discussed previously in the Section 2.9.2 under CASE I, static and CASE II, dynamic workload environment. As mentioned earlier, based on variation in speed of cores, we can group the cores into different bins, namely, fast, typical and slow cores. For a given throughput, if not all the cores are needed to meet the required performance, a choice has to be made on which cores to select. It has been shown that the energy-efficiency of a design consisting of cores that are selected from fast group is higher than a design that selects cores from the typical or slow group [76]. Therefore, in this section we show the impact of aging only on the speed of the cores.

CASE I: Static workload environment

In a static workload environment, the system will be configured to deliver a constant throughput. For example, consider a system configured to operate at 50 MHz throughput. From Table 2.15, we know that the most optimal energy configuration to achieve 50 MHz throughput, is by using 8 cores operating at 0.3 V. If the system consists of 32 cores, policy A will use fastest 8 cores for the entire duration of time. Moreover, policy B will try to age all the cores uniformly, i.e., migrates the workload from 8 fastest cores to the next 8 in the distribution, at regular interval of time. Fig. 2.21 shows histogram of the speed distribution of the cores in the system with both the policies over various activity periods from 1 day to
20 years. The curved lines show the curve fit for the histogram. For better clarity, we have removed the bar histogram from the remaining graphs in the same figure.

At age 0, both policies will have similar distributions of cores. However, with circuit activity, aging and different policies in selecting the core, the distribution will vary. All graphs will shift on the X-axis to right, showing deterioration in speed.

We measure the deterioration in speed of the cores using three parameters, the median ($\mu$), worst case ($w$), and standard deviation ($\sigma$) for both the policies. We then evaluate the difference in these parameters between both the policies and represent it as (1) $\mu_d$, which is the difference in the median of core speeds between both the policies, (2) $w_d$, which is the difference in the speed of the slowest core (worst case), and (3) $\sigma_d$, which is the difference in the standard deviation of the core speeds between both the policies. All the three parameters for each activity period is shown in the Fig. 2.21. The extent to which the distribution shifts on the X-axis, shows a loss in performance and therefore, energy-efficiency.

**Figure 2.21:** Distribution of all the cores in the system based on speed using policy A and policy B under static workload environment.
For example, from the Fig. 2.21 after 1 day of activity, we observe that the overall speed degradation of the cores with policy B is 3% higher as compared to cores operated with policy A. Further, standard deviation (variance) in speed using policy A is less by 13.3% as compared to policy B. The difference between policy A and policy B increases with further aging. For example, from the Fig. 2.21 after 20 years of operation, we observe $\mu_d$ to increase from 3% to 14.5%, $w_d$ to increase from 3% to 18.5%, and $\sigma_d$ to increase from 13.3% to 19.4%. The results above demonstrate that the overall deterioration in speed using policy B is worse than policy A.

As seen before from Fig. 2.20 after 4 years the 8-fastest core (policy A), due to 100% activity, will now no longer be the fastest. The cores in the next region (typical) will have similar speed. One can argue if the system should migrate the workload from 8-fastest core to next 8-typical cores. This approach is similar to policy B to some extent. Nevertheless, the impact of aging on delay is not linear [44]. From Fig. 2.18, we will observe that the delay degradation due to aging between 0 to 1 year is 18%, whereas from 1 year to 2 years and 2 years to 5 years is only 2% and 7.5%, respectively. Therefore, if the system now migrates to using these never-been used cores they will deteriorate, at a much higher rate than the 8-fastest core. With that we conclude that for a static workload environment, such system should always use policy A over policy B, from aging point of view.

Additionally, for both policies, it is necessary to evaluate that the cores after deterioration is still able to meet 50 MHz throughput. Further, from the Fig. 2.21 it is clear that policy B will wear out faster than policy A, in delivering the required throughput. In such cases, the system will have to reconfigure by either using higher voltage or using more number of cores to be able to deliver 50 MHz of throughput. In other words, a system that is using policy B will lose efficiency faster as compared to a system using policy A.

**CASE II: Dynamic workload environment**

A more realistic environment for such system will be a dynamic workload in which the system can adapt to variable throughput as needed. Typically, such system will spend most of the
Figure 2.22: Distribution of all the cores in the system based on speed using policy A and policy B under dynamic workload environment.

time serving low workload with a spurt of high workload activity in between. For example, consider a system that needs to cater 50 MHz for 95% of the time and 500 MHz throughput for 5% of the total time period. From the Table 2.15, system will use [8 0.3V] and [32 0.35V] configuration to deliver these throughputs. When policy A is deployed, the system will use fastest 8 cores in the [8 0.3V] configuration. Similarly, policy B will distribute the job uniformly among all the cores at regular interval. Nevertheless, both policies will use all 32 cores when delivering 500 MHz throughput.

In Fig. 2.22 we plot histogram of the distribution of cores based on speed, using policy A and policy B for the above scenario. Similar to the static environment, $\mu_d$, $w_d$, and $\sigma_d$ for various activity periods is shown in the Fig. 2.22. We consider different activity period from 1 day to 20 years. For example, after 1 day of operation, we observe that the overall speed degradation of the cores with policy B is 1.1% higher as compared to cores operated with policy A. Further, standard deviation in speed using policy A is less by 6.8% as compared
to policy B. The difference in the speed deterioration of cores between the system using policy A and policy B increases with further aging. After 20 years of operation, we observe that $\mu_d$, $w_d$, and $\sigma_d$ increases from 1.1% (1 day) to 3.7% (20 years), 1.2% (1 day) to 8.2% (20 years), and 6.8% (1 day) to 33.3% (20 years) respectively. This demonstrates that the overall deterioration in speed using policy B is worse than policy A. Although the difference between policy A and policy B under dynamic environment is less than that seen under static environment. Nevertheless, the system using policy B under dynamic environment will wear out faster than the system using policy A.

2.10 Conclusions

In this chapter, we proposed a hybrid methodology consisting of VS and ACS methodology to gain energy efficiency. This methodology avoids use of any on-chip DC-DC converter and feedback circuits, necessary by traditional DVS technique. We studied their effects under different workload environments for various DSP core applications that favors parallelism. We showed that the proposed methodology can achieve nearly 8% (min) to 77% (max) of energy savings as compared to VS or ACS alone.

Later, we presented the impact of PV on the energy-efficiency of homogeneous many-core systems. We extended the analysis under two different workload environments, namely, static and dynamic. Various DSP core subsystems were considered in the analysis to quantify the results. We observed that to achieve highest level of energy efficiency, a system configuration will consist of, optimal number of cores, $N_{opt}$, and their optimal operating voltage, $V_{dd_{opt}}$. We also observed that the configuration, $[N_{opt} V_{dd_{opt}}]$, vary widely between a PVI design versus a PVA design.

Furthermore, we observed that in the static environment settings, a process variation aware (PVA) design is able to reduce the energy consumption of different DSP designs by an average of 29%. We also observed that in a dynamic environment, under different workload
conditions, PVA design can enjoy up to an average of 51% reduction in energy compared to a PVI design for different DSP applications.

In order to further increase the energy efficiency of the system, we proposed design corner aware (binning) methodology for selectively choosing the cores. We showed that binning methodology can achieve on an average 49.3% increase in energy efficiency as compared to the system designed without the knowledge of speed variations, with little calibration overhead.

In this chapter, we also studied the impact of aging on the speed of the cores in many-core DSP systems. We evaluated two different policies in judiciously selecting the cores based on their aging profile. We observed policy A, which selects fastest available cores, outperforms policy B, an advocate of uniform aging of all cores, under both static as well as dynamic workload environments.

The next goal beyond this study seek to evaluate the impact of PV on actual hardware. In order to achieve that we have designed a prototype chip consisting of 8 homogeneous FIR filter cores. The cores available on the chip are capable of operating from nominal voltage to near-threshold voltages. In the next chapter, we will present the design details of prototype chip fabricated in 90nm node, as well as present our findings based on measurements performed over 20 different dies.
Chapter 3

Design Details and Test Results for an ASIC Prototype

In this chapter, design details of a prototype chip fabricated on 90nm technology node and its findings are presented. The chip consists of 8 homogeneous FIR cores, which are capable of running from near-threshold to nominal voltages. In 20 chip population, we observe 7% variation in speed among the cores at nominal voltage (0.9V) and 26% at near-threshold voltage (0.55V). We also observe 54% variation in power consumption of the cores. In our proposed methodology, we leverage the benefits of voltage scaling for obtaining energy efficiency while compensating for the loss in throughput by exploiting parallelism present in various DSP designs. Based on chip measurements, we demonstrate that such a hybrid method consumes 6.27% - 28.15% less power as compared to simple dynamic voltage scaling over different workload environments. For any desired throughput, optimum number of cores and their optimum operating voltage are chosen based on the speed and power characteristics of the cores present inside the chip. Further, we will demonstrate the impact of intra-die and inter-die process variation on the system parameters. Later in the chapter, we will present analysis on energy-efficiency of such systems with changes in ambient temperature.
3.1 Motivation

The analysis presented in the Chapter 2 confirms that circuit operating at near-threshold voltage regions are greatly impacted by Process Variation (PV) effects [86]. PV affects both speed and power characteristics of the cores. Further, process variability increases with shrinking technology [92] and their effects in the near-threshold voltages are more prominent. In the presence of PV, the cores of a homogeneous many-core design can behave heterogeneously in terms of the operating speeds and power consumption. In other words, due to PV some cores operate at higher speeds while some other operate at lower speeds. Similarly, power consumption can be considerably different among different cores of a multi-core design due to the PV. Therefore, the algorithm should incorporate PV effects on each core while determining $N_{opt}$ and $Vdd_{opt}$.

In the 20 chip population, we observe 7% variation in speed among the cores at nominal voltage (0.9 V) and 26% at near-threshold voltage (0.55V). We also observe a great variation in power consumption characteristics of the cores. For example, there is 54% variation in power consumption of the cores, all running at 50MHz at near-threshold voltage (0.55V). Although the process variability is relatively low in 90nm, its impact at near-threshold operating voltages cannot be ignored.

A simple solution to bring homogeneity back to the design is to run all the cores at the speed of the slowest core in the design. This will be the case in a typical global VS system. We call such system as BASE. However, our proposed approach of gathering characteristic behavior of each core on the die after fabrication (performed only once) will help in further increasing the energy-efficiency of the system. For example, when the number of optimal cores required is less than the total cores present in the system. Proposed methodology can selectively choose the cores and the system can operate at the speed defined by the current set of cores selected rather than the slowest in the system. Nevertheless, for workload conditions which will use all the cores in the system, throughput of the system for both the approaches will still be determined by the slowest core.
Furthermore, we will show the impact of intra-die and inter-die PV effects on the optimal number of cores, $N_{opt}$ and optimal operating voltage, $V_{dd\text{opt}}$ based on our proposed PVA design. We apply the hybrid Active Core Scaling (ACS) and Voltage Scaling (VS) methodology to an 8-core FIR design, capable of catering to a wide range of desired throughputs from 100 MHz to 2.5 GHz. We observe that our proposed methodology can achieve up to 6.27% to 28.15% of energy savings as compared to systems with only VS capabilities. It should be noted that the goal of this work is not to design the fastest or more energy efficient FIR filters. Rather, the main objective of this work is to find the amount of energy savings achieved by employing a combination of ACS and VS along with careful consideration of PV effects. Several other circuit-level and architecture-level techniques that are orthogonal to our approach can be employed in addition to our methodology to further improve the energy efficiency of the design.

Besides process, integrated circuit on a chip can also be impacted by other sources of variation such as temperature, power supply noise, and current starvation [92]. In this work, we ignore power supply noise variation and current starvation, which are dynamic in nature. An increase in temperature typically causes a circuit to slow down due to reduced carrier mobility and increased interconnect resistance. However, in our case for low VDD (near-threshold voltages) the circuit is operated in temperature inversion. Here, the effect of decreasing threshold voltage with temperature exceeds the mobility degradation. Consequently, the circuit speeds up with increased temperature and vice versa. Later in this Chapter, we will also present impact of temperature variations on the core characteristics in terms of speed and power. In addition, we will also evaluate its impact on energy-efficient architectures.

This work has been done in two phases. In the first phase that was presented in the Chapter 2, an extensive simulation study of multi-core DSP designs was demonstrated and presented in terms of energy consumption, speed of operation, and number of cores etc. [72, 75]. In the second phase, which is presented in this chapter, a many-core FIR ASIC chip is designed using 90nm process to verify the findings presented in Chapter 2.
The main contributions presented here are as follows,

1. We present a scalable-throughput 8-core FIR design, which employs careful characterization, and a hybrid method of scaling the number of cores and choosing the optimum voltage in order to achieve maximum energy efficiency while taking process variation into account.

2. We provide the frequency and power measurements of an 8-core FIR filter manufactured in 90nm IBM technology, which we use as a vehicle to show the effectiveness of the above hybrid method in saving energy.

3. We present an extended analysis of process variation effect on the number of cores and operating voltage for a population of 20 chips over a wide range of throughputs [100MHz - 2.5GHz].

4. We present impact of temperature variation on energy efficient architectures.

Although our work is focused on a homogeneous many-core design with an application that can be parallelized, its benefits are not limited to such chips. In today’s designs, chips are typically a collection of several IP designs with various functionalities. Our work will enable those functions that can be parallelized (typically DSPs) to reach a higher level of energy efficiency especially in the face of process variation.

The rest of this chapter is organized as follows. Section 3.2 presents the related work. The design architecture of dynamically scalable throughput of a homogeneous multi-core system and the algorithm used to choose the minimal energy architecture is explained in detail in Section 3.3. Section 3.4 describes the implementation details of our chip and testing methodology of an 8-core FIR chip. Findings from hardware is presented in Section 3.6. Finally, in Section 3.7, we will conclude by summarizing this work.
3.2 Related Literature

In the recent past, there has been a lot of focus from IC design researchers about the energy savings in many-core DSP systems. This section provides a summary of the related work in this area.

[85] presented the very first 180mV FFT architecture that runs in the sub-threshold region. In this work, saving energy-per-operation with voltage scaling was explored. The resulting throughput of the FFT presented was very low, 164Hz at 180mV and 10MHz at 350mV. Since then there has been a wealth of research towards exploring near/sub-threshold systems and process variability [14–16,95]. In the past many prototypes have demonstrated the possibility to extend traditional voltage-scaling limit to below the threshold voltage, i.e., sub-threshold voltage region [10,85].

Extensive study has been done in the past and is ongoing, to analyze and model process variation (PV) effects, particularly on many-core designs [9,11,41,69]. Optimization considering the speed and power trade-offs employing Dynamic Voltage-Frequency Scaling (DVFS) methodology has been explored by various researchers [6, 39, 42, 48, 70]. Further, [45] recommends a technique of applying DVFS for many-core general purpose processor. Most of these work considers designs operating in the super-threshold region, where the impact of process variation is not very high. However, in the near-threshold region the process variability impacts are high and hence, cannot be ignored [86]. Therefore, in this chapter, we present the impact of intra-die and inter-die PV at near-threshold voltages on speed and power characteristics of 20 chips each containing 8 cores.

[68] demonstrated a methodology to select cores efficiently for operation in a many-core system. However, in their approach the optimization algorithm was based only on performance (speed) and excludes power or energy. For designing an energy-efficient solution, it is important to analyze the tradeoff between, performance and power. Further, [38] gave helpful insights by exploring variability under Voltage/Frequency islands.
presented a very interesting variability in Fmax profiles for an 80 core processor at 65nm operating at a minimum of 0.8V. Although in their work individual impact on Fmax and power was mentioned, the authors did not address the issue of energy efficiency gained or lost by this variability. We extend that work by analyzing impact of process variability on system parameters such as the number of cores, and operating voltages while designing an energy-efficient system. Additionally, we will evaluate temperature variability impact on such systems.

In summary, all the above mentioned work built a solid background for our contributions. Nevertheless, they either lacked the analysis considering the effects of PV on the design system at near-threshold region of operation or was incomplete in terms of different workload conditions the system can operate. Our work builds on prior work by presenting a design for a many-core DSP system that needs to cater a wide range of throughputs while achieving maximum energy-efficiency in the presence of PV. In the current system we assume that predetermined fixed and quantized voltages are available to the design. This work, however, does not include the complexity involved in generating these voltages as presented in [15].

We also present a prototype fabricated in 90nm node with 8 homogeneous cores operating at near-threshold region, to show the effectiveness of the algorithm in choosing maximum energy-efficient architecture for a desired throughput. Finally, we will present the impact of temperature variability on energy-efficient architecture.

3.3 Design Architecture and Methodology

In this section, we first discuss the system architecture details. Later, we present system parameter selection methodology for choosing the most energy-efficient architecture.

The architecture of a homogeneous many-core system is illustrated in Fig. 3.1. It consists of multiple cores along with a few operating voltages to choose from. The system architecture is built to adapt to any desired throughput in the field. This type of application is highly
Figure 3.1: Architecture of a dynamically scalable throughput homogeneous many-core systems.

desirable for environments where the requirement of the system changes over time. When the system is in operation, an event detector or a user will decide on a desired system throughput. For example, in case of an acoustic sensor, which is employed in a field to detect passing vehicles, the resolution of the filter needs to be increased when an interesting event is detected. The challenge in designing such a system is to select for a desired throughput, an optimum total number of cores ($N_{opt}$), as well as its optimum operating voltage ($V_{dd opt}$) that will be available during the deployment. Even though the number of cores ($N$) can be chosen arbitrarily, for most DSP applications, the control signals and data flow (bus connections, etc.) becomes increasingly complicated and inefficient, if $N$ is not a power of 2. Hence, in this work, we study a many-core FIR system, where the number of cores is chosen to be a power of 2, specifically 1, 2, 4 and 8.

Before deployment, one-time characterization of all cores is performed. Finite State Machine (FSM) of a typical controller is also shown in Fig. 3.1. Using an algorithm that is described in section 3.3.1, for every desired throughput that the system is intended to operate, the controller calculates $[N_{opt} V_{dd opt}]$. The configuration is then stored as a Look Up Table (LUT) in a non-volatile memory and can be read by the controller during operation. In this
work, we have assumed system to operate in a wide range of throughputs from 100 MHz to 2.5 GHz.

### 3.3.1 Design Parameter Selection

The prototype ASIC chip designed in this work consists of 8 cores. Hence, the system architecture can be built by using 1 active core (1C), 2 active cores (2C), 4 active cores (4C) or with all the 8 cores active (8C). A system with a certain number of active cores can be operated at different voltages to meet the desired throughput. An operating voltage between the cutoff voltage (circuit stops operating) and nominal voltage can be picked for the cores. The typical minimum voltage measured on the chip for FIR core was 0.4 V. Hence, we use an operating voltage range of 0.4 V to 0.9 V with a resolution of 50 mV. For a given desired throughput, the algorithm presented in this section determines the optimal parameters $[N_{opt} \ V_{dd_{opt}}]$ to achieve maximum energy efficiency. Ideally, one would expect that for a fixed throughput, $[N_{opt} \ V_{dd_{opt}}]$ tuples will remain the same from chip to chip. However, due to the presence of inter-die PV effects $[N_{opt} \ V_{dd_{opt}}]$ tuples vary from chip to chip. Later, in Section 3.6, we will see that the speed and power characteristics of the cores differ significantly. Section 3.6 will also provide experimental results of $[N_{opt} \ V_{dd_{opt}}]$ tuples for 20 fabricated many-core FIR chips.

Further, if the homogeneous cores have indeed identical core characteristics within a single chip, for 1C architectures choosing any 1 core out of 8 will not change the energy efficiency of the system. Similarly, for 2C and 4C architectures, choosing any 2 or any 4 cores out of 8 will still run at the same efficiency. However, due to the presence of intra-die PV, the cores within the same chip will differ in speed and power characteristics. Hence, while choosing $[N_{opt} \ V_{dd_{opt}}]$ tuples, it is also important to find the right set of cores ($\text{core}_{opt}$) within the 8 cores available.

Along with PV whose effects are static in nature, circuit characteristics can also be influenced by temperature. Temperature can affect two main parameters, carrier mobility and
threshold [89]. With an increase in temperature, carrier mobility as well as the threshold voltage decreases. At the circuit level, that translates to increase in delay (decrease in performance) and leakage power. However, this behavior changes with the operating voltage. The combined effect at cold (-40°C), room (25°C) and hot (85°C) temperature is shown in Fig. 3.2. At high VDD, delay has a negative temperature coefficient, i.e., it decreases with temperature. This is mainly because the impact of temperature at high VDD is dominated by mobility degradation. However, at low VDD, the delay has a positive temperature coefficient i.e., the delay decreases with increase in temperature. This is due to the fact that at low VDD, decrease in threshold voltage will have a dominant impact on the circuit characteristics. Later in the Section 3.6.6, we will analyze the impact of temperature variation on the energy-efficient configurations at voltage of interest.

In summary, for a desired throughput, system parameters of a minimum energy architecture will have \([N_{opt}, Vdd_{opt}, core_{opt}]\). These parameters once calculated can be stored in the LUT for the controller to pick during operation. In the current system, we will need 2 bits (4 options) to store \(N_{opt}\) configuration (1, 2, 4 or 8). Further, we will need 2 bits (4 options) to store \(Vdd_{opt}\) configuration and 3 bits (8 options) to \(core_{opt}\) configuration (C1, C2, C3, C4, C5, C6, C7, C8). Therefore, for every desired throughput we will need 7 bits (2+2+3) to store all \([N_{opt} Vdd_{opt} core_{opt}]\) configuration. In this work, we consider 16 possible desired throughput conditions that the system should meet, and hence, a total of \(16 \times 7 = 112\) (14
bytes) of memory will be required to store all the configurations. Area overhead of 14 bytes of memory is negligible as compared to the system area. Additionally, based on the latest flash memory available [25], the power consumption of 14 bytes of flash memory will be in nano-watt range and therefore, is negligible as compared to the power consumption of the cores. In a real design system, for example, consider a system with 256 cores, memory that is needed to store all the configurations of \([N_{opt} \ V_{dd_{opt}} \ cor_{opt}]\) for all the throughputs will be 32 bytes.

Later in Section 3.6, we will study a minimum-energy architecture system parameters based on the chip measurement results. We will also compare the energy efficiency of the system built using the proposed methodology of VS and ACS as compared to the BASE system with only VS capabilities.

### 3.4 Prototype Implementation Details

In this section, we will present implementation details of our prototype ASIC chip.

The top level chip layout, shown in Fig. 3.3 is implemented in 90nm technology using the ARM standard cell library. The chip core area is 580\(\mu m\) x 1000\(\mu m\) and overall chip die size is 1.5 \(mm^2\) including pad cells. Out of the seven metal layers, five metal layers are used for signal and clock routing and the top two layers are used for power and ground. Chip has 76 IO pins for interface and uses 80-pin QFP package for testing. The main components of the chip are summarized below.

#### 3.4.1 Clock Management

There have been three challenges in designing a clock distribution network for this chip. First, to accurately measure the power dissipation of each core, we need to ensure that at any given time only one core is active, which implies that the clock to all other inactive cores
should be turned off. We have used clock gating technique to achieve this. Second, due to limitation of 90nm technology IO library, we cannot route any off-chip signal with greater than 100 MHz frequencies into the chip. To allow testing of all cores at a frequency higher than 100 MHz, we needed a mechanism to generate high on-chip stable clock frequencies. For this purpose, we have designed and implemented on-chip Ring Oscillators (ROs). Third, to characterize each core at different throughputs, we should have a range of frequencies available on the chip. For this reason, we have designed a 4-bit programmable on-chip clock divider.

(a) Clock Generator Module: We have designed Standard cell RO (S-RO) using inverters from standard cell library. S-RO is gated using NAND cell, thereby saving dynamic power when not in use. After carefully studying simulation results and considering the desired range of frequencies, 4 S-ROs are implemented with 15, 25, 33 and 43 inverters. These are fixed frequency signals for a given operating voltage (VDD) for the chip. If the supply voltage for the chip is increased, the frequency generated out of the S-ROs will increase and vice-versa.

(b) Programmable Clock Divider: We have designed programmable clock divider using standard cells. There are four pins dedicated to program the divisor. With this, it is capable
of dividing an external off-chip clock or an on-chip S-RO clock by following values: 1, 2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 64, 96, 128, 192 and 384. Integration of this module along with the clock generator module offers the flexibility of generating a wide range of on-chip stable clock frequencies.

3.4.2 Controller Design

There are two primary controllers in this chip running in two different voltage domains.

(a) Main controller: Main Controller as shown in Fig. 3.3 is in a separate voltage domain than the cores of the chip. This is to ensure that the controller always works at nominal voltage (VDDH). The controller provides control and data interface to all the cores. Although the interface on the chip runs at very low speed, each core is capable of running at very high speed (850 MHz for FIR). Therefore, we create two clock domains in our chip: the slow one is for the interface logic and the fast one is for the cores.

(b) Datapath controller: This controller works in the same voltage domain as the cores at any given time (VDDL). For example, if the cores are operating at 0.45 V, the controller also operates at 0.45 V. This controller is responsible for assembling 32-bit data through a 2-stage inverter to interface with the core. Further, the controller also collects the 32-bit data from the core and sends it to the main controller via Level Converters (LC).

3.4.3 Level Converters (LC)

We have 32 custom designed level converters based on [93]. 32 LCs are marked in Fig. 3.3. LC is used to interface signals coming from VDDL voltage domain to VDDH domain.
3.4.4 Homogeneous Cores

We have implemented two different functionality cores, 16-bit multiplier core and 8-bit FIR filter. Multiplier core is a fully combinational design, whereas FIR core is pipelined. In the current work, we present the result on the FIR and leave the multiplier study for future work. 8-bit FIR core is implemented in a floorplan area of $100\mu m \times 100\mu m$. We create a macro block of one FIR core, after synthesis, verification, cell placement and routing. This way all 8 FIR cores will be exactly identical to each other except for the interface signals from the controller. This is the best any designer can do while designing a homogeneous many-core systems. Each core can be individually enabled using 3-bit module select pin. Each core is connected to VDDL voltage domain, which is controlled via power pins on the IO. VDDL can be changed from near-threshold (or the cutoff voltage of FIR) to nominal voltages.

3.5 Testing Setup

In this section, we will describe the testing methodology used for all the cores for the 20 packaged chips. Later, we discuss characterization overhead.

The test setup is shown in Fig. 3.4(A). It consists of a PCB board with National Instrument
(NI) VHDCI connector interface to NI PCI-6552 board [54], LabVIEW software (PC), oscilloscope to measure power, and three voltage supplies. One voltage supply is used for IOs (2.5V), one voltage supply for the main controller (VDDH) (1.2V) and the other is used for selecting the core voltage (VDDL). An ASIC chip is inserted in the ZIF socket on the board. The LabVIEW waveform editor tool is used to generate stimulus. The input stimulus is then sent to the chip using a VHDCI connector connected through the PCI-6552 board. The LabVIEW software is also used to monitor signals from the chip. Using this setup, we carry out performance and power measurements of all the cores. Later, we will also present setup to perform temperature variation analysis.

3.5.1 Functional Testing and Speed Measurements

Each core is functionally verified by testing at different operating voltages ranging from nominal voltage to its cutoff voltage. At each operating voltage, the maximum frequency at which the core can operate is measured. We use the on-chip S-RO clocks to test the functionality of each core. Later in section 3.6, we will present the range of these on-chip frequencies. Further, a similar procedure is repeated for all 8 cores. Operating voltages are changed with 50 mV steps.

3.5.2 Power Measurements

Using an oscilloscope, power traces are measured for all cores over different operating voltages. Power is measured for different input stimulus sent to the chip using LabVIEW software. In the current prototype, power gating is not implemented. Hence, leakage power of all the cores is integrated with the dynamic power measurements. Leakage power is also very negligible for 90nm node and therefore, variation in the leakage power between the cores can be ignored. The accuracy of the power measurement is determined by the accuracy of the current source meter we have used, which in this case is 10 ns.
3.5.3 Temperature Measurements

In order to perform an analysis based on changes in ambient temperature on this chip, we use ESPEC SH-241 model [1]. The test setup is shown in Fig. 3.4(B). Temperature range available with this chamber is -40°C to 150°C. However, the VHDCI connector cable that connects PCI board to the chip can be operated only in the 0°C range to 55°C range. This limited our temperature change analysis between the window 5°C to 50°C. We collect power and frequency characteristics of all the cores using the same methodology mentioned above for two different temperatures, T0 (5°C) and T1 (50°C). Later in Section 3.6.6, we will present the impact of temperature changes on the energy efficiency based on the configuration that was chosen at room temperature.

3.5.4 Characterization Overhead and Scalability

The current prototype consists of 8 cores, and therefore, one-time characterization of all the cores for 20 chips is not time consuming. To test all 8 cores using 10 different test patterns, it will take 14080 (10 test-patterns * 4 clock-cycles * 11 voltages * 8 cores * 4 clock-settings) clock cycles to test one chip. Assuming an average of 200 MHz clock speed, total calibration time will be 70 µsec. Current characterization assumes serial testing of all cores. Further optimization can be achieved by parallelizing the calibration procedure and thereby reducing the time overhead. However, this will still be a one-time procedure.

In the scenario with many numbers of cores, for example 100s of cores, individual characterization by running a functional pattern can become challenging and laborious. Further, characterizing the cores based on different ambient temperature is highly impractical. As an alternative, we can instead implement Ring Oscillators (ROs) within each chip. We can calibrate the variation in speed of the cores with the difference in the RO frequencies [56]. Further, using the EKV model we can quickly and accurately characterize a circuit for its speed, active and leakage power, from nominal voltage to the near-threshold region [61].
This will accelerate the process of one-time characterization. In this work, our goal was not to decrease the calibration time, but to evaluate the impact of PV on intra-die and inter-die homogeneous cores. As we had only 8 cores on a single chip, we went ahead with the manual calibration method. In the future, we would recommend incorporating ROs within each core for calibration purpose.

3.6 Results

This section summarizes the results and our analysis of various experiments.

We first present the range of on-chip clock frequencies available and the percentage variation of these clocks over 20 packaged chips. Later, we will present the results of \([N_{opt} Vdd_{opt}]\) tuples achieved for a range of desired throughputs for all chips. This will help us gain insight into the difference in the total number of cores picked by each chip (inter-die variation) as well as difference in which specific core \([core_{opt}]\) are picked (intra-die variation) for different throughputs.
3.6.1 Range of Frequencies Available on Chip

Four S-RO (SC0, SC1, SC2, SC3) along with 16 choices of programmable divider can generate a range of frequencies from 2.5 MHz to 980 MHz as shown in Fig. 3.5(a). SC0 is the fastest clock with 15 inverter stages, followed by SC2 with 23 stages, SC3 with 33 stages and SC4 with 43 stages. These clocks are later used to test the functionality of each core. Further, we measured S-RO frequencies across 20 chips. Due to inter-die PV effects, frequencies that are generated by S-ROs will differ from chip to chip. Variation in frequencies measured for SC2 clock for 20 chips is shown in Fig. 3.5(b). At nominal voltage we found a variation of 60 MHz (17%) as compared to the mean frequency due to PV. If we exclude the outliers then it will be 5% variation in clock frequencies across 20 chips.

3.6.2 Speed and Power Measurements

In this section, we will analyze the variation in speed and power consumption for all the cores at nominal voltage. 8 cores for all the 20 chips (160 cores) were functionally verified for multiple operating voltages until they fail. Maximum frequency of operation for each core is noted. Fig. 3.6 (a) shows distribution of maximum frequency measured at nominal
Table 3.1: Set of \([N_{opt} Vdd_{opt}]\) tuples chosen by different chips [0-19] over a range of desired throughputs [100MHz - 2.5GHz].

<table>
<thead>
<tr>
<th>CHIP ID</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100 MHz</td>
</tr>
<tr>
<td>Chip-0</td>
<td>[1 0.55V]</td>
</tr>
<tr>
<td>Chip-3</td>
<td>[1 0.40V]</td>
</tr>
<tr>
<td>Chip-5</td>
<td>[1 0.65V]</td>
</tr>
<tr>
<td>Chip-10</td>
<td>[1 0.50V]</td>
</tr>
<tr>
<td>Chip-15</td>
<td>[1 0.45V]</td>
</tr>
</tbody>
</table>

voltage for all 160 cores. At nominal voltage a standard deviation of 68.5MHz (7% compared to mean) is measured. A higher variation of up to 26% is seen in the lower voltages, which is not shown for brevity.

Similarly, we plot power consumption of all 160 cores at an operating frequency of 50MHz as shown in Fig. 3.6 (b). Standard deviation of 0.25mW (54% compared to mean) is observed. We observe similar trends for different operating voltages. This shows that in our chips, PV is more prominent in power consumption compared to speed. Hence, \([N_{opt} Vdd_{opt}]\) tuples for different throughput will differ greatly. We will present this analysis in the next section.

3.6.3 Impact of PV on choosing \([N_{opt} Vdd_{opt}]\)

In this section, we present the analysis on minimum energy architecture \([N_{opt} Vdd_{opt}]\) chosen for different chips over a wide range of desired throughputs. Set of \([N_{opt} Vdd_{opt}]\) tuples chosen by 5 different chips for a few of the throughputs is tabulated in Table. 3.1. For example, in chip number 15 (last row), the most energy efficient architecture for achieving a desired throughput of 100 MHz to 300 MHz is one core operating at 0.45 V. However, for achieving a throughput of 400MHz one core operating at 0.45 V can no longer cater, so the voltage needs to be increased to 0.55 V. For a throughput range of 500 MHz to 600 MHz, four cores operating at 0.45 V is the most energy efficient.
Figure 3.7: $N_{opt}$ chosen by different chips [0-19] over a range of desired throughputs [100 MHz - 2.5 GHz]

Figure 3.8: $V_{dd_{opt}}$ chosen by different chips [0-19] over a range of desired throughputs [100 MHz - 2.5 GHz]
In Fig. 3.7 we plot a contour diagram of $N_{opt}$ chosen by all the chips over the entire range of desired throughputs. From Fig. 3.7, we observe that the change in the energy efficient architecture, i.e. from using one core to two cores to four cores and to eight cores are not uniform across all the chips. This shows the impact of inter-die PV on the characteristics of cores inside each chip. For example, in chip number 5, one of the cores is very fast and power efficient to cater to 9 different throughputs. At the same time for chip number 15, there are many cores which are slow.

Further, Table 3.1 also shows $V_{dd_{opt}}$ chosen by different chips over few desired throughputs. We plot a surface diagram of $V_{dd_{opt}}$ chosen by all the chips in Fig. 3.8, to analyze this variation. From the figure as well as a table, we conclude that due to inter-die variation, the characteristics of each of the cores within each chip is different from another. Hence, the energy efficient architecture chosen to achieve the same desired throughput will be different for different chips. For example, from table 3.1, for achieving a desired throughput of 600MHz (column 3), chip 0 will choose 2 cores operating at 0.55V. However, chip 15 will choose 4 cores operating at 0.45V to operate with maximum energy efficiency.

Additionally, we will observe that for many chips to run the whole system energy-efficiently for the entire dynamic range, we will need only a few operating voltages that are close to the threshold voltage. From the table 3.1, chip number 10 needs four voltages of 0.5V, 0.55V, 0.6V and 0.65V. With our architecture, the system can be built with four dedicated voltages. Further analysis can be done to find the efficiency lost for the chips whose optimum voltages are more than four, but are forced to use only four choices. In the current analysis, the algorithm offers the freedom to pick any voltages with 50 mV steps.

### 3.6.4 Impact of PV on choosing $[core_{opt}]$

In this section, we present analysis on $core_{opt}$ over a range of desired throughputs for 20 chips.
Earlier seen in Fig. 3.7, for throughputs of the range 100 MHz to 2 GHz, most of the architectures use 1, 2 or 4 cores. There is a total of 8 cores available on each chip. Ideally, all the cores should have the same speed and power characteristics and hence for 1-core architectures, choosing any core out of 8 should not change the system behavior. This is similar for 2 cores and 4 core architecture as well. However, due to intra-die PV, these cores after fabrication are no longer homogeneous and are different in their characteristics.

Fig. 3.9 shows $core_{opt}$ for some of the chips for all desired throughputs. The marked circles are the cores that are active. Also shown in the figure is the operating voltage of active cores. It is important to note that in our design we are considering a common operating voltage for all cores in order to simplify the design. Fig. 3.9 presents an interesting insight into the existence of intra-die PV impact on homogeneous cores. Hence, while choosing the most energy efficient architecture, determining $core_{opt}$ is also very important. This also brings attention to the characterization process which is performed only once. Careful and rigorous characterization of each core is required to run the system with maximum energy efficiency.
The characterization process can also be initiated once in every 5 or 10 years. This will recalibrate \([N_{opt} \ Vdd_{opt} \ core_{opt}]\) incorporating aging effects.

### 3.6.5 Energy Savings

In this section, we present results of energy-savings obtained using our proposed hybrid methodology of VS and ACS as compared to the BASE system with only VS capabilities. BASE system is built using 8 cores (maximum available on the chip), whereas hybrid can choose to build the system using 1, 2, 4 or 8 cores. The BASE system uses voltage scaling technique to meet the targeted throughput along with reduced energy consumption. The hybrid methodology will optimize for \([N_{opt} \ Vdd_{opt} \ core_{opt}]\) to meet the targeted throughput. Both systems have access to same voltages with 50mV steps.
Fig. 3.10 illustrates the energy savings of hybrid systems with respect to BASE for various workload environments for 20 chip population. On average, hybrid method is 28.15%, 19% and 6.27% more energy-efficient as compared to the BASE system for low workload (10 MHz - 300 MHz), medium workload (400 MHz - 800 MHz) and high workload (900 MHz - 2.5 GHz) environments, respectively. The savings of hybrid over simple VS is more pronounced in low throughput workloads. This is mainly because hybrid can choose to use 1, 2, 4 or all 8 cores depending on the voltages available and throughput needed. For low workload environments, optimum number of cores required is less than the maximum available on the chip. Therefore, we see higher energy savings of hybrid systems for low or medium workload environments as compared to BASE. However, for higher workload environments, both hybrid and BASE might end up using all the 8 cores available on chip to meet the throughput requirement. Hence, energy savings of hybrid as compared to BASE is comparatively low. In application cases where the workload varies and is low for extended periods of time, hybrid systems will be highly beneficial as compared to BASE with only VS capabilities. The energy savings observed from the chip measurement results reconfirms our results from the simulation environment that was presented in the Table 2.10.

3.6.6 Impact of Temperature Variation

In this section, we present our analysis on the impact of temperature change over the characteristics of the cores as well as study on the energy-efficiency of the system to change in temperature. Based on the setup that was summarized in Section 3.5.3, we measure total power for all cores. Later, to study the distribution of the cores and impact of temperature change, we plot the median (μ) and standard deviation (σ) of average total power for all cores in Chip 3, measured at different temperatures, Cold (T0: 5°C), Room (23°C) and hot (T1: 50°C). Fig. 3.11 shows the distribution. From the figure, we observe that with the increase in temperature total power decreases for cores and therefore, the distribution shifts to the
left. Similarly, with decrease in temperature, leakage power increases and hence total power increases, which makes the distribution w.r.t room temperature shift to the right. The observation made based on our chip result due to changes in temperature is similar to the results presented in [15]. In the current setup the difference between cold and room temperature is 18°C, whereas the difference between hot and room temperature is 27°C.

We also evaluate the impact of change in temperature for cores within the same die. For example, Fig. 3.12 (A) shows variation in total power for all cores in Chip 3 measured at 0.65 V under two temperature settings T0 and T1. From the figure, we will observe that the variation in power is different for different cores. Let us first study variation under T0 temperature w.r.t room temperature. For example, Core 0 (C0) shows 2% change in total power consumed, whereas Core 7 (C7) shows 4.8% change. Similarly, variation in power under T1 temperature is also different for different cores. For example, compared to their power consumption at room temperature C0 consumes 6.7% less power, whereas C7 consumes 3.9% less power. Although variation in temperature impact cores differently, the difference between them is quite small at around 1% - 3%.

To study the impact of temperature variation from die-to-die, we measured total power consumed for chip 5 under the same settings as above, i.e. 50 MHz frequency and operating voltage of 0.65 V. The graph is shown in Fig. 3.12 (B). From the figure, we can see that the impact is different from chip to chip. For example, let us consider the case of decrease in temperature (T0). We can see C3, which in Chip 3 underwent 4.2% change, in chip 5

**Figure 3.11:** Distribution of cores based on total power consumed for Chip 3 measured at 50 MHz frequency and an operating voltage of 0.45 V.
undergoes only 1.7% change. Similarly for an increase in temperature (T1) case, we can see C7, which shows -3.9% change in Chip 3 undergoes -6.3% change in Chip 5.

Increase in temperature will also lead to decrease in frequency. Similarly, a decrease in temperature will lead to increase in frequency. Our current setup to measure maximum frequency achievable for the core was summarized in Section 3.6.1. As the frequency of the core is high, we are limited to use on-chip RO clocks and programmable divider and therefore, the frequencies that are available on the chip are discrete as shown in Fig. 3.5 (A). The smallest change in frequency that is available is 10% (40 MHz). In our current setup of 18°C change (T0) as well as 27°C change (T1) we didn’t observe change in frequencies. It is possible that the change was less than 10% that is difficult to capture with the current setup constraint. Nevertheless, the cores operate at the frequency determined at room temperature setting.

We also study the effect of temperature on the energy efficiency of the system at various throughputs. There will be energy-efficiency lost or gained. In order to do this, for every targeted throughput and derived optimal configuration of \([N_{opt}, V_{dd_{opt}}, \text{core}_{opt}]\), we calculate the energy consumed by the system at T0 (5°C) and T1 (50°C). The difference in energy consumption compared to room temperature for two different chips, chip 3 and chip 5 are shown in the Fig. 3.13. We evaluate the energy difference under 3 workload environments; low workload (10 MHz - 300 MHz), medium workload (400 MHz - 800 MHz) and high
workload (900 MHz - 2.5 GHz).

From the figure, we will see at T0 (cold), over different workload environment, total energy consumed will be more as compared to room temperature. This illustrates that if the system is built with the configuration that is obtained using room temperature settings, system will incur 4.42%, 3.68% and 4.47% increase in total energy as compared to room temperature under different workload environment. These differences are slightly smaller for Chip 5. Similarly, at T1 (hot), the system will gain energy efficiency, because the cores will have less leakage power due to increase in temperature. Therefore, from Fig. 3.13, we observe 4.2%, 4.6% and 3.4% less energy for chip 3 under different workload environments.

In the above case, the energy efficiency lost (4%) as the temperature decreases is not significant. However, as the temperature further decreases, energy efficiency will further deteriorate. At the same time, the cores will be faster than those measured at room temperature. Further study can be done if the energy efficiency can be regained by reconfiguring. For example, if the room temperature configuration suggested [4 0.65V], an analysis on if energy efficiency can be gained by either using fewer cores or operating at lower voltages. In other words, the system should reconfigure in order to run at an optimal energy.

Similarly, when the temperature increases (T1), energy efficiency are gained, i.e. cores dissipate less power. However, as temperature further increases, frequency of core will decrease. Beyond a certain threshold, the core will no longer be able to meet the throughput. For example, if the room temperature configuration suggested [4 0.65V], beyond certain temperature 4 cores operating at 0.65 V will no longer be able to meet the throughput due to increase in temperature. In that case, either number of cores should be increased or the operating voltage. In other words, the system should reconfigure in order to meet the required throughput. Besides, one will have to evaluate the overhead in saving system configuration at different temperature, characterizing it as well as a mechanism to determine the change in temperature. We will leave this study for our future work.
Figure 3.13: Difference in total energy consumed at T0 and T1 temperature w.r.t room temperature, for Chip 3 and Chip 5 under low workload (10MHz-300MHz), medium workload (400MHz-800MHz) and high workload (900MHz-2.5GHz) environment.

3.7 Conclusions

In this chapter, we verified the effectiveness of an energy-saving method, where the number of cores and their operating voltage are pre-selected based on process variation effects and desired throughputs during a characterization phase. The verification is done using an 8-core FIR design fabricated in 90nm IBM process. Up to 7% and 26% variation in frequency is observed among the cores at 0.9 V and 0.55 V, respectively. It is observed that different chips require a different number of cores and different voltages to meet the same throughput due to process variation.

Although the process variability is relatively low in 90nm, its impact at near-threshold operating voltages cannot be ignored. It offers a good platform to validate our idea and methodology in building process variation aware energy-efficient architectures. Further, up to 28% energy savings are observed compared to a BASE system, where voltage scaling is used to meet different throughput requirements. There will be some energy efficiency lost in the off-chip DC-DC conversion for obtained the required operating voltages which we haven’t considered in our analysis. However, this will be common across both the methodologies of VS, and our proposed methodology.

We also presented the impact of temperature variations on the core characteristics in terms of
speed and power. Based on the measurements from a chip at different ambient temperatures, we evaluated temperature variation impact on energy-efficient architecture configuration. If the system is configured based on the room temperature settings, we observed that with an 18° change in temperature, the system will lose about 4% of its energy-efficiency.

While testing the prototype chip, many shortcomings in the design were noted. For example, manual calibration of cores by running patterns that were used, is not a scalable approach. Measuring the frequency and power of each core at different operating voltages and for various temperature settings, can be a laborious task. An investigation into using circuits such as a ring oscillator (RO) within each core that can be representative of the core’s speed and power is necessary. Further, a study to calibrate the impact of PV on RO that can be representative of the impact of PV on the core is also needed.

In the current measurement setup, National Instrument VHDCI cable was used in order to send the data from the PC to the PCB board [54]. This cable can be used in the temperature range of 0°C to 55°C. An adequate interface is necessary in order to enable experiments for a wide range of temperature settings. We can take all of the above learnings to improve the design in order to build the next prototype in future.

In the next chapter, we turn our attention to another form of variation in the design, namely voltage variations. If the voltage variation exceeds the noise margin of the design, it can cause computational errors. Due to their dynamic nature, these faults are relatively harder to detect at the time they appear and therefore, addressing them at runtime is necessary.
Chapter 4

Voltage Variation Aware Energy-Efficient Designs

In this chapter, we continue our analysis on system variations by studying voltage noise fluctuations. Voltage fluctuation in the power distribution network due to fast changes in the supply current (di/dt) can give rise to dynamic voltage noise. If the voltage noise exceeds the noise margin of the design, it can cause computational errors. Further, transient changes in the power supply not only impact reliability and performance, but also threaten system security. In this chapter, we first quantify the magnitude and the timescale of voltage transients as measured on a 28nm FPGA. We demonstrate that the extreme activity in the fabric can cause enormous drops of over 30%, which is 10 times larger than what is typically allowed. Second, we give insights into intra-die voltage fluctuations that are generally difficult to detect. Third, we propose a variable-resolution on-chip digital sensor that can measure high speed, voltage transients, which are shorter than 4ns. Finally, we propose an Emergency Monitoring System (EMS) that along with the sensor can be used to reduce timing margins, in order to gain area and power efficiency. The EMS monitors and measures the magnitude of voltage noise in the design and generates an alert for any potential failure that can occur in the design due to voltage fluctuations.
Figure 4.1: (A) Voltage noise source in a many-core environment sharing the same PDN. (B) Voltage noise characteristics as measured on 28nm technology node.

4.1 Motivation and Background

The recent electronics trend is towards many cores SOC, by packing more general purpose processor (cores) onto a single die to gain performance benefits. However, energy efficiency has emerged as an even more important design goal. To gain energy efficiency, designers depend heavily on architectural power-saving techniques such as clock and power supply gating. Further, with many-core processors under a common power grid becoming the norm, any sudden increase in the switching activity of a particular block requires additional current to be supplied to that area of the chip [66]. The parasitic inductance in the power grid network as well as the chip package causes a di/dt drop and hence, a transient voltage drop in the Power Distribution Network (PDN). Turning on and off (gating) of the high number of gates, which is also equivalent to charging and discharging of capacitances can cause voltage fluctuations in the PDN. One such example is shown in Fig. 4.1 (A), where power gating of one of the cores, creates a transient voltage fluctuation experienced by neighboring cores. We refer to these voltage fluctuations in the power supply as voltage noise. Since the fluctuation remains for a very short time and depend on the activity in the cores, they are also referred as dynamic voltage noise.

Voltage noise characteristics measured at 28nm technology node on an FPGA [98] is shown
in Fig. 4.1 (B). We define the supply voltage noise in terms of number of undershoots and overshoots, their widths, and decaying factor. For example, in Fig. 4.1 (B) we observe four undershoots and four overshoots, each 5 ns in width, and the total fluctuation settles down in 40 ns. The shape and the magnitude of voltage noise is a complex function of on-chip and on-board decoupling capacitors, their locations, transistor technology node, packaging material, and packaging capacitances. Therefore, the characteristics of voltage noise will vary across different chips and boards.

Undershoots in voltage noise characteristics will cause a momentary decrease in the supply voltage. Consequently, the gates that experience undershoot in the supply will observe momentary slowdown in their switching speed. Similarly, overshoots can cause gates to speed up as the voltage seen by the gate is more than the nominal supply voltage. As voltage noise is local to a region, only the delay paths (slow or fast) in the vicinity of that region will be affected [31]. Collectively, it can create dynamic, non-permanent faults within the design. Due to their dynamic nature, these faults are relatively harder to detect at the time they appear and therefore, addressing them at runtime is necessary.

We ran a small experiment in order to analyze the impact of voltage noise on the characteristics of a design. Using HSPICE simulation, we induced voltage noise in the power supply on a variety of circuits. We choose the shape and magnitude of the noise based on experimental results provided in [98]. Specifically, we used dampen SINE wave function for supply (VDD) to obtain a signal with the same characteristics as shown in Fig. 4.1 (B). Power supply at nominal voltage is 1V and we choose ~15% voltage noise, which is one of the observed amounts in [98]. We then observe its impact on both the slowest as well as the fastest paths for a variety of designs.

Slowest paths in the design determines the maximum speed at which the circuit can run reliably. Meanwhile, the fastest path determines the minimum hold time necessary for the design to operate. If the fast paths become faster due to any variation, it can result in a total functional failure. Table 4.1 summarizes our results. Typically, with undershoot in the
Table 4.1: Impact of voltage noise on timing paths with 15% voltage noise in the supply.

<table>
<thead>
<tr>
<th>Circuits</th>
<th>Slowdown of the slowest path</th>
<th>Speedup of the fastest path</th>
</tr>
</thead>
<tbody>
<tr>
<td>32-bit adder</td>
<td>21.4%</td>
<td>4.0%</td>
</tr>
<tr>
<td>32-bit multiplier</td>
<td>22.1%</td>
<td>14.3%</td>
</tr>
<tr>
<td>32-bit divider</td>
<td>20.7%</td>
<td>4.3%</td>
</tr>
<tr>
<td>FFT</td>
<td>23.5%</td>
<td>11.0%</td>
</tr>
<tr>
<td>DWT</td>
<td>17.0%</td>
<td>3.0%</td>
</tr>
<tr>
<td>iDCT</td>
<td>14.8%</td>
<td>4.0%</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>19.9%</strong></td>
<td><strong>9.5%</strong></td>
</tr>
</tbody>
</table>

power supply, we observed a slowdown in the critical path delay and similarly, with every overshoot in the supply we observed a speedup. From Table 4.1, we observe on an average a 19.9% slowdown and a 9.5% speedup for various designs. The design will experience such a dynamic slowdown or speedup while it is running. If ignored, these dynamic slowdowns and speedups can lead to dynamic faults in the system.

With aggressive scaling, shrinking device sizes, increasing transistor density, and increasing System-on-Chip (SOC) complexity, dynamic variations in the design are increasing too. These variations in device and circuit parameters degrade the performance and energy efficiency of microprocessors. The traditional approach for the processors to tolerate these variations is to use guard bands or timing margins, trading lower power efficiency or performance for operational robustness. However, as the industry moves toward smaller device feature sizes, the safety timing margins must grow, as circuit behavior susceptibility to variation increases at reduced feature sizes [57, 63]. This further decreases the power efficiency. Hence, an efficient solution that can shave off some of these timing margins is desirable.

To summarize, there is a need to design a system that can dynamically predict or measure the variations at runtime and alarm the system to avoid any potential functional errors. Additionally, one should also analyze the complexity involved in designing such a system, such as area, performance and power trade-off with the ability to measure the variation
accurately. Further, with the transient times being an the order of picoseconds, any proposed solution needs to have a very fast response time. In the work presented in this chapter, we target the aforementioned problem.

In summary, the main contributions presented in this Chapter are:

1. A novel demonstration of extreme voltage undershoot and overshoot caused by fabric activity on a 28\textit{nm} FPGA.

2. Initial findings on the presence of intra-die voltage variations that can go undetected using a static single sensor.

3. Design and implementation of an on-chip, fast, and accurate digital sensor capable of sensing and measuring voltage fluctuations in PDNs.

4. Tradeoff analysis of sensor resolution, its speed, and area overheads.

5. Development of an Emergency Monitoring System that is capable of alarming the functional units against short-term voltage fluctuations. Such a system is employed to shave-off safety timing margins in order to gain area and power savings.

The rest of this chapter is organized as follows: In Section 4.2 we will summarize the related work. In Section 4.3, we will review the experimental setup to generate voltage transients in FPGA fabric and study the architectural details of the proposed sensor. Further, in Section 4.4, we will discuss the tradeoff analysis and overheads associated with the design. Later, in Section 4.5, we will present insights into the presence of intra-die voltage variation in a design and in Section 4.6 we will present an application in which the proposed sensor system can be used. Finally, in Section 4.7 we will conclude by summarizing this work.
4.2 Related Literature

Voltage sensing based on a Ring Oscillator (RO) has been explored by many researchers [8, 97]. However, in order to achieve a good resolution for RO based sensing, it needs to run for many cycles and hence, has a very slow response time. Multiple combined oscillators were also explored to improve the resolution to 125 ns [51]. Nevertheless, it is still far too slow in detecting and reacting to nanosecond scales of events that exists in 32nm node and below. Thus, they are better suited for sensing static effects or slow transients. Some platforms provide a single on-chip sensor that can estimate the voltage at a package pin.

A variation-resilient circuit design technique is presented in [43] for maintaining parametric yield. It utilizes an on-chip phase locked loop (PLL) as a sensor to detect Process, Voltage and Temperature (PVT) variations. Moreover, the complexity of integrating such custom circuits with functional units, will add area overhead in terms of custom circuitry as well as custom routing resources necessary for such components. Many state of the art ICs can sense voltage at various locations on the die, and at speeds, high enough to detect transient effects of the range of 10 ns [49, 59]. However, with shrinking technology node and reducing decoupling capacitor size, the fluctuations are high-frequency (<4 ns). Hence, faster response time circuit is needed to measure these fluctuations with a high resolution.

Abundant research work at architecture as well as circuit level is currently on-going towards building an on-chip fluctuation monitors that can dynamically detect and correct timing violations [12, 13, 19, 23, 29, 33, 60, 65]. For example, the Critical Path Monitor (CPM) by IBM [49] uses Time Division Converter (TDC) circuit to monitor delay variation of paths, which are replicas of critical paths. However, typically all designs are optimally pipelined and has many equal sized (number of combinational gates) critical paths. Due to process variation effects, there are high chances that the critical path identified before is no longer a critical path after fabrication. In cases where the new critical path is away from the sensor location, the circuit can run into localized voltage emergencies [31] without the sensor detecting it. Hence, it is very important to design a sensor independent of the critical path.
with variable resolutions in measuring the localized variations. This methodology may need tuning to match the real circuit behavior. Further, resolution achievable by the Critical Path Monitor (CPM) methodology is fixed, 17 mV/bit, operating at 48 MHz. In contrast, our approach can have variable resolution sensing component of 20 mV/bit, 10 mV/bit or below.

The Razor system [2,23,26] proposed replacing critical flipflops with in-situ latches that can help reduce the voltage to achieve energy efficiency. Any signal that arrives shortly after the clock edge is flagged as an error. The problem with such a technique is that it introduces significant hold time constraints that are difficult to meet and also increases the risk of race failure. Additionally, the added complexity in designing such accurate, in-situ latches makes the implementation process much more complex. Small paths in the design can sometimes trigger false alarm and hence, balancing them by adding extra buffers is necessary, thereby, adding to the overall area overhead [3]. Although a Razor design could be justified for a voltage scaling scenario, it is too complicated and expensive to implement for occasional dynamic voltage variations. The main advantage of our methodology as compared to Razor system is in the overall design time and verification effort. Our sensors can be independently designed as well as verified, and then can easily be integrated with the rest of the design. In addition, razor systems suffer from relatively higher performance and area overhead. For example, Bubble Razor adds 1% - 3% performance penalty to the normal operation of the functional unit due to the insertion of special flops, while our approach has no performance overhead. Bubble razor also adds 8% - 20% (error recovery) area overhead for Cortex-M3 implementation [26].

In this work, we propose a much simpler design of a fully-digital on-chip process variation aware voltage sensor, which can be used to estimate the extent of variations and apply the right amount of corrective actions to avoid any potential failures. Our technique does not involve modifying the datapaths of Design-Under-Test (DUT). Our total area overhead along with the controller is around 2.46%.
4.3 Design of On-chip Voltage Sensor System

In this section, we will discuss the architectural details of our proposed fully digital on-chip voltage sensor system. The goal of this system is to measure voltage noise and detect voltage emergencies. Further, in this section we will also describe our methodology to generate voltage transients, and discuss architectural details of each component of the voltage sensor system.

Fig. 4.2 shows a high level diagram of our proposed system. It consists of a voltage sensor and an Emergency Monitoring System (EMS). Our proposed voltage sensor consists of two main components, namely sensing component and digitizing component. All the components are connected to the same power supply as the DUT. Given that all these components are digital, the system can be easily integrated with other design units and placed close to them. Close placement with the circuit responsible for generating the voltage transients will further help in increasing the accuracy of noise measurement. The sensor measures the voltage value on the PDN whereas EMS determines if the design has run into emergencies and generates an alarm.

Generating Voltage Transient:

In order to test the power grid and methods of sensing, a technique for generating voltage transients is needed. In this section, we conduct experiments that result in significant voltage
transients and characterize these changes.

In the past, flip-flops were used as shift registers to generate transients [99]. However, logic blocks are spread sparsely throughout a fabric, and in our testing it was not very effective in creating larger voltage transients. We propose a novel method of generating transients: simultaneous switching of dense interconnect resources. Unused wires and programmable interconnect points (PIPs) represent an enormous amount of capacitance that can be charged and discharged. After placement and routing of a design, a signal can be connected to a large number of unused PIPs, for instance with assistance from the TORC tools [79]. Interconnect has been used in the past for unconventional purposes [81], but to our knowledge never for inducing transients. This approach not only enables the generation of transients, it also provides some insight into an interesting related question: can change in on-chip activity cause out-of-spec voltage excursions? The answer is yes, and to a surprising extent.

The flow is represented in Fig. 4.3(a). After placement and routing of a design, we connect the signal net to a large number of unused PIPs, with assistance from the TORC tools [79].
The *blockfill* tool takes in the initial coordinates, \( [X_i, Y_i] \) and the final coordinates, \( [X_n, Y_n] \) and connects all the unused pips within this co-ordinate to the source net as shown in Fig. 4.3(a). Now, whenever the source net is toggled, all the *PIPS* connected to it are switched simultaneously. We control the switching of the source net through MicroBlaze which can be programmed using a C program in real time.

We simultaneously switched about half of the PIPs (5M out of 10M) on a Xilinx Kintex-7 FPGA residing in a KC705 board, and measured the response on a package supply pin with an oscilloscope. A trace is provided in Fig. 4.3(b). The events caused a 31% undershoot at the pins, more than ten times the allowed 3% fluctuation. Detection of soft errors was outside the scope of this work, but clearly a 31% undershoot can cause timing errors in aggressive designs. The minimum voltage observed was 688 mV, below the specified minimum data retention voltage. Just as surprising was the amount of overshoot, reaching 14% above nominal. In general, we found that the amount of voltage swing was nearly linear with the number of PIPs switched.

In our experiments we use the same voltage noise waveform to design and calibrate the sensor. The magnitude of undershoot and overshoot voltage is directly proportional to the total number of gates (capacitances) switching simultaneously. Therefore, to cover a wide variety of switching scenarios, we use different magnitude of voltage noise, ranging from 5% to 15% fluctuation with respect to the nominal voltage of 1V. Using HSPICE, we model the power supply noise behavior. Specifically, we use dampening SINE wave function for supply (VDD) to obtain a signal with the same characteristics as shown in Fig. 4.3 (b).

### 4.3.1 Design of Sensing Component

In this section, we will present the architectural details of the sensing component of the sensor. We propose a high-speed voltage transient sensor using digital logic gates, which is easily available in any generic standard cell library. Since the circuit is all digital, the sensor can sit within the design very close to the functional unit generating transient and share the
same PDN.

The architecture of the sensing component is shown in Fig. 4.4. Transistor delay and gate delay are sensitive to the supply voltage. Now, whenever there is undershoot or overshoot in the supply voltage the transistor delay will increase or decrease, respectively. We exploit this behavior in sensing time-dependent voltage noise. We first start with analyzing the delay changes due to voltage noise for different gates. Our goal is to find a gate that is most sensitive to voltage variations.

Delay line can be implemented using multiple approaches such as a chain of inverters or buffers [49]. In our experiments, we first consider all potential gates such as NAND, NOR, BUF, INV, XOR, FLOP and LATCH, which are typically available in all standard cell libraries, that can be used to implement a delay line. We analyze their delay with respect to both 'rising' and 'falling'. Fig. 4.5 shows delay changes of some of these gates (S-BUF, INV, BUF, MUX) with 15% voltage noise. S-BUF is a special buffer, also called as DELAY cells in most available standard cell libraries. These cells are generally used to fix the hold time issues in the design. From the figure, we see that S-BUF is equally sensitive to both 'rising' as well as 'falling', whereas a regular buffer is more sensitive to falling edge and less sensitive to rising edge. In the absence of such a gate in the library, the most sensitive gate should be chosen. Although, sensitivity of the gates can be increased further by using custom cells and applying transistor sizing techniques, we avoid using any such tweaks in order to streamline the process of design and verification of the digital system.

**Figure 4.4:** Architecture of sensing component.
Figure 4.5: Rise (left) and Fall (right) delay for different cells in the delay line component for 32nm library.

Figure 4.6: Delay sensitivity of SBUF based delay element for different technology nodes.

Instead, to further increase the resolution of the sensing component, we cascade the cells together as shown in Fig. 4.4. Individual cells of the sensing component need to be chosen such that the delay sensitivity increases with each cascading stages. Nevertheless, each cascading stage will add its own unit gate delay and therefore, will slow down the total response time of the sensor. Further, low stage counts will yield less resolution, but faster response time. Hence, the total number of stages of the sensing component should to be carefully chosen. Later in the Section 4.4.2, we will present trade-offs between cascading stages of the sensing component, achievable resolution of the sensor and total length (area overhead) of the digitizing component.
The result in Fig. 4.5 corresponds to gates in a 32\textit{nm} technology node. With technology scaling the gate delay changes and hence, it might impact delay sensitivity of the sensor. Therefore, we perform further study on delay sensitivity of the sensing component for different technology nodes. From our experiments, we observe that the sensitivity of all the gates increases with shrink in the technology node. In Fig. 4.6, we plot delay sensitivity of SBUF with 15\% variation in supply voltage for 90\textit{nm}, 45\textit{nm}, 32\textit{nm} and 28\textit{nm} nodes. We observe delay sensitivity to increase from 14.8\% in 90\textit{nm} technology to 25.5\% in 28\textit{nm} technology.

4.3.2 Design of Digitizing Component

In electronics Time-to-Digital Converters (TDCs) \cite{37} or time digitizers are devices commonly used to measure a time interval and convert it into digital (binary) output. TDC circuits are widely used and extensively tested in mitigating soft errors in integrated circuits for various applications \cite{53}. However, in our scope of work we explore how TDC circuits specifically can be used in detecting dynamic voltage fluctuations. We use (TDC) circuit to digitize small fluctuations measured by the sensing component. Fig. 4.7 shows the architecture of the digitizing component. An input pulse zooms through the latch chain, racing...
against a clock. The bit position of the transition between 1s and 0s in the latch output provides an indication of the timing of the original event.

For example, in the Fig. 4.7, we show three such input pulses, nominal, late and early. Once the sensor is calibrated, nominal pulse will produce a binary code at the output of all the latches and that becomes the base reference (000111 in the figure). A voltage droop in the PDN will increase the propagation delay of the pulse within the sensing component. In this specific case, the sensor will read 011111. Similarly for an overshoot when the propagation delay reduces, the sensor will read a lower number, in this case, 000001. The sensor needs to be reset before reading the next value. We use level-sensitive latches instead of the conventional flip-flops. This allows a reset and a sample to occur in the same clock phase.

In Fig. 4.8, we show details of the timing diagram of the important signals within the sensor. The sensor, when implemented in 32nm technology node, can operate at a maximum TDC clock speed of 1.5 ns. In other words, we can sample TDC output at 667 Mega Sample Per Second (MSPS). Since two samples should be observed before a voltage change is sensed, this sensor can detect fluctuations of <4 ns or lower. TDC is implemented close to the sensing component and hence, it will also be affected by the voltage noise on its power supply. However, the setup time of the latch that is used to implement a TDC is in the order of picoseconds. Therefore, the readings from TDC can still accurately capture sensor output.

4.3.3 Calibration of the Sensor

In this section, we will present the technique used to calibrate the sensor measurements. The input pulse to the sensing component is a periodic signal assumed to be coming from an on-chip clock generator. During operation, we can dynamically choose the offset of the input pulse with the latch clock signal. This will change the arrival time of the pulse at the input of the digitizing component. For example, consider a TDC we want to calibrate at the 3rd bit position. As shown in the Fig. 4.8, once the specification of Delay Element (DE) is fixed, the arrival time at DE input will determine how many TDC flops will latch 0 and
how many of them will latch 1. The arrival time is set such that the transition of 0 to 1 is at 00111111. The '0' and '1' is not balanced exactly at midpoint (4-bit). From the voltage fluctuation characteristics Fig. 4.1 (B), we observe that the undershoots tend to be larger than the overshoots. Hence, while calibrating the sensor we align the transition 1-0 such that there is extra room for an undershoot (00001111).

Each sensor on the chip will undergo a calibration process. Further, the calibration can be customized for a specific design or an application. For example, if the designs are more prone to overshoot (fast paths), the nominal sensor readings can be adjusted to create more room for an overshoot.

Process variation effects can impact the sensor delay (net delay as well as gate delay) and this will introduce variations in the pulse arrival at TDC flop inputs. Therefore, each sensor sitting in a different location on the die needs individual calibration. The calibration process, i.e. adjusting the arrival time of input pulse at Delay Element (DE), helps us in maintaining the TDC output at 00111111 at nominal voltage. Since the calibration (selecting the offset) is done post silicon at the time of deployment, sensor readings are not affected by process variation.

**Figure 4.8:** Timing diagram of all the input/output signals of the sensor.
### Table 4.2: Simulation setup

<table>
<thead>
<tr>
<th>Library Model and Technology</th>
<th>PTM Model 32nm</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDL Simulation</td>
<td>Verilog, Synopsys VCS</td>
</tr>
<tr>
<td>Voltage sensor design</td>
<td>Synopsys HSPICE, NanoSim</td>
</tr>
<tr>
<td>Power Simulation</td>
<td>Synopsys Prime Power PX</td>
</tr>
</tbody>
</table>

#### 4.3.4 Design of Emergency Monitoring System (EMS)

We have designed an Emergency Monitoring System (EMS) around the sensor which will determine and generate alarm if the system runs into voltage emergencies. Once the sensor is calibrated and the resolution of the sensor is fixed, EMS stores the location of the edge $0 \rightarrow 1$ for nominal supply conditions. Each time the sensor reads a value, the system will compare the relative position of the edge $0 \rightarrow 1$. If the displacement is more than a preset threshold, it will raise an alarm.

For example, consider a 4-bit TDC and 20.17 mV/bit resolution sensing component. If the nominal reads 0001 and the system is designed to raise an alarm if the voltage noise exceeds 40 mV (4%), the system will raise an alarm for sensor readings of 0111 or 1111. Voltage emergency threshold in the EMS system can also be configured dynamically based on the application or design.

#### 4.4 Sensor Characteristics and Overhead Analysis

In this section, we will discuss the methodology used for performing the simulations as well as analyze the trade-off and overhead in designing the voltage sensor system.
4.4.1 Methodology

Table 4.2 gives the summary of the tools and models used for all simulations. Sensor system as well as all the functional units is implemented using 32nm PTM technology [5]. Impact of voltage noise is then observed over the functional units and the sensor system.

4.4.2 Resolution Trade-off

The resolution of a sensor is defined as minimum voltage variation measured for every bit reading (mV/bit). Fig. 4.9 shows resolution achieved with different cascaded stages of S-BUF in the sensing component of the sensor to measure a voltage drop of 242 mV (~ 25% voltage noise). As mentioned earlier in section 4.3.1, we will observe that the resolution of the sensor increases with increasing stages. However, every increase in the resolution comes with a price of TDC area overhead, i.e. minimum sensor bits necessary to measure same magnitude of voltage noise.

For example, from Fig. 4.9 we observe that for measuring a change of 242 mV with a 3-stage sensing component and a resolution of 14 mV/bit, we would need a TDC of 18-bit length. However, for measuring the same amount of voltage variation with 4-stages in the sensing component and a resolution of 10 mV/bit, we would need 24-bit TDC; an increase of 6 bits. Additionally, every stage in the sensing component will further slow down the response time.
of the sensor. Hence, there is a clear trade-off between the resolution of the sensor, TDC length necessary, and the total response time of the sensor. For further analysis we will use two different resolution sensors, namely 20.17 mV/bit and 10.08 mV/bit.

From Fig. 4.10, we observe that the sensor voltage measurements are linear with the PDN voltage noise. We plot the readings of two different resolution sensors, 10.8 mV/bit and 20.17 mV/bit. Both these sensors measure voltage noise with a very small error percentage of 0.03% and 0.01% respectively.
4.4.3 Implementation Overhead

In this section we will discuss the overheads incurred in terms of area and power for implementing the voltage sensor system.

Table 4.3 summarizes this information for an FFT design. Each row in the table begins with the maximum supply voltage noise that the sensor and the EMS system can detect. The next column shows the corresponding slowdown in the critical path due to voltage fluctuation. For example, in the third row, it means that without the sensor system, the "safe" clock has to be 16.6\% slower than the frequency at nominal voltage. The next two columns provide the information about the sensor characteristics in terms of number of bits required to measure the voltage noise and area overhead associated with the sensor. Area and power overhead numbers are represented in percentage with respect to the total design area. The area overhead increases as the maximum voltage noise to be measured increases. The next column represents an area overhead in implementing an EMS for the sensor. We then compare the total overhead of the system, which includes sensor (detecting) and EMS ( alarming).

Table 4.3 shows the overhead of sensor with two different resolutions, 20.17 mV/bit and 10.08 mV/bit. As the resolution as well as magnitude of voltage fluctuation increases, TDC length increases. Therefore, the area and power overhead of both the sensor as well as EMS will also increase. However, from Table 4.3, even with \sim 15\% voltage fluctuation and \sim 30\% performance degradation (timing margin), the area overhead of the total system is only 2.5\%. For all the measurements we observed a maximum power overhead of only 0.1\% with respect to the design.

Generating a periodic signal to the sensitizing component can also be considered as an overhead in the design. If not already available, it can be generated by implementing ring oscillator based clock, which has a relatively low area overhead [30]. However, in the current study it is assumed to be available on chip.
Figure 4.11: (a) Setup for measuring intra-die voltage variations. (b) Sensor readings from UART. (c) Placement of the two sensors and switching fabric in the FPGA.

4.5 Insights into Intra-die Voltage Variation

Some platforms provide a single on-chip sensor that can estimate the voltage at a package pin. The single sensor approach is highly inefficient in analyzing the overall magnitude of voltage variations happening in other corners of the chip. Pinpointing the location and timing of droops with on-line sensing could potentially enable improved designs. We carried out an experiment to gain insights into detecting intra-die voltage variation using our proposed sensor.

4.5.1 Experimental Setup

On the FPGA fabric, we implement two sensors with their delay-lines. We individually calibrate the delay-line of each sensor. Entire system architecture which is shown in Fig. 4.11(a), is assembled using an Embedded Development Kit (EDK) tool by Xilinx. Two sensors (same architecture) are now placed in two different regions of the die as shown in Fig. 4.11(c). Using the methodology discussed in Section 4.3, we generate the switching fabric very close to TDC0 sensor. Switching of the net, which will generate voltage transient is controlled using MicroBlaze (processor within an FPGA). Both sensor readings are stored in the local memory and transferred to the PC using the UART port. We have shown an example of the
sensor readings in Fig. 4.11(b). If the figure is rotated 90° anticlockwise and the readings are observed, we can see the voltage fluctuation trace similar to the one shown in Fig. 4.3(b). The readings in the Fig. 4.11(b) did overflow from 64-bin length and is shown for illustrative purposes only.

4.5.2 Experimental Results

The readings obtained from both sensors, TDC0 and TDC1 are shown in Fig. 4.12. The initial fluctuations, up to 125 clock cycles are due to default system activity. We toggle the switching fabric every 100 ns (50 clock cycles). Because of switching of such a high capacitance net, the entire power distribution network will see the ripple. Both sensors are successful in recording such a sudden fluctuation. Further, since the TDC0 sensor is closer to the switching fabric than TDC1, it will sense higher fluctuations than TDC1. From the figure, the TDC0 shows 23 total bit displacement as compared to TDC1 with only 15 bit displacement. TDC0 could record higher fluctuations than TDC1. The readings were similar when we implemented the switching circuit closer to TDC1 than TDC0. In this case, TDC1 sensor recorded higher displacement than TDC0.

This experiment illustrates that intra-die voltage fluctuation can exist within the design. We also demonstrate the capability of our proposed sensor in capturing these intra-die voltage fluctuations. The magnitude of voltage noise seen by the circuit closer to the switching fabric will be much higher than the circuit which is farther. Moreover, due to dynamic changes in the switching activity, the location of the voltage noise may vary. In other words, fluctuations may go undetected if a fixed location sensor is used.
4.6 Application: Reducing Safety Timing Margin

In this section, we will discuss how our proposed sensor system can be used instead of the current traditional approach to tolerate emergencies.

4.6.1 Motivation

The ability of the sensors to sense nanosecond voltage transients suggests several possibilities. One such advantage is in building fault-tolerant systems. Whenever emergencies occur, sensor outputs can be used as a feedback mechanism with either stalling the pipeline or re-evaluating. Current approaches to tolerate PVT fluctuations is to add extra safety timing margin. As shown in Fig. 4.13, extra timing margin is added to the critical path delay. Added safety margin is responsible for over-designing, which leads to performance degradation, area penalty, and power overheads. If we can replace the safety margin with a system that will help in determining such emergencies in real time, we can move our design to a more optimal point in power performance charts. We propose using our voltage sensor system in the design and reducing the safety timing margin incorporated during the design stage. Reduction in the timing margin can be chosen depending on the functional unit as well as the application. For example, consider an FFT DSP core that is used for image processing application. Such application can tolerate errors without much performance degradation (image quality) and therefore, smaller timing margins will suffice. For applications in which error cannot be
tolerated, more timing margin of 25% to 30% is required. The proposed sensor system can be used by judiciously shaving the timing margin based on application.

### 4.6.2 Experimental Methodology

For a given design, we first implement the fastest possible design by setting the timing constraint to zero. Based on this, we get a BASE delay, area, and power characteristics. Then, we consider a variety of timing margins from 5% to 30% added to the critical path delay for safe operation of design in an event of voltage variations. Higher voltage fluctuations as well as the more safety critical application will require more margin. We again implement the design, this time with a relaxed timing constraint that includes the extra timing margin. We can call this as a Voltage Variation Aware (VVA) design. This VVA design is slower, but is more area and power efficient. In the field, both the BASE and VVA will use the same clock frequency to operate. While BASE is tolerating the voltage fluctuations by its natural over designed circuitry, the VVA uses its alarm system to determine any voltage emergencies.

### 4.6.3 Experimental Results

In this section, we will present efficiency of deploying our proposed sensor system instead of adding extra safety timing margins and over-designing to tolerate emergencies. First, we
Figure 4.14: Area and power savings for FFT design.

Table 4.4: Area and power efficiency of using a VVA design over BASE design.

<table>
<thead>
<tr>
<th>Design</th>
<th>5%</th>
<th>10%</th>
<th>15%</th>
<th>20%</th>
<th>25%</th>
<th>30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adder</td>
<td>20.5</td>
<td>31.8</td>
<td>16.6</td>
<td>34.9</td>
<td>16.7</td>
<td>35.4</td>
</tr>
<tr>
<td></td>
<td>14.0</td>
<td>42.6</td>
<td>19.7</td>
<td>42.1</td>
<td>17.0</td>
<td>42.2</td>
</tr>
<tr>
<td>Mult</td>
<td>16.3</td>
<td>16.1</td>
<td>23.6</td>
<td>23.3</td>
<td>26.3</td>
<td>28.4</td>
</tr>
<tr>
<td></td>
<td>26.7</td>
<td>30.4</td>
<td>28.3</td>
<td>32.2</td>
<td>30.0</td>
<td>35.2</td>
</tr>
<tr>
<td>FPadd</td>
<td>7.2</td>
<td>0.3</td>
<td>7.7</td>
<td>0.2</td>
<td>10</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td>11.5</td>
<td>3.7</td>
<td>11.7</td>
<td>4.1</td>
<td>14.1</td>
<td>6.5</td>
</tr>
<tr>
<td>FPMult</td>
<td>4.9</td>
<td>17.4</td>
<td>7.4</td>
<td>17.4</td>
<td>7.1</td>
<td>17.5</td>
</tr>
<tr>
<td></td>
<td>7.9</td>
<td>19.1</td>
<td>8.2</td>
<td>19.0</td>
<td>11.6</td>
<td>20.0</td>
</tr>
<tr>
<td>FFT</td>
<td>6.3</td>
<td>5.9</td>
<td>10.5</td>
<td>12.6</td>
<td>17.7</td>
<td>22.9</td>
</tr>
<tr>
<td></td>
<td>19.5</td>
<td>24.5</td>
<td>20.8</td>
<td>25.8</td>
<td>22.1</td>
<td>27.5</td>
</tr>
<tr>
<td>DWT</td>
<td>23.1</td>
<td>12.9</td>
<td>25.8</td>
<td>11.1</td>
<td>27.5</td>
<td>9.7</td>
</tr>
<tr>
<td></td>
<td>30.1</td>
<td>9.5</td>
<td>32.3</td>
<td>10.3</td>
<td>39.5</td>
<td>11.3</td>
</tr>
</tbody>
</table>

examine this over an FFT design. Fig. 4.14 shows the area and power savings when our proposed system is used to remove safety timing margins of 5%, 10%, 15%, 20%, 25% and 30%. For example, from Fig. 4.14 if the designer could shave off 30% extra timing margin, and instead employ the sensor system, despite the overhead of the system an area saving of up to 23% and power saving of up to 27% can be achieved.

Table 4.4 provides similar information for a variety of circuits across several timing margin reductions from 5% to 30%. It can be seen that for the same timing allowance, different designs, experience different amounts of saving. For example, from Table 4.4 for 10% allowance in timing, Floating point adder gains no power efficiency while the integer adder receives 34.9% power reduction. This can be explained in Fig. 4.15, which shows the critical path...
dependency with voltage fluctuations. From the figure, we observe that the slope and hence, the degree of deterioration of critical path (slow down) with the voltage noise is different for different circuits. This is due to different characteristics of the critical path, such as number of gates, their types, drive strengths, type of edge traversing (rise/fall) etc. From the Table 4.4, we observe typical area savings of 4% - 40% and power savings of 2% to 42%. Hence, our proposed sensor system offers great area and power efficiency when deployed in reducing extra safety timing margin in designs. A savings of VVA system come at a cost of occasional re-evaluation or stalling of the pipeline.

4.7 Conclusion and Research Directions

We have proposed a fully digital on-chip process variation aware voltage sensor, which can be used to estimate the extent of voltage variations to detect any potential failures. It can be used to measure time-dependent, nanosecond-scale voltage transients. Voltage noise characteristics as measured on a 28nm FPGA was later used to analyze its impact on various designs.
With the help of our proposed sensor system we gave insights into the presence of intra-die voltage variation that exists and can go undetected when using the traditional static sensor approach. We demonstrated the tradeoff between the resolution of the voltage sensor achievable and the overhead incurred in its implementation.

Our proposed sensor, when implemented in 32nm technology node, can operate at a minimum clock speed of 1.5ns. In other words, we can sample output from the voltage sensor at 667 MSPS (mega sample per second). Further, we showed the capability of the proposed sensor in reducing safety timing margins and, thereby, achieving area and power savings. Under this application, our proposed sensor system will be 6% - 27.5% more power efficient than the traditional approach.

The ability to sense momentary voltage anomalies suggests several possibilities. One of the most promising ones is the enhanced protection against cryptographic voltage attacks, whether originating off-chip or on-chip. High-speed sensors would be implemented in a trusted portion of the fabric, or eventually as a hard IP block by the vendor. Upon detection of an anomaly, a system could attempt to zero out its secret information or shutdown. Low latency detection solutions like the one presented here are needed such that the time the cryptographic module spends in a compromised state is minimized. Further research is needed in validating that such sensing solution is resistant to attack.
Chapter 5

Conclusions

In this thesis, system level approaches using Voltage Scaling (VS) and Active Core Scaling (ACS) were presented that can greatly increase the energy-efficiency of a homogeneous many-core DSP design. At system level variability can impact both, speed, and power characteristics of a design. As was shown, this vastly impacts the energy-efficiency of the system. Hence, in this thesis we first present an accurate assessment of the impact of variabilities on the performance as well as energy efficiency of a design through detailed analysis. Later, we present a system-level solution in order to mitigate the undesired effect as well as maintain energy efficiency.

At first, we showed that the proposed hybrid methodology of VS and ACS can achieve nearly 8% (min) to 77% (max) of energy savings as compared to VS or ACS alone. Later, we presented the impact of PV on the energy-efficiency of such systems. We extended the analysis under two different workload environments, namely, static (fixed) and dynamic (scalable) throughput systems. Various DSP core subsystems were considered in the analysis to quantify the results. We observed that to achieve the highest level of energy efficiency, a system configuration will consist of optimal number of cores, $N_{opt}$, and their optimal operating voltage, $V_{dd_{opt}}$. We also observed that the configuration, $[N_{opt} V_{dd_{opt}}]$, varies widely between a Process Variation Ignorant (PVI) design versus a Process Variation Aware
(PVA) design.

We compared the energy-efficiency achievable by both PVA and PVI design for a range of desired throughputs. We showed that in the fixed throughput workload environment, PVA design will use an average of 29% less energy as compared to PVI systems. Furthermore, we presented that for a scalable throughput system, where the workload can change with time, PVA design will enjoy an average of 51% reduction in energy compared to a PVI design for various DSP applications.

We further analyzed that for a homogeneous many-core systems, PV can lead to variations in the core characteristics, mainly in terms of its speed and power. For a many-core system, now instead of a single frequency, the system will consist of cores with variation in speed and power. In this case, we showed that by judiciously selecting the cores necessary to meet the desired throughput, one can further increase the energy efficiency of the system. We introduced design corner aware (binning) methodology for selectively choosing the energy-efficient cores. System parameters necessary to define an energy efficient configuration will now consist of \([N_{opt} \ Vdd_{opt} \ Core_{opt}]\). We observed that the binning methodology can achieve on an average 49.3% increase in energy efficiency as compared to the system designed without the knowledge of speed variations.

As part of the investigation, we analyzed the impact of aging on the speed of the cores in many-core DSP systems. We evaluated two different policies in selecting the cores based on their aging profile. We observed policy A, which selects fastest available cores, outperforms policy B, an advocate of uniform aging of all cores, under both static as well as dynamic workload environments.

Prototype chip details of a homogeneous 8-core FIR systems, fabricated in 90nm technology node were presented. The prototype offered a good platform to validate our idea and methodology in building PVA energy-efficient architectures. In a 20 chip population, up to 7% and 26% variation in frequencies were observed among the cores at 0.9 V and 0.55 V respectively. The chip measurement results showed the difference in the system parameters
among the 20 chips under various workload environments. The variation in the configuration exists due to the impact of inter-die PV on the 8-cores in the system. Further, variations in \( Core_{opt} \) for different chips showed the impact of intra-die PV in a homogeneous many-core systems. We also analyzed energy efficiency of PVA systems as compared to global VS systems, and observed up to 28% energy savings by the PVA systems.

We further extended the chip measurement experiments for three different ambient temperatures. We presented the impact of temperature variations on the core characteristics in terms of speed and power. Later, we presented the impact of temperature variations on the energy-efficiency of a system. We observed that if a system is configured based on the room temperature setting, the system will lose 4% of its energy-efficiency with 18\(^\circ\) change in temperature.

In this thesis, we also investigated the problem of dynamic voltage variations. We presented design details of a fully digital on-chip PVA voltage sensor, which can be used to estimate the extent of variations and apply the right amounts of corrective actions to detect voltage emergencies. Our proposed voltage sensor, when implemented in a 32\(nm\) technology node can operate at a minimum speed of 1.5\(ns\). Therefore, this voltage sensor can be used to measure time-dependent voltage transients of the order of nanosecond speed. Further, we demonstrated the capability of our proposed sensor in reducing safety timing margins and, thereby, achieving area and power savings. In the above mentioned application, the proposed sensor system will be 6\% - 27.5\% more power efficient than the traditional approach of using guard bands.

Through the work presented in this thesis, we have analyzed the impact of process, voltage, temperature, and aging in a many-core design environment. The goal of this thesis was to provide a multi-dimensional investigation into the design of variation-aware energy-efficient homogeneous many-core systems.
Bibliography


[22] Saurabh Dighe, Sriram R Vangal, Paolo Aseron, Shasi Kumar, Tiju Jacob, Keith A Bowman, Jason Howard, James Tschanz, Vasantha Erraguntla, Nitin Borkar, et al.


