

# **Robust Low-Power Signal Processing and Communication Algorithms**

A Thesis Presented to  
The Academic Faculty

by

**Muhammad Mudassar Nisar**

In partial fulfillment  
of the Requirement for the Degree  
Doctor of Philosophy in the  
School of Electrical and Computer Engineering



School of Electrical and Computer Engineering  
Georgia Institute of Technology  
Atlanta, GA 30332

May 2010

# **Robust Low-Power Signal Processing and Communication Algorithms**

Approved by:

Dr. Abhijit Chatterjee, Advisor  
School of Electrical and Computer  
Engineering  
Georgia Institute of Technology

Dr. Sudhakar Yalamanchili  
School of Electrical and Computer  
Engineering  
Georgia Institute of Technology

Dr. Hsien-Hsin Sean Lee  
School of Electrical and Computer  
Engineering  
Georgia Institute of Technology

Dr. Saibal Mukhopadhyay  
School of Electrical and Computer  
Engineering  
Georgia Institute of Technology

Dr. Santosh Pande  
College of Computing  
Georgia Institute of Technology

Date Approved: December 07, 2009.

*Dedicated to my loving parents,  
Dr. Nisar Ahmed and Shamim Akhtar.*

## ACKNOWLEDGMENTS

I would like to express my sincerest gratitude to Prof. Abhijit Chatterjee for accepting me in his research group; his continuous support and guidance during the course of my research. He has been a source of great inspiration and motivation. His guidance, not just limited to academic research, has been invaluable for me during these years. I also take the opportunity to thank faculty members, Prof. Sudhakar Yalamanchili, Prof. Hsien-Hsin Sean Lee, Prof. Gabriel Rincon-Mora, Prof. Saibal Mukhopadhyay, and Prof. Santosh Pande for agreeing to serve on my proposal and dissertation committees and their valuable recommendations on my work.

I would like to thank Semiconductor Research Corporation (SRC), Giga Scale Research Center (GSRC), and National Science Foundation (NSF) for the support during various stages of my graduate studies at Georgia Tech. I sincerely appreciate the professional and personal support from my colleagues in our research group over the years. I thank Raj, Maryam, Vishwa, Jayaram, Shayam, Shreyas, Sehun, Hyun, and Deuk for their wonderful company.

I am also thankful to my wife Anmber Mudassar for her motivation to complete this dissertation. I am extremely grateful to my parents, brothers, and sisters for their unwavering and continuous support during the course of my life. My family has been the source of great inspiration for me and I am greatly indebted to them for their love and encouragement.

# Table of Contents

<b>ACKNOWLEDGEMENTS .....</b>	<b>IV</b>
<b>LIST OF TABLES .....</b>	<b>VII</b>
<b>LIST OF FIGURES .....</b>	<b>VIII</b>
<b>SUMMARY .....</b>	<b>XI</b>
<b>CHAPTER 1 - INTRODUCTION .....</b>	<b>1</b>
1.1 MOTIVATION.....	1
1.2 THESIS ORGANIZATION.....	3
<b>CHAPTER 2 - NANO-CMOS CHALLENGES: SOFT ERRORS, POWER, PROCESS VARIATIONS.....</b>	<b>5</b>
2.1 SOFT/TRANSIENT ERRORS .....	5
2.1.1 Soft Error Resilient Design.....	8
2.2 POWER CONSUMPTION IN DIGITAL CIRCUITS.....	9
2.2.1 Power Reduction Techniques .....	14
2.2.2 Power/Performance Adaptive Design.....	16
2.3 PROCESS VARIATIONS.....	18
2.3.1 Process Variation Tolerant Circuit Design .....	22
2.3.1.1 Design-Level Variation Tolerance.....	22
2.3.1.2 Post-Manufacture Variation Tolerance.....	23
<b>CHAPTER 3 - SYSTEM DESIGN MODELING .....</b>	<b>25</b>
3.1 OFDM TRX POWER CONSUMPTION.....	29
3.2 SIMULATION SETUP .....	30
3.2.1 Channel Modeling.....	30
3.2.2 Baseband TRX Model.....	31
3.3 FPGA IMPLEMENTATION.....	32
<b>CHAPTER 4 - GUIDED PROBABILISTIC COMPENSATION FOR LOW-POWER FILTERS.....</b>	<b>35</b>
4.1 OVERVIEW .....	36
4.2 STATE VARIABLE SYSTEM REPRESENTATION .....	39
4.3 LINEAR DIGITAL SYSTEM AND CHECKSUM-BASED ERROR DETECTION .....	41
4.3.1 Operator and State Gain – Gain Matrix .....	43
4.4 GUIDED COMPENSATION FOR LOW-POWER OPERATION .....	45
4.4.1 Shadow-Latches.....	45
4.4.2 Guided Error Compensation .....	46
4.4.3 Guided Error Compensation for Shared Hardware Implementation .....	49
4.4.4 Low Precision Checksum Effects.....	52
4.4.5 Dynamic Supply Voltage Control.....	52
4.5 SIMULATION SETUP .....	54
4.5.1 Simulation Results at Fixed Voltages.....	56
4.5.2 Simulation Results with DVOS .....	59
4.5.3 Power Savings and Area Overhead.....	62
4.6 CONCLUDING REMARKS .....	64

<b>CHAPTER 5 - SOFT ERROR MITIGATION AND LOW-POWER OPERATION OF NON-LINEAR FILTERS .....</b>	<b>66</b>
5.1 NON-LINEAR CIRCUITS – CHECKSUM BASED ERROR DETECTION .....	66
5.2 NON-LINEAR CIRCUITS - PROBABILISTIC SOFT ERROR COMPENSATION .....	70
5.2.1 Training Phase – Correction Vector Calculation .....	73
5.3 GUIDED PROBABILISTIC ERROR COMPENSATION FOR LOW-POWER .....	75
5.4 EVALUATION .....	76
5.4.1 Simulation setup.....	76
5.4.2 Probabilistic Compensation .....	78
5.4.3 Guided Probabilistic Compensation .....	80
5.5 CONCLUDING REMARKS .....	83
<b>CHAPTER 6 - CHANNEL AND VARIATION ADAPTIVE LOW-POWER BASEBAND PROCESSING .....</b>	<b>84</b>
6.1 PRIOR WORK .....	85
6.1.1 Tuneable Wordlength.....	85
6.1.2 Dynamic Voltage Scaling in a Pipeline Architecture .....	87
6.1.3 Power-Frequency Management .....	88
6.1.4 Voltage Overscaling and Algorithmic Noise Tolerance .....	88
6.2 MOTIVATION AND OVERVIEW .....	90
6.3 CHANNEL DRIVEN ADAPTATION TECHNIQUES FOR LOW-POWER .....	91
6.3.1 Locus Based Channel and Variation Tolerant Low-Power Processing .....	91
6.3.1.1 Adaptation Metric – EVM .....	93
6.3.1.2 Input Signal Scaling and Voltage Adjustment.....	95
6.3.1.3 Signal Scaling and Supply Voltage Control .....	100
6.3.1.4 Effects of Process Variations .....	100
6.3.1.5 Path Oscillation Timing Tests (POTTs) .....	101
6.3.1.6 Loci Based Operation: System Design and Characterization Phase .....	103
6.3.1.7 Evaluation .....	107
6.3.2 Dual Nested Loop Architecture .....	109
6.3.2.1 Proposed Power Control Methodology: Overview .....	110
6.3.2.2 Guided Probabilistic Compensation for Low-Power Digital-Filters .....	113
6.3.2.3 Signal Quality Metrics .....	114
6.3.2.4 Error Rate Metric for Modulating Supply Voltage.....	114
6.3.2.5 Nested Loop Architecture .....	115
6.3.2.6 Evaluation .....	117
6.4 APPLICATION DRIVEN CHANNEL AND VARIATION TOLERANT LOW-POWER BASEBAND DESIGN .....	121
6.4.1 Feed Forward - Image Quality Metric .....	123
6.4.2 Feedback - Channel and System Performance Metric .....	124
6.4.3 Locus based Operation - Design and Characterization Phase .....	124
6.5 EVALUATION .....	126
6.6 CONCLUDING REMARKS .....	128
<b>CHAPTER 7 - CONCLUSIONS AND FUTURE WORK .....</b>	<b>130</b>
<b>REFERENCES.....</b>	<b>132</b>

## List of Tables

Table 1: Parameter variations (nominal) in different technologies [53].	19
Table 2: Parameter variations ( $3\delta$ ) in different technologies [53].	20
Table 3: Technology parameter variation ( $3\delta$ /nominal values).	20
Table 4: WLAN access card power consumption in different protocols [84].	30
Table 5: Change in Full-Adder delay with VOS (65nm CMOS technology).	55
Table 6: Increase in error rate with VOS in LPF.	56
Table 7: Control parameter settings of feedback controllers calculated.	60
Table 8: A 15-tap LPF implemented in 65nm technology.	64
Table 9: Effect of sampling frequency on system SNR.	79
Table 10: Area overhead of the proposed schemes.	82
Table 11: Power consumption in baseband demodulator under voltage scaling.	118

## List of Figures

Figure 1: Soft error rate in different circuit blocks. ....	5
Figure 2: Comparison of the SRAM bit SER with the flip-flop/latch SER [11]. ....	7
Figure 3: (a) SER in different circuits. (b) Qcritical in logic/latches/SRAM [12]. ....	7
Figure 4: System complexity trend in non-mobile devices [20]. ....	10
Figure 5: System performance trend in non-mobile devices [20]. ....	10
Figure 6: System power consumption trend in non-mobile devices [20]. ....	11
Figure 7: Leakage and dynamic power contribution in total power. ....	12
Figure 8: SRAM cell leakage comparison between 65 nm and 45 nm [26]. ....	13
Figure 9: High-K metal gate leakage reduction [27]. ....	13
Figure 10: Limitation of battery technology. ....	14
Figure 11: Average number of dopant atoms in the device channel for different technology nodes [55]. ....	19
Figure 12: Frequency and leakage variation in 130 nm technology [58]. ....	21
Figure 13: Variation reduction using ABB. ....	24
Figure 14: Block diagram of an OFDM baseband transceiver. ....	26
Figure 15: Bits are mapped to complex numbers representing amplitude and phase (a) QPSK modulation (b) QAM-16 modulation. ....	27
Figure 16: OFDM symbol power spectrum [83]. ....	28
Figure 17: Virtual sub-carriers (null sub-carriers) for the out-of-band noise filtering. ....	29
Figure 18: Power distribution in state of the art WLAN transceivers [84]. ....	30
Figure 19: OFDM transceiver model. ....	32
Figure 20: Altera Stratix II DSP development board. ....	33
Figure 21: The OFDM TRX implementation in Simulink. ....	34
Figure 22: Proposed linear checksum based voltage overscaling scheme. ....	39
Figure 23: State variable system representation. ....	40
Figure 24: Linear state matrix representation with a checksum code. ....	42
Figure 25: State variable form representation with checksum based error detection. ....	42
Figure 26: Structure of a state-variable system with shared operators. ....	43
Figure 27: Gain Matrix corresponding to the system shown in Figure 26. ....	44
Figure 28: Reduced precision shadow-latch for error monitoring in MSB bits. ....	46
Figure 29: Guided probabilistic compensation methodology. Error detection and compensation is performed in the next clock cycle. ....	47
Figure 30: GPC architecture implementation. ....	48
Figure 31: Computational block implementation with shared hardware. ....	50
Figure 32: Guided probabilistic compensation in shared-pipelined architecture. ....	51
Figure 33: Supply voltage control. ....	53
Figure 34: Frequency spectrum of a low-pass FIR filter. ....	56
Figure 35: (a) Frequency spectrum of input applied to the LPF. (b) Frequency spectrum of filter output with and without probabilistic compensation. ....	57
Figure 36: LPF input, erroneous and compensated output under VOS in ....	58
Figure 37: (a) Simulation setup for DVOS (b) Input to LPF with added AWGN. ....	59
Figure 38: Proportional controller with preset error rate of 5%. ....	61
Figure 39: Proportional controller with preset error rate of 25%. ....	61
Figure 40: PI-controller, preset error rate of 5% (a) LPF regular and DVOS output ....	62
Figure 41: PID-controller, preset error rate of 5% (a) LPF regular and DVOS output ....	62



Figure 42: Power savings achieved with guided probabilistic compensation. ....	63
Figure 43: Non-linear digital circuit. ....	67
Figure 44: Time-freeze linearized circuit. ....	68
Figure 45: Training phase - Compensation vector selection. ....	74
Figure 46: Guided probabilistic error compensation with DVOS control. ....	76
Figure 47: Non-linear circuit used for simulation results. ....	77
Figure 48: SNR gain by probabilistic error compensation on different operators. ....	78
Figure 49: SNR gain at Operator 1 on different time instances. ....	79
Figure 50: System error rate and voltage values under GPC scheme. ....	80
Figure 51: System SNR and power savings using GPC scheme. ....	82
Figure 52: OFDM packet with preamble and added search symbol. ....	86
Figure 53: Supply voltage controller based on system preset and sampled error rate. ....	86
Figure 54: Switching activity based voltage-frequency controller in ....	88
Figure 55: Voltage overscaling for low-power filter operation. ....	89
Figure 56: Supply voltage feedback control ....	89
Figure 57: Block diagram of adaptation metric based receiver architecture. ....	92
Figure 58: EVM vs. BER relationship [90]. ....	94
Figure 59: QPSK constellation spread with varying channel conditions. ....	95
Figure 60: Critical path reduction with signal scaling. ....	96
Figure 61: Input (Voltage) scaling for low-power operation in an array multiplier. ....	97
Figure 62: Transposed form pipelined FIR filter. ....	98
Figure 63: (Top Row) QPSK constellation points with voltage scaling alone. (Bottom Row) Constellation points with combined W-V <sub>dd</sub> scaling. ....	99
Figure 64: EVM degradation with voltage scaling and combined W-V <sub>dd</sub> scaling. ....	99
Figure 65: Supply voltage as a function of W and V <sub>t</sub> . ....	101
Figure 66: Path sensitization for oscillation test. ....	102
Figure 67: Block level representation of POTTs scheme. ....	102
Figure 68: (a) System design phase. (b) System characterization phase. ....	104
Figure 69: Signal quality based module-level “voltage, wordlength” scaling. ....	107
Figure 70: Optimal power locus for a wireless receiver under changing channel conditions. ...	107
Figure 71: Device locus for run-time operation (a) symmetric voltage/wordlength modulation of modules (b) independent voltage/wordlength scaling on baseband modules. ....	108
Figure 72: Power distribution of the OFDM baseband receiver under 20% process variations. ....	109
Figure 73: Proposed real time feedback architecture. ....	109
Figure 74: Proposed real-time dual feedback control architecture. ....	110
Figure 75: Dual nested loop control strategy. ....	112
Figure 76: The qualitative relationship between supply voltage, wordlength, EVM and error rate. The optimal operating point of nested loops is defined by the quality requirements of end signal. .....	116
Figure 77: Fixed Channel (a) Outer loop - Wordlength modulation under EVM constraint. (b) Inner loop – Voltage modulation under Error rate constraint. ....	118
Figure 78: System wordlength and voltage for QPSK and QAM-16 modulation, for different channel conditions. Process variations result in different voltage settings. ....	119
Figure 79: Average power consumption in the baseband demodulator with the proposed nested loop architecture. ....	119
Figure 80: (a) Image received at preset EVM=30% and error rate=2%. (b) Image received with EVM=35% and error rate=10%. ....	120

Figure 81: Power histogram under process variations.....	121
Figure 82: Application driven power savings methodology.....	122
Figure 83: IQM vs. system EVM.....	124
Figure 84: $N_p \times V$ number of loci are stored at design time. System selects from a $V$ number of loci at run-time depending upon the end-application performance requirements. ....	126
Figure 85: (Top to bottom) Image X with 7-bit drop, Image X with 8-bit drop, Image Y with 7-bit drop.....	127
Figure 86: Object tracking in a swinging ball video.....	128
Figure 87: Proposed soft error mitigation, low-power and process tolerant techniques for robust, low-power signal processing and communication algorithms.....	131

# SUMMARY

The successful pursuit of the Moore's-law in the semiconductor industry has enabled the integration of highly complex functionalities on a single chip, thus enabling the proliferation of electronic devices in every spectrum of daily life. A new genre of such devices is wireless electronics (smart phones, netbooks, etc.) for mobile applications. However, the advent of Nano-CMOS (CMOS technologies below 90 nm) has brought new challenges for design, process, and test engineers. Soft errors have become a major reliability concern. These errors occur because of the combined effects of atmospheric radiation, and reduced noise margins. Another challenge is process variation that results in large spread in the delay and power distribution of circuits, and results in parametric yield loss. System power consumption is another major design challenge in Nano-CMOS because of high integration of transistors within a single chip and high amount of leakage current.

This thesis presents circuit-level techniques for soft error mitigation, low-power design with performance trade-off, and variation-tolerant low-power design. The proposed techniques can be divided into two broad categories. First, error compensation techniques are presented, which are used for soft error mitigation and also for low-power operation (with the help of guided circuitry as explained in next chapters) of linear and non-linear filters. Second, a framework for variation-tolerant low-power operation of wireless devices is presented. This framework analyzes the effects of circuit "tuning knobs" such as voltage, frequency, wordlength precision, etc. on system performance, and power efficiency. Process variations are considered as well, and the best operating tuning knob levels are determined, which results in maximum system wide power savings while keeping the system performance within acceptable limits. Different methods are presented for variation-tolerant and power-efficient wireless communication.

# CHAPTER 1

## INTRODUCTION

### 1.1 Motivation

Technology scaling as predicted by Gordon Moore [1][2] has been the driving force behind semiconductor industry growth. The scaling of transistor size in every generation provides advantages in power, performance, and cost. Every new technology generation [3]

- reduces power consumption by about 50% and energy consumption per transistor by about 65%;
- reduces gate delay by 30% thereby increasing operating frequency by 43% (although performance gain per device has decreased for Nano-CMOS because of threshold voltage limitations [4][5]);
- doubles transistor density, thus decreasing the cost per transistor.

However, at the same time that transistor sizes reduced, it became feasible to put more transistors in circuits to achieve complex functionalities, which ushered in an era of portable devices. The power consumption increased in high-performance, functionally complex devices and became a key design parameter in handheld devices. Also, circuits fabricated with these highly scaled technologies undergo a lot of intra-die and inter-die process variations. These variations result in significant circuit delay and power variations and thus affect the overall yield. Moreover, because of scaled node capacitances and supply voltages, circuits are more susceptible to soft errors.

Power consumption is a major concern in the design of mobile devices fitted with limited battery life. Power consumption in a device is divided into two main categories, dynamic power consumption when the device is active and leakage power consumption when device is in idle state. Multiple supply voltage, clock gating, and dynamic voltage and frequency scaling are well-known techniques for dynamic power reduction. Similarly, techniques such as multiple threshold voltages, power gating, and body biasing are employed in circuits to reduce the leakage power consumption. Some of the above-mentioned techniques are employed at the gate level and some at the component level. In many performance tolerant digital signal processing (DSP) applications, significant power savings can be achieved by exploiting the trade-off between system performance and power.

Process variations in Nano-CMOS technologies can cause up to a 20x deviation in the leakage power and a 1.3x variation in the circuit delay. Manufacturing yield is negatively impacted because of such large variations in the leakage power and circuit delay. Many circuits are designed with high process variations tolerance to have high yield at fabrication. The well-known design-time variant-tolerant techniques are gate sizing and supply voltage scaling. Application of forward body bias (FBB), reverse body bias (RBB), or adaptive body bias (ABB) to modulate the threshold voltage ( $V_t$ ) are the widely used post-manufacture circuit tuning techniques that bring the circuit delay and leakage power within an acceptable range.

Nano-CMOS circuits are more susceptible to soft errors because of lower supply voltage, smaller transistor sizes i.e. reduced node capacitances, and shorter depth of pipeline stages in modern VLSI circuits. Module redundancy such as triple module redundancy (TMR), hardened flip-flops, and gate sizing are the commonly practiced techniques for soft error resilient design. Traditionally, soft error tolerant techniques are applied to mission critical systems operating in high radiation environments. However, with the advent of Nano-CMOS soft errors are also a

major threat to correct operation of systems at terrestrial level. Therefore, better error mitigation techniques are needed to take full advantage of highly scaled devices than module redundancy that normally has an overhead of more than 200% in terms of area and power.

The objective of this research is to develop soft error resilient techniques with minimum overhead, variation-tolerant low-power techniques and system level low-power techniques. Many of these techniques are presented with a wireless OFDM system as a test vehicle. The organization of the thesis is presented in next section.

## **1.2 Thesis Organization**

The focus of this work is on soft error resilient, process variation tolerant, and low-power design techniques in Nano-CMOS. The motivation for this thesis is provided in Chapter 1. Chapter 2 details the above mentioned challenges and state of the art techniques for combating these problems. Quantitative analysis of the Nano-CMOS challenges is also presented to show the current trends in the semiconductor industry.

Chapter 3 details the OFDM transceiver, which is used as a test vehicle for the robust low-power signal processing techniques presented in this thesis. The OFDM transceiver is chosen as the test circuit as it contains all the modules required for testing of the proposed soft error mitigation, low-power, and process tolerant techniques. A brief introduction on the orthogonal frequency division multiplexing (OFDM) scheme is also provided in the chapter. Wireless channel and the transceiver modeling are explained. The OFDM transceiver implementation in hardware is explained as well.

Chapter 4 introduces the guided probabilistic error compensation (GPC) technique for low-power operation of the linear digital circuits. Linearized checksum codes are used for the first time along with time-delayed latches (shadow latches) to perform error compensation under

dynamic voltage overscaling. Since many DSP circuits are inherently noise-tolerant, the objective of the proposed scheme is to reduce system power with minimal impact on system performance, i.e., output signal quality. The state variable representation of systems is also presented in the chapter, and prior work in probabilistic error compensation in linear systems is summarized.

Chapter 5 presents the soft error mitigation technique in non-linear circuits using the linear error correcting codes. The proposed mitigation technique performs probabilistic error compensation with the minimum hardware overhead. Application of the guided probabilistic compensation method to non-linear filters for low-power operation is also presented.

Chapter 6 introduces channel driven variation tolerant low-power design methodologies, which are implemented in the wireless OFDM transceiver. The proposed schemes always strive to operate the system at the worst-case acceptable performance limit, thus saving considerable power when the operating conditions are not worst-case.

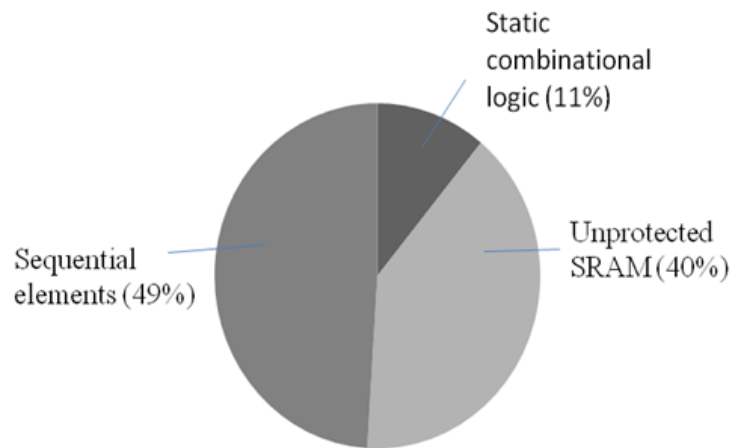
Chapter 7 concludes the major contributions of the thesis and provides a direction for the future work.

## CHAPTER 2

# NANO-CMOS CHALLENGES: SOFT ERRORS, POWER, PROCESS VARIATIONS

### 2.1 Soft/Transient Errors

As feature sizes decrease with new fabrication technologies, single event transient (SET), single event upset (SEU), and multiple bit upset (MBU) effects dominate the radiation response in microcircuits. It is known for a long time that charged particles cause SEU in latches and memory elements (SRAM, DRAM) [6][7]. Moreover, they cause SET in combinational logic, clock lines and circuit control lines. The distribution of soft errors in combinational, sequential and memory elements [8] is shown in Figure 1. SET effects are becoming more prominent in the nanotechnology (<90 nm) as with reduced node capacitances, higher clock frequencies and lower noise margins, many of these transients are captured as errors in the latching circuitry [9][10].



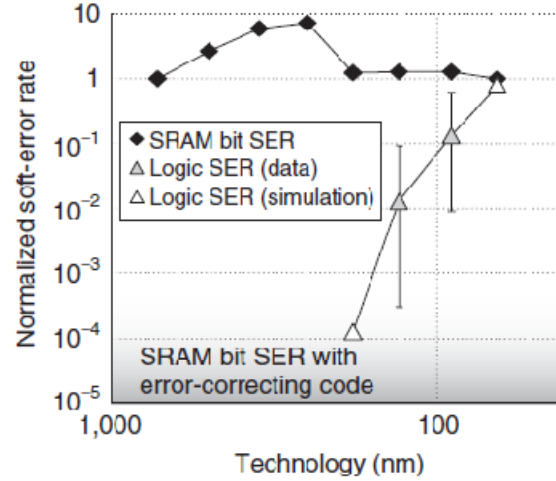
**Figure 1: Soft error rate in different circuit blocks.**

The strike of a charged particle on a microcircuit results in generation of electron-hole pairs. These pairs are generated as a result of interactions of ion with bounded electrons in material,



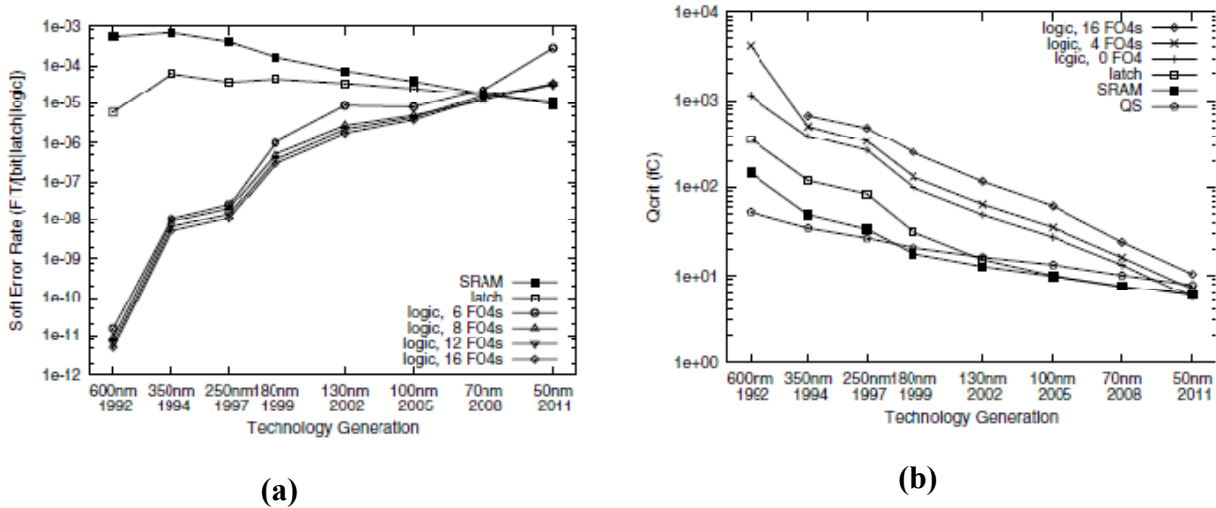
causing ionization of the material. These electron-hole pairs are of no consequence in bulk silicon since they eventually recombine. However, in presence of electric fields, these electron-hole pairs quickly drift in opposite directions to be captured by the voltage sources responsible for the field. In bulk CMOS ICs electric fields are present at every p-n junction. Therefore, if an ion strikes a junction connected to a signal node, a transient current of some duration is observed at the node. In data storage elements such as latches, SRAM and DRAM, the effect of this transient current depends on the circuit response to the charge collected on a signal node. This means that a signal node capacitance ( $C$ ) determines the amount of voltage swing ( $dV=dQ/C$ ) as a result of collected charge ( $dQ$ ). Whenever, the collected charge reaches a critical value ( $Q_{critical}$ ) sufficient to drive a node voltage past the switching voltage, data value flips at the signal node (causing an error). In fast combinational circuits, a transient can flow to a latching element and can be stored as a valid signal (causing an error).

The effects of SET had been limited in combinational circuits because of three masking techniques, 1) Logical masking; a transient on a gate fails to cause an error at the circuit output as subsequent gates in the path are off, 2) Electrical masking; a transient is attenuated by electrical properties of subsequent gates to a level where it does not affect the circuit output, 3) Latching window masking; a transient reaches the latch but falls outside the latching window period. However, in the nanotechnology these masking mechanisms are becoming inadequate and more of the current transients are causing errors at a circuit output. In recent work [11][12], it is shown that error rate in combinational circuits will reach that of unhardened data storage elements. Figure 2 shows the increasing SER sensitivity in combination logic with scaling, the variation in logic SER is due to the dozens of logic types tested.



**Figure 2: Comparison of the SRAM bit SER with the flip-flop/latch SER [11].**

Figure 3(a) shows the soft error rate (SER) trend in SRAM, latches and logic for different technology generations. It is obvious from the graph that soft error rates in combinational logic are increasing at a much higher rate than in data storage elements. The primary reason for SER increase in logic circuits is the decrease of  $Q_{critical}$  with technology scaling as shown in Figure 3(b). Typical sizes of logic gates are larger than memory elements (density is more important in memory), thus making  $Q_{critical}$  scaling more pronounced in logic circuits as compared to memory with decreasing feature sizes.



**Figure 3: (a) SER in different circuits. (b)  $Q_{critical}$  in logic/latches/SRAM [12].**

### 2.1.1 Soft Error Resilient Design

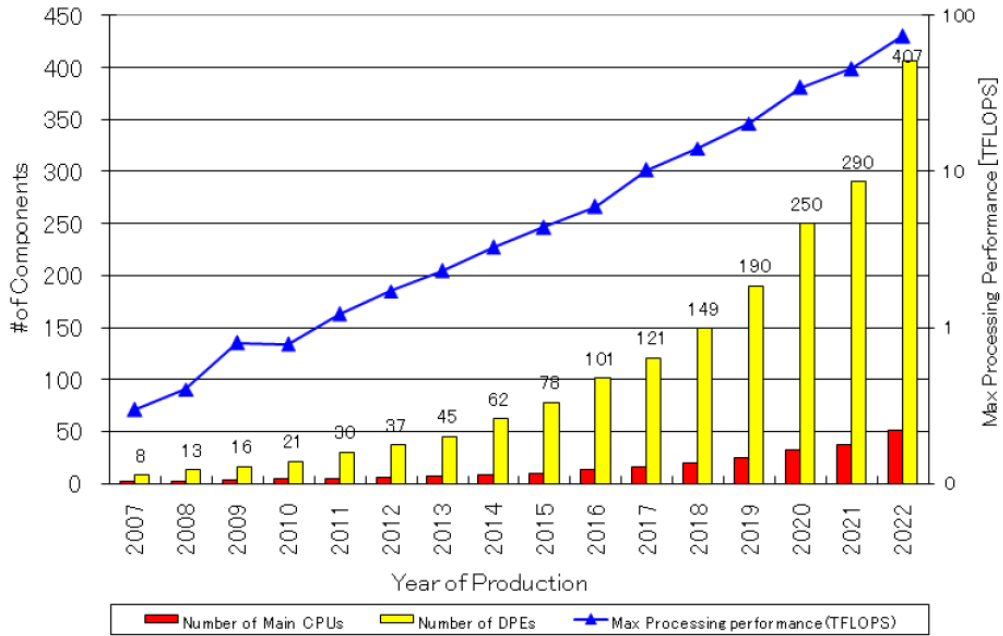
Current soft error (radiation) hardening techniques can be categorized as radiation hardening (Rad-hard) by fabrication (RHBF) and radiation hardening by design (RHBD). The RHBF approach includes methods such as substrate engineering, silicon on insulator (SOI), deep-trench isolation and guard ring oscillators. The RHBF techniques are quite effective in mitigating the radiation effects but they come at expense of increased fabrication costs, low yield and much higher ramp-up time to new fabrication technologies. The RHBD techniques include circuit, gate, and transistor level solutions. These techniques aim to reduce the probability of single event effects (SEE) observation at the primary outputs with a minimal impact on the circuit delay, power, and area. In general, error correcting codes and hardened memory cell designs are used in memory elements (SRAM/DRAM). In sequential logic, techniques such as increased capacitive loading at gates, gate resizing, new gate designs (hardened flip-flops [21]) and hardware redundancy (Muller C-elements [8], SET immune latch [13]) are used.

Traditionally, system level fault-tolerant techniques employ redundancy such as hardware, software, time, and/or information. The hardware redundancy techniques, such as TMR (triple module redundancy), have high hardware overhead. The high area and power costs associated with these techniques make them impractical for general applications. At the same time, such high cost is not necessary in most cases, especially in non-real-time systems. Therefore often techniques such as time redundancy, partial duplication, and software redundancy are employed. Such techniques have a less hardware overhead but have a negative impact on the system performance. In the literature, there are many circuit level techniques for the protection of flip-flops (latches) from errors. In [13], it was proposed to replace every latch with two or three latches that are clocked with a fixed phase-delay or their inputs arrive after a fixed phase-delay. A voting circuitry is used to pick the correct latch value. The redundancy of the scan flip-flops is

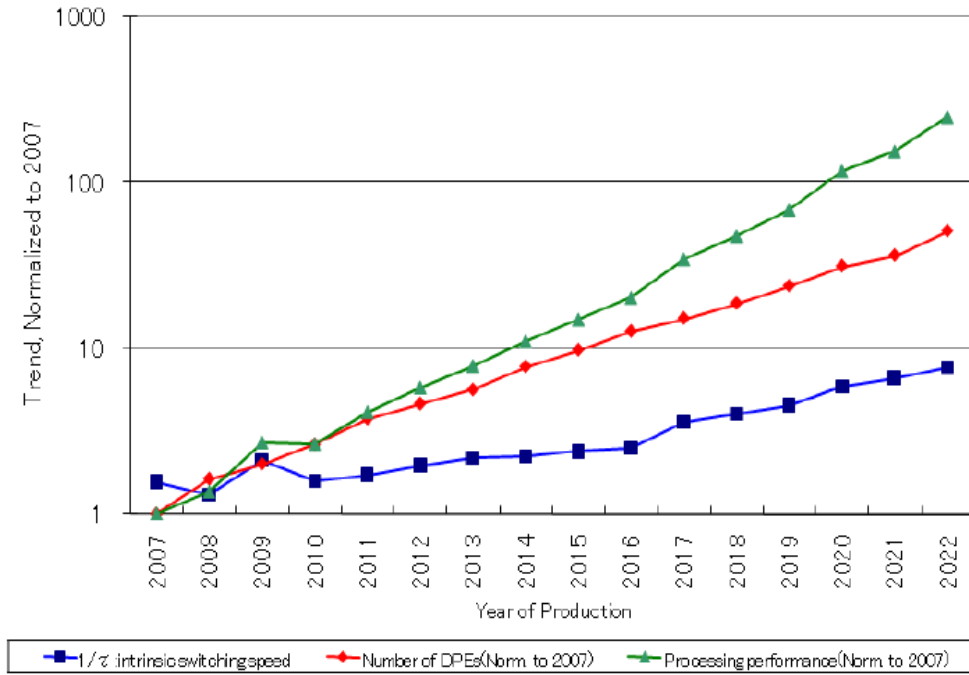
used along with the Muller C-element to protect the flip-flops against transient errors in [8]. The Muller C-element is a two-input and one-output component, which keeps its output value if its two inputs do not match. A soft error immune latch that keeps its state on three different nodes was proposed in [21]. When the value is destroyed in one of the nodes, the other two nodes still hold the right value. Techniques are present in past work for hardening the combinational circuits against transient errors by optimizing the gate sizes and increasing the gate load capacitances [18].

## **2.2 Power Consumption in Digital Circuits**

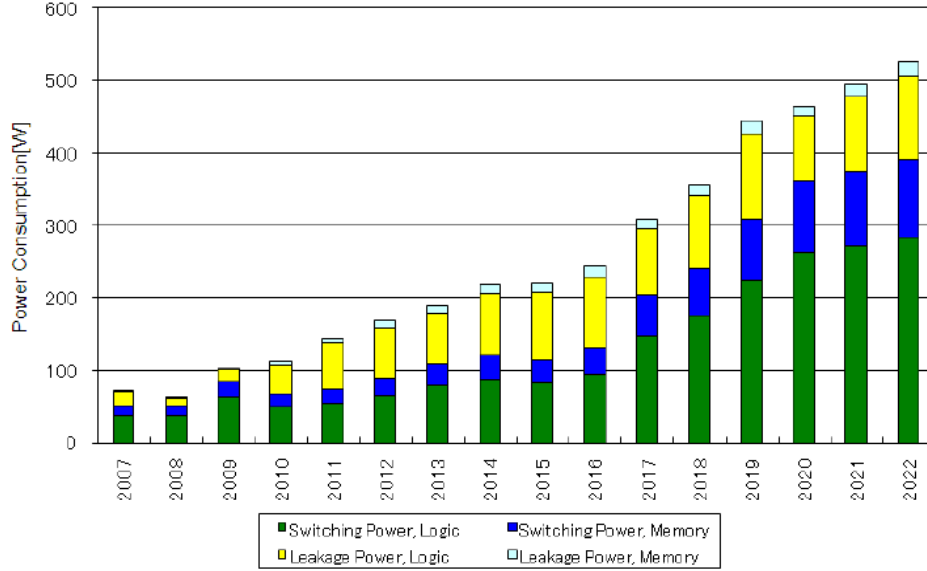
Power consumption has been increasing in the digital systems because of high device integration to implement complex functions on a single chip and high performance requirements. The successful pursuit of the Moore's Law has continually reduced the cost of silicon devices, thus enabling the implementation of highly complex systems. Figure 4 and Figure 5 shows the system complexity and performance requirements (predicted) trends in non-mobile devices. There is a predicted trend of 30% year-to-year increase in system complexity and performance requirements of up to 70 TFLOPS by the year 2022. Similar trends are predicted for the mobile devices as well. These highly complex and high-performance systems will consume considerable power as shown in the predicted trend of Figure 6.



**Figure 4: System complexity trend in non-mobile devices [20].**



**Figure 5: System performance trend in non-mobile devices [20].**



**Figure 6: System power consumption trend in non-mobile devices [20].**

Power consumption in digital circuits has two main components: dynamic and static. Dynamic power loss is due to the switching activity of the circuit nodes and is proportional to the node capacitances, supply voltage, switching activity of the nodes, circuit operating frequency, and short circuit power. Short circuit power makes roughly 10% of the dynamic power consumption [22][23]. Dynamic power consumption in a digital circuit is given below

$$P_{dynamic} = 0.5 \times C \times V_{DD}^2 \times \alpha \times f + P_{sc} , \quad (1)$$

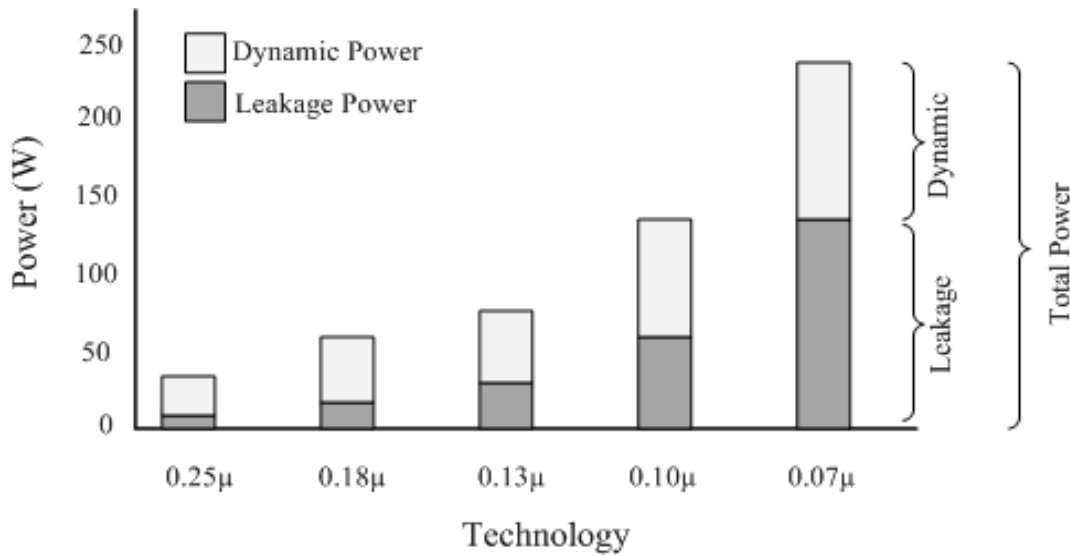
where  $C$  is the total capacitance,  $V_{DD}$  is the supply voltage,  $\alpha$  is the switching activity,  $f$  is the operating frequency and  $P_{sc}$  is the short circuit power.

Static power consumption is the power loss in a circuit when the circuit is idle. It is composed of three components: sub-threshold leakage, drain junction leakage and gate leakage. Sub-threshold leakage is the dominant factor and causes power dissipation as given below

$$P_{sub-th} = V_{DD} \times I_s \times e^{\frac{V_{gs}-V_{th}}{n \cdot V_T}} \times (1 - e^{\frac{V_{ds}}{V_T}}) , \quad (2)$$

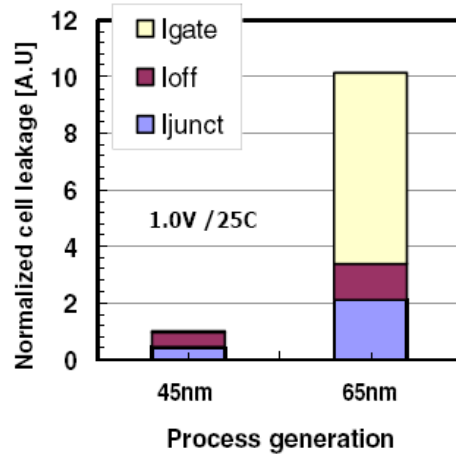
where  $V_{DD}$  is the supply voltage,  $I_s$  is the process and circuit dependent constant,  $n$  is the sub-threshold swing coefficient,  $V_{th}$  is the threshold voltage, and  $V_T$  is the thermal voltage. Sub-

threshold power loss has increased drastically in the scaled technologies because of the exponential relationship of sub-threshold current to threshold voltage. Also, since the supply voltage is not scaled down as much as the feature sizes to have high-performance transistors [24], gate leakage has increased because of low gate oxide thickness. As a result, static power consumption has become a critical design parameter in recent technologies. Since recently, the dynamic and static power consumption trend in recent technologies is shown in Figure 7 [25].

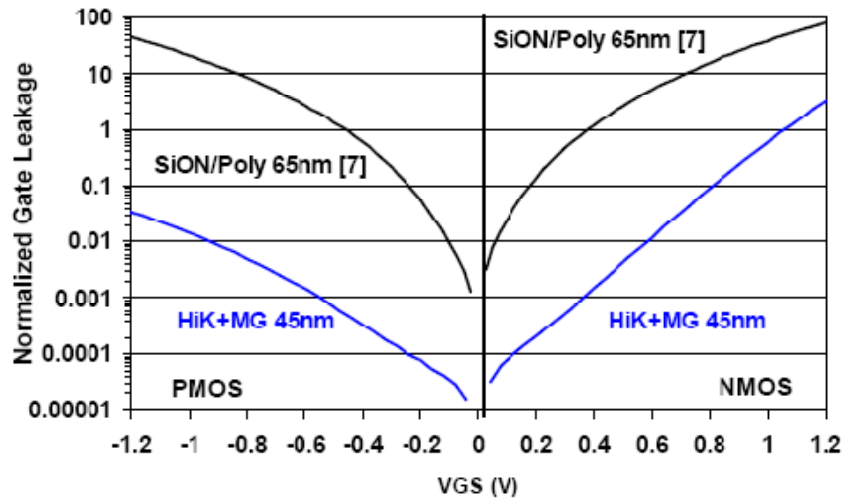


**Figure 7: Leakage and dynamic power contribution in total power.**

However, the advent of high-k metal gates is promising to drastically decrease the leakage current in the circuits. It is observed in [26] that 45 nm high-k metal gates reduce the leakage current in a SRAM bit cell by 10x as shown in Figure 8. These gates provide dramatic gate leakage reduction as compared to 65 nm CMOS bulk technology, gate leakage is reduced by >25x for NMOS and by 1000x for PMOS devices (see Figure 9). Coupled with other leakage reduction techniques such as the gate length increase, it can be argued that dynamic power will remain the dominant power consumption source in the future digital circuits.



**Figure 8: SRAM cell leakage comparison between 65 nm and 45 nm [26].**

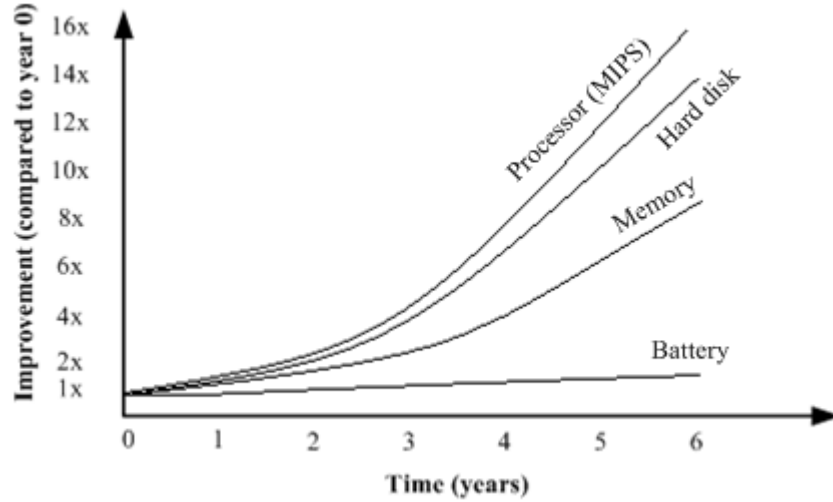


**Figure 9: High-K metal gate leakage reduction [27].**

Battery life in handheld devices has always been a design concern. Handheld devices are becoming omnipresent, and with their increased capabilities have found usage in almost every aspect of life. However, battery life is not increasing at the same pace as the energy requirements of handheld electronics. Lithium-ion batteries offer the highest capacity among today's rechargeable batteries. Their capacity has increased by about 10% per year [28] and lags far behind the increase in computation power. Even the improved battery capacities in the future, probably will not quench the energy needs of handheld devices or diminish the importance of power management. Figure 10 shows the improvement in battery life compared to processor



(MIPS), hard disk, and memory capacity over the years. It is obvious from the figure that battery energy storage capacity is lacking far behind the other technologies [25].



**Figure 10: Limitation of battery technology.**

### 2.2.1 Power Reduction Techniques

Power consumption is a major design parameter and a lot of work has been done in the field of power optimization [29]. Power optimization techniques range from system level to circuit level. In the following, some well-known power optimization techniques for leakage and dynamic power are briefly discussed. The focus of the discussion is on energy-efficient systems that provide performance on demand. These energy-efficient systems strive to operate with minimum power while satisfying the minimum system performance requirements.

As is obvious from Figure 7, static power consumption is a major concern in highly scaled technologies. The most natural way to reduce the leakage current is to turn off the supply voltage of a circuit in the standby mode. This is achieved using a power gating technique [30]. In a power gating scheme, one NMOS transistor called the sleep transistor is placed in series with the logic block to create a virtual ground. During normal mode of operation the sleep transistor is on, therefore the circuit functions as usual. However, during the standby mode the sleep transistor is

turned off, thus disconnecting the ground from the circuit. In practice dual threshold voltage (dual- $V_t$ ) or multi-threshold voltage transistors (MTCMOS) are used in power gating. In these technologies, low  $V_t$  transistors are used to implement the logic and high  $V_t$  transistors are used as sleep transistors. Body bias control is another useful technique for reducing the leakage current of a circuit. Reverse body bias is used to increase the threshold voltage of transistors, which in turn decreases the circuit leakage. However, this method is becoming less effective as the supply voltage is scaled down in new technologies [31]. The leakage current of a gate is a strong function of its inputs. For example, the minimum leakage current of a Nand2 gate is in the case when both of its inputs are low. In this case, both the NMOS transistors present in series in the Nand2 gate are off, thus offering maximum effective resistance. This in turn is known as the “stack effect” i.e. the phenomenon when the leakage through a stack of two or more off transistors in series is significantly less than a single device leakage. Different methods such as the Boolean satisfiability problem [31], 2-to-1 multiplexer [32], and gate modification are used to control the gate inputs for maximum leakage control.

The other major source of power, dynamic power consumption as given in equation 1 can be reduced by

- decreasing the switching capacitance,
- decreasing the supply voltage ( $V_{DD}$ ).

Clock gating is a commonly used circuit-level technique for reducing the switching capacitance of a circuit [34][35]. The switching activity in the unused circuit is eliminated by disabling the clock to that portion of the circuit. The method results in power savings by eliminating the switching activity in the flip-flops, gates, and clock tree of a circuit block. Dynamic supply voltage scaling is another very effective technique for reducing the power consumption of a circuit.

### **2.2.2 Power/Performance Adaptive Design**

The data wordlength optimization in a circuit decreases the power consumption by reducing the switching activity. Simulation based techniques [36]-[39] have been proposed to find the optimal wordlength for digital signal processing algorithms in wireless communications and filtering applications. Such algorithms optimize the wordlength according to the predetermined system-level performance metrics. Often, the resulting digital circuits are implemented with large wordlength values. A dynamic wordlength tuning technique is presented in [40] for digital baseband OFDM signal processing algorithms. The scheme allows dynamic adjustment of the wordlengths of digital filtering and FFT operations by continuous monitoring of the error vector magnitude (EVM) of the demodulated signal, where EVM is the system performance metric. When the performance of the system is adequate for the required quality i.e. EVM value is higher than a predetermined threshold, data wordlength is reduced. Otherwise, the system uses the longer wordlength. Clock gating is used in the scheme for reducing the data wordlength by halting the redundant portion of the circuit.

Because of the quadratic relationship of the supply voltage with power, voltage scaling is a very effective power savings technique. In the multiple static voltage islands technique, different islands (components) in the circuit are operated at different voltages depending on their timing constraints [41]. The dynamic voltage and dynamic frequency (DVDF) scaling technique is employed in many pipelined architectures [42][43]. Decreasing the supply voltage increases the delay of the circuit. This makes it necessary to reduce circuit clock frequency as well. The core idea behind this power savings technique is to provide performance on demand. In RAZOR [45], the concept of performance on demand with the output quality feedback was presented. The feedback system modulates the supply voltage depending on the error rate in the system. The error rate is calculated with the help of shadow latches. These shadow latches are augmented

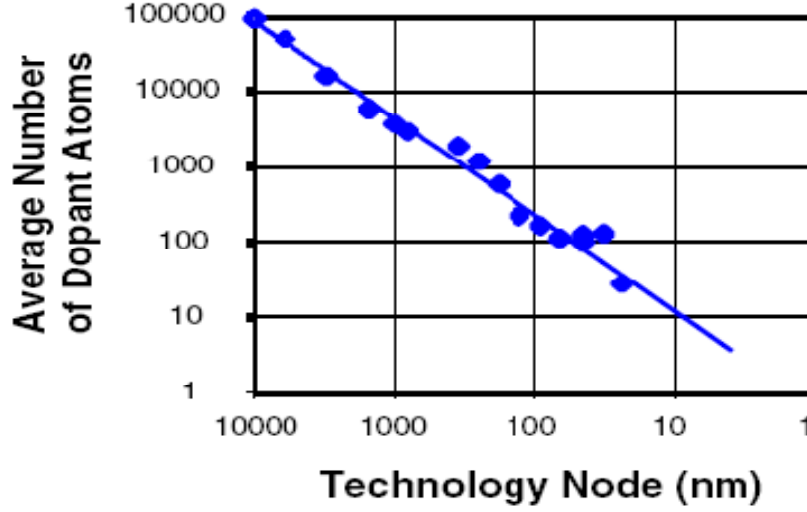
with the main flip-flops but are operated at a delayed clock. At a reduced voltage, if the logic path meets the setup time of the main flip-flop, then the main flip-flop and shadow latch will have the same data. However, if the logic path does not complete its computation in time, the main flip-flop will latch an incorrect data, while the shadow latch will latch the late arriving correct value. In such a case, the comparison of shadow latch value with the main flip-flop value will generate an error signal. The incorrect value will be flushed from the pipeline, incurring a one cycle penalty. The supply voltage is modulated by a proportional controller based on system error rate. If the error rate in the system is less, then it means that voltage can be reduced further. However, if the error rate is high, supply voltage is increased to limit the number of errors occurring. The technique exploits the data dependence of circuit delay and results in considerable power savings.

Algorithmic noise tolerant techniques [46][46][47] allow energy efficient digital signal processing (DSP). The core idea is to permit errors to occur in the DSP block and then correct them via a separate error control block. This approach of error/noise tolerance achieves higher energy efficiencies compared to the noise-mitigating techniques. Voltage overscaling is used in the main DSP block to save power. It is observed that even when the circuit logic delay is marginally longer than the critical path delay of the circuit (due to supply voltage reduction), the resulting logic error rate increases marginally because the circuit-critical paths are excited infrequently by the applied stimulus [51]. As the supply voltage is decreased further, this error rate increases rapidly leading to a large deterioration in the output signal quality. Different error control blocks have been presented in previous work. In [46], linear prediction-based output approximation and error cancellation methods are used for error control blocks. In linear prediction schemes, it is assumed that errors are correlated across time and can occur with a frequency that is sufficiently less than  $1/2N_p$ , where  $N_p$  is the number of predictor taps. The

accuracy of the error cancellation scheme and the resulting system performance depends on how well the system is “trained” to perform error cancellation for an input signal with specified statistics. A technique that uses reduced precision redundancy is presented in [47]. In this approach, the DSP block is duplicated but with reduced precision. The quantization noise resulting from the reduced precision quickly becomes a bottleneck in such a scheme. To mitigate the quantization noise, a least significant bits (LSB) error estimator is implemented, which compensates for the quantization noise in the system.

## 2.3 Process Variations

Process variations occur during processing and masking steps of a wafer. Both transistors and interconnects undergo physical process variations because of the imperfections in the processing steps [52][53]. Process variations can be divided into inter-die process variations and intra-die process variations [54]. Inter-die process variations affect all devices on a die in the same way. Intra-die process variations affect devices on a die in a random or locally correlated way. Process variations alter the device geometric characteristics, and the material parameters. The geometric variation in a CMOS transistor consists of oxide thickness ( $T_{ox}$ ), effective channel length ( $L_{eff}$ ) and device width ( $W$ ). Channel-doping variation is the most significant variation in the materials property of devices and results in the threshold voltage ( $V_{th}$ ) variation. Figure 11 shows the decreasing number of dopant atoms in the channel for different technology nodes.



**Figure 11: Average number of dopant atoms in the device channel for different technology nodes [55].**

As the number of dopant atoms in the channel decreases with device scaling, the impact of the variation associated with the dopant atoms increases. Nominal and  $3\delta$  parameter variations of  $L_{\text{eff}}$ ,  $T_{\text{ox}}$ ,  $V_{\text{th}}$ , and  $W$  are summarized in Table 1 and Table 2, respectively. Percentage parameter variation is taken as the ratio of  $3\delta$  to the nominal value and is given in Table 3. From Table 3, it is obvious that the parameter variations have increased with every technology generation, with the most pronounced affect on  $L_{\text{eff}}$ .

**Table 1: Parameter variations (nominal) in different technologies [53].**

Parameter	1997	1999	2002	2005	2006
$L_{\text{eff}}(\text{nm})$	250	180	130	100	70
$T_{\text{ox}}(\text{nm})$	5	4.5	4	3.5	3
$V_{\text{th}}(\text{volts})$	0.5	0.45	0.4	0.35	0.3
$W(\mu\text{m})$	0.8	0.65	0.5	0.4	0.3

**Table 2: Parameter variations ( $3\delta$ ) in different technologies [53].**

Parameter	1997	1999	2002	2005	2006
$L_{\text{eff}}(\text{nm})$	80	60	45	40	33
$T_{\text{ox}}(\text{nm})$	0.4	0.36	0.39	0.42	0.48
$V_{\text{th}}(\text{mvolts})$	50	45	40	40	40
$W(\mu\text{m})$	0.2	0.17	0.14	0.12	0.1

**Table 3: Technology parameter variation ( $3\delta/\text{nominal values}$ ).**

Parameter Variation ( $3\delta/\text{nominal}$ )	1997	1999	2002	2005	2006
$L_{\text{eff}}$	32%	33%	35%	40%	47%
$T_{\text{ox}}$	8%	8%	9.8%	12%	16%
$V_{\text{th}}$	10%	10%	10%	11%	13.3%
$W$	25%	26.2%	28%	30%	33.3%

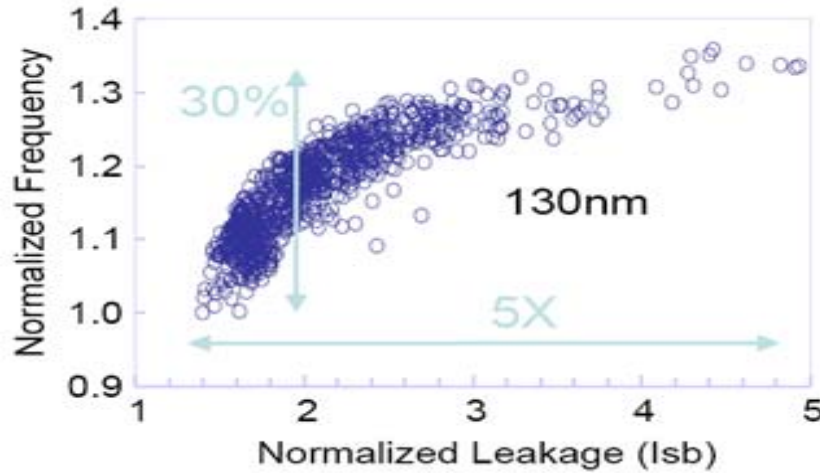
Process variations result in large spread of delay and frequency characteristics of devices. The frequency and leakage variation of circuits in a wafer for 130 nm technology are shown in Figure 12. It can be seen that because of parametric variations, devices undergo a 30% frequency variation and a 5x variation in standby leakage current ( $I_{\text{sb}}$ ). This huge variation in frequency and leakage power has resulted in frequency binning and affects the overall yield. High-frequency chips with high  $I_{\text{sb}}$  and low-frequency chips with reasonably high  $I_{\text{sb}}$  are discarded. Sub-threshold voltage ( $V_{\text{th}}$ ) variation is the main contributor in this huge  $I_{\text{sb}}$  variation. The  $V_{\text{th}}$  relation with sub-threshold current is given as

$$I_{sub-V_{th}} = I_s e^{\frac{V_{GS}-V_{th}}{nV_T}} \left(1 - e^{\frac{V_{DS}}{V_T}}\right), \quad (3)$$

where  $I_s$  is the process and circuit-dependent constant,  $n$  is the sub-threshold swing coefficient,  $V_{GS}$  is the gate-to-source voltage,  $V_{DS}$  is the drain-to-source voltage,  $V_{th}$  is the threshold voltage, and  $V_T$  is the thermal voltage. Similarly, supply voltage relation with circuit delay is given as

$$\tau_d = \frac{C_L V_{DD}}{\beta (V_{DD} - V_{th})^\gamma}, \quad (4)$$

where  $V_{DD}$  is the system supply voltage,  $\beta$  is the device trans-conductance,  $C_L$  is the circuit load capacitance,  $V_{th}$  is the threshold voltage and  $\gamma$  is the velocity saturation coefficient, which is between 1.2 and 1.5 for the current technologies. As is obvious from equations 3 and 4, sub-threshold voltage  $V_{th}$  has an exponential relationship with the leakage current and a linear relation with the circuit delay. Therefore, variation in  $V_{th}$  affects the circuit leakage power more adversely as compared to the circuit delay.



**Figure 12: Frequency and leakage variation in 130 nm technology [58].**



### 2.3.1 Process Variation Tolerant Circuit Design

The variation tolerant circuit design techniques can be categorized into two main categories: the design-level optimization techniques and the post-manufacture techniques. The design-level optimization techniques are discussed first.

#### 2.3.1.1 Design-Level Variation Tolerance

Conventionally, a static timing analysis technique is used for designing different circuit parameters such as supply voltage, threshold voltage and transistor sizing for given power, area and delay constraints [61]-[65]. However, many devices designed for the nominal case (using static timing analysis) will fail because of the increased variations in new technologies. Designing the circuits for the worst-case is extremely conservative and is unacceptable in many applications because of high power and delay costs.

Statistical timing analysis techniques striving to optimize the circuit power and performance by gate sizing while considering process variations were recently proposed [66][69]. A Lagrangian-based relaxation is proposed for transistor sizing for circuits under intra-die and inter-die variations in [66]. The objective of the technique is to meet the delay requirements of a circuit with a certain degree of confidence while keeping the area and power within a given set of constraints. The optimization complexity of the procedure is linear and results in a 19% area/power savings compared to the worst-case design. A sensitivity-based heuristic technique for dual- $V_{th}$  assignment and gate sizing is proposed in [66]. The objective of the technique is to minimize the leakage power. Each gate in a circuit is assigned two statistical sensitivity metrics, one is the sensitivity of a gate to the gate size and the other is the sensitivity of a gate to the  $V_{th}$ . The algorithm tries to find the best gates to have the high  $V_t$  and the gates to be sized up, such that the circuit leakage and its variation are minimized. The algorithm has a relatively higher

complexity of  $O(n^3)$  and results in a leakage power reduction of 15-35% compared to deterministic analysis. The statistical gate sizing technique of [68] reduces the delay variation by 72% at a cost of 20% increase in design area. The technique of [69] performs yield improvement with simultaneous delay and leakage constraints. This technique employs a non-linear optimizer and reports a 40% yield improvement compared to the deterministic approach.

### 2.3.1.2 Post-Manufacture Variation Tolerance

Adaptive body (substrate) biasing is one of the more important post-manufacture techniques to minimize the impact of process variations [70][71]. This technique is used for leakage power reduction and also circuit delay adjustment. For an NMOS device with the substrate connected to ground, a negative bias called the reverse body bias (RBB), increases the threshold voltage ( $V_{th}$ ). On the contrary, a positive bias called the forward body bias (FBB), decreases the  $V_{th}$ . Similarly, for a PMOS device with the substrate connected to  $V_{DD}$ , a voltage lower (higher) than  $V_{DD}$  is used for RBB (FBB). The relationship of source-bulk (substrate) voltage with  $V_{th}$  is given as

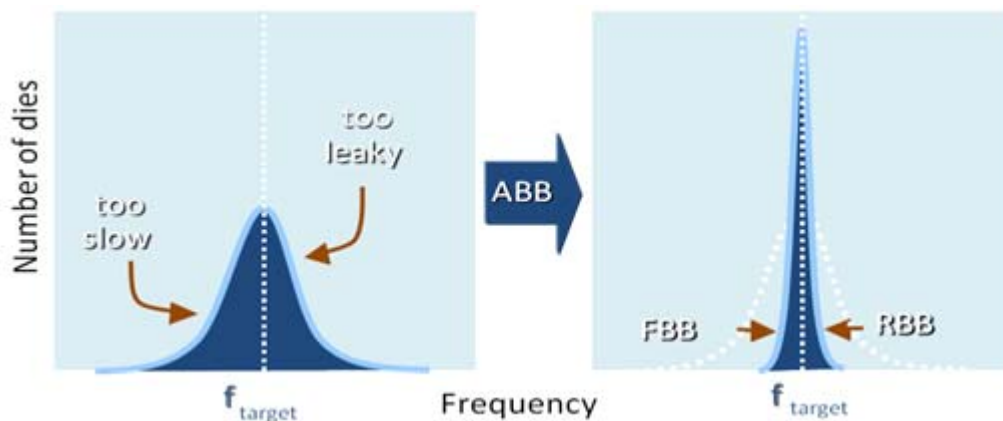
$$V_{th} = V_{to} + \gamma(\sqrt{2|\phi_f| + V_{sb}} - \sqrt{2|\phi_f|}), \quad (5)$$

where  $V_{to}$  is the threshold voltage at  $V_{sb} = 0$ ,  $V_{sb}$  is the source-bulk voltage,  $\gamma$  is the body-effect coefficient, and  $\phi_f$  is the substrate Fermi potential. An increase in  $V_{th}$  (RBB) reduces the leakage current at the cost of increased circuit delay, as evident from equations 3 and 4. Similarly, a decrease in  $V_{th}$  (FBB) reduces the circuit delay at the cost of increased circuit leakage.

Adaptive body biasing (ABB) can be used to compensate for both intra-die and inter-die process variations [71]. For dies with frequency higher than the target frequency ( $f_{target}$ ), dies are

reverse body biased to save on leakage power. Similarly, slow devices are FBBed to satisfy the delay constraints. ABB reduces the variations in the target frequency by moving the operating frequency of the slow dies to the right, and fast dies to the left, as shown in Figure 13. It is shown in [71] that as the technology scales, junction tunneling leakage increases under RBB. Therefore, RBB is losing its effectiveness in reducing leakage current by approximately 4x in every technology generation.

The supply voltage scaling, which is a very effective technique for power savings, is also explored as a tool to reduce the variability in circuit delay and power. It is shown in [73] that adaptive supply voltage scaling is as effective as ABB for reducing the circuit delay and power variation. However, it is also shown that applying these two techniques together does not result in significant improvement in performance variability. A leakage sensor and variable size keeper for dynamic logics was proposed in [74][75]. These sensors are placed in different regions of a die to estimate the leakage current. Leakage information is used to choose the best keeper size for the dynamic gates in that region that minimizes the leakage current.



**Figure 13: Variation reduction using ABB.**

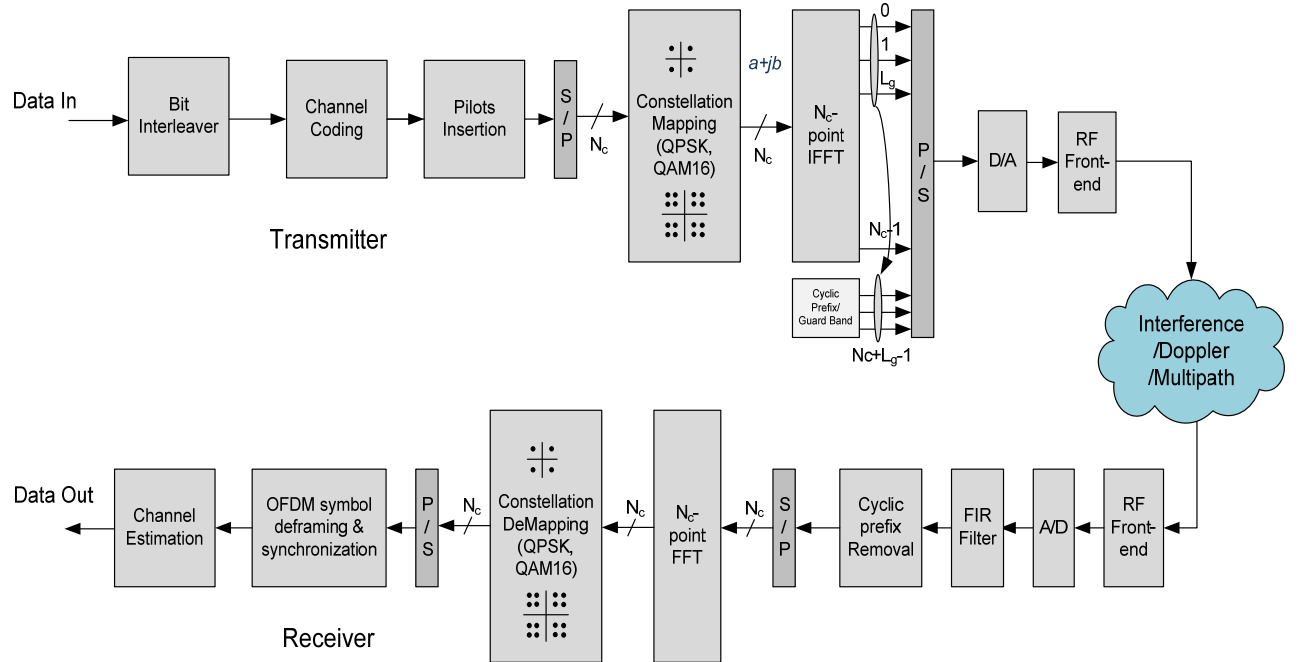
## **CHAPTER 3**

### **SYSTEM DESIGN MODELING**

In this chapter, the fundamentals of the orthogonal frequency division multiplexing (OFDM) baseband transceiver (TRX) are presented. The OFDM TRX is used as the test platform for the robust, low-power, and variation tolerant system level techniques presented in this work. The simulation model and hardware implementation of the OFDM transceiver (as used in this work) is also summarized. In recent years, a great deal of interest has been shown in the OFDM modulation method because of its high spectral efficiency and ability to cope with high-attenuation channels without the need of complex-equalization filters. The OFDM method has become a fundamental scheme in wideband digital communication over wireless and copper medium and is used in applications such as digital video broadcast, digital audio broadcast, wireless internet (WiFi), and mobile networking (WiMax, LTE).

OFDM is a multi-carrier modulation scheme in which a large number of closely spaced orthogonal sub-carriers are used to carry data. Data is divided into several channels and is modulated on orthogonal sub-carriers (one sub-carrier per channel) using modulation schemes such as BPSK, QPSK, QAM-16, etc. The block-level implementation of an OFDM system is shown in Figure 14. In the OFDM transmitter, incoming serial data is optionally bit interleaved and channel coded. Pilots (known data), which are used for the timing and frequency synchronization of an OFDM frame, are inserted into the main data stream. A serial-to-parallel converter (S/P) takes the input-data stream and splits it into  $N_c$  parallel streams, where  $N_c$  is the number of sub-carriers. The data rate of each of these streams is  $1/N_c$  times the original data rate. These parallel streams of bits are mapped to complex-valued symbols ( $S_n=0,1,...,N_c-1$ ) in the constellation-mapping block. One, two, four or more bits are mapped to a complex-valued

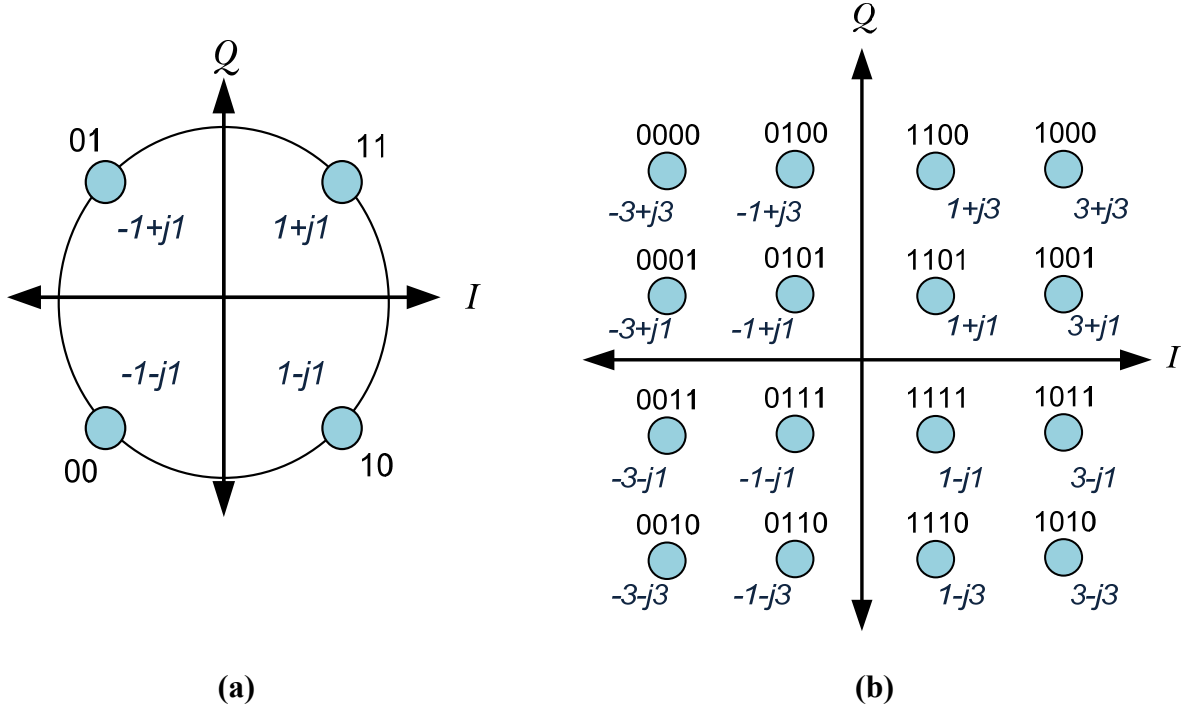
symbol depending upon the modulation scheme (BPSK, QPSK, QAM16, etc.), as shown in Figure 15. The IFFT takes the complex-valued symbols representing  $N_c$  frequencies as input and modulates them into a time-domain signal. The  $N_c$  parallel-modulated source symbols at the output of the IFFT block are referred to as an OFDM symbol. In the OFDM scheme, a cyclic prefix, also known as the guard interval, is added to every OFDM symbol to minimize the effects of inter-symbol interference (ISI) and inter-channel interference (ICI). This data is then passed through a digital-to-analog converter (D/A), up converted, and the resultant RF-signal is transmitted in the channel.



**Figure 14: Block diagram of an OFDM baseband transceiver.**

At the receiver, the incoming data is down converted and is digitized by an analog-to-digital converter (A/D). A low-pass FIR filter is used for decimation and high-frequency noise filtering. The cyclic prefix is removed from the data and a serial-to-parallel converter splits the digitized data into  $N_c$  streams. The FFT block demodulates the incoming time-domain signal symbols into

$N_c$  sub-carriers representing the pilots and data. The pilot symbols are extracted from the OFDM frame for synchronization purposes and the de-framed OFDM symbol is then mapped to bits (depending upon the modulation scheme). These data bits are then processed by the higher layers in the protocol stack.



**Figure 15: Bits are mapped to complex numbers representing amplitude and phase (a) QPSK modulation (b) QAM-16 modulation.**

In OFDM, the sub-carrier frequencies are orthogonal to each other. The orthogonality of the sub-carriers allows close placement of the sub-carriers in a given spectrum, thus ensuring efficient usage of a given bandwidth. The spacing between the sub-carrier frequencies is

$$F_s = \frac{1}{T_s}, \quad (6)$$

where  $T_s$  is the OFDM symbol period. For the  $N_c$  sub-carriers in a system, the source symbol duration ( $T_d$ ) before serial-to-parallel conversion is

$$T_d = \frac{T_s}{N_c}. \quad (7)$$

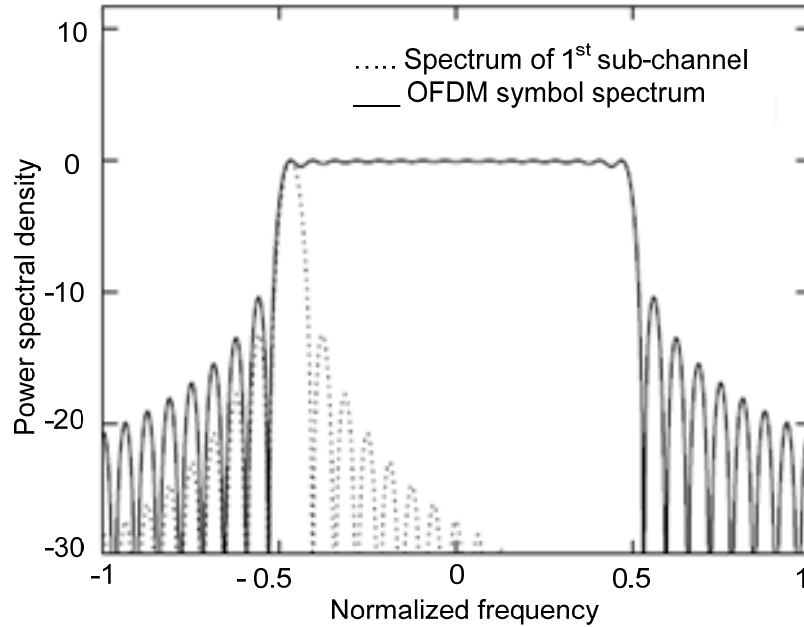
The envelope of an OFDM symbol with rectangular pulse shaping has the form

$$x(t) = \frac{1}{N_c} \sum_{n=0}^{N_c-1} S_n e^{j2\pi f_n t}, 0 \leq t \leq T_s.$$

The  $N_c$  sub-carrier frequencies are located at

$$f_n = \frac{n}{T_s}, \quad n = 0, 1, \dots, N_c - 1.$$

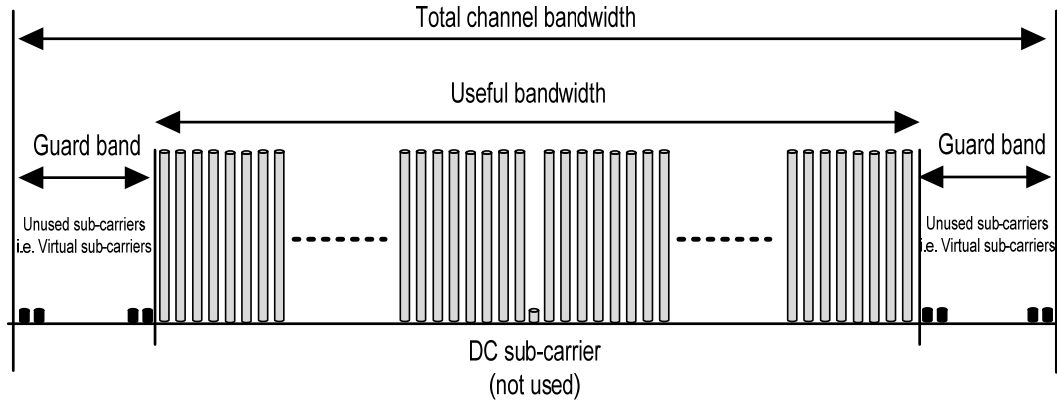
The power spectrum of an OFDM symbol versus the normalized frequency is shown in Figure 16. The dotted curve illustrates the spectrum of the first sub-carrier and the solid line indicates the power spectrum of an OFDM symbol as a sum of the individual power spectrums of the  $N_c$  sub-carriers; each sub-carrier is spaced apart by a frequency of  $F_s$ . Only channels at the band edges contribute to the out-of-band power emission.



**Figure 16: OFDM symbol power spectrum [83].**

To minimize the out-of-band noise, a straight forward method is to use an FFT of higher size than required for the data transmission and to insert null sub-carriers (virtual sub-carriers/guard

band) at the edges of the band spectrum (see Figure 17). Also, a null sub-carrier is placed in the middle of the spectrum to avoid the DC problem.



**Figure 17: Virtual sub-carriers (null sub-carriers) for the out-of-band noise filtering.**

### 3.1 OFDM TRX Power Consumption

Figure 18 shows typical power consumption in an OFDM TRX. On the transmitter side, the power amplifier (PA) is the most power consuming device, followed by the digital signal processor (DSP). On the receiver side, the receiver DSP along with the forward-error-correction block (FEC) consumes roughly 60% of the total receiver power. Table 4 summarizes the power consumed in different WLAN access cards under different modes of operation. It is obvious from the table that more power (energy per unit time) is consumed in the transmitter than in the receiver. However, in typical applications, the transmitter is active only 12% of the time as compared to the receiver. Therefore, a TRX ends up consuming more energy in the receiver than in the transmitter. Hence, significant energy savings can be achieved by applying power saving techniques in the receiver block.



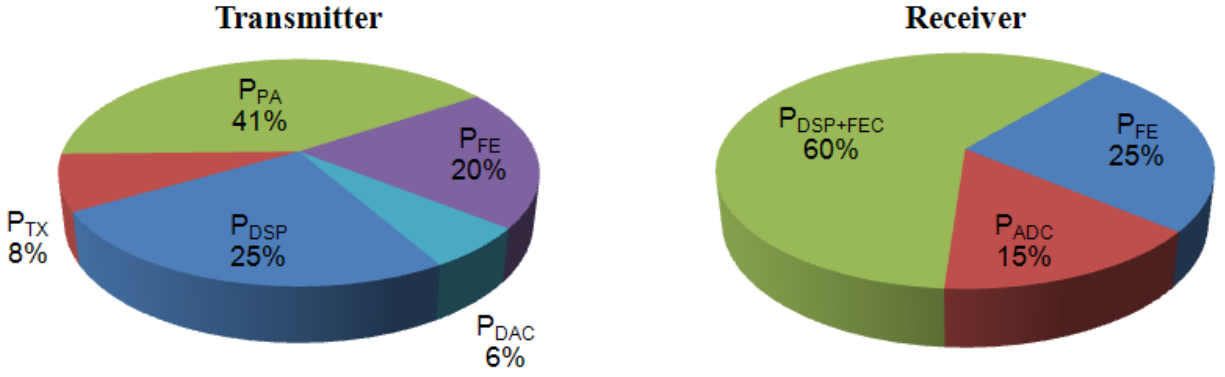


Figure 18: Power distribution in state of the art WLAN transceivers [84].

Table 4: WLAN access card power consumption in different protocols [84].

Mode	802.11b	802.11a	802.11g
Sleep	132 mW	132 mW	132 mW
Idle	544 mW	990 mW	990 mW
Receive	726 mW	1320 mW	1320 mW
Transmit	1089 mW	1815 mW	1980 mW

## 3.2 Simulation Setup

An OFDM baseband TRX is used as a test vehicle for the robust low-power signal processing techniques presented in the following chapters. Two important aspects of the simulation setup are channel modeling and baseband TRX modeling as described next.

### 3.2.1 Channel Modeling

For realistic wireless channel modeling, three major effects related to the OFDM wave propagation are explained below:

- Propagation losses are the incurred attenuations in the radio waves as they travel through the medium. These losses are modeled by simply attenuating the radio signal and adding a white noise to the signal.

- Multipath and fading effects are experienced because of deflections of the radio waves from different obstacles. These effects are modeled using a FIR filter. The length of the FIR filter defines the maximum delay spread in the channel.
- Interference in the channel is modeled as a combined effect of microwave and adjacent channel interference. Adjacent channel interference is due to out-of-band power emission of the neighboring bands and affects the carrier modulated signal. Its effect is modeled as

$$IntAdj(t) = f\{A(t), f_c(t)\}_{OFDM},$$

where  $A(t)$  is the time-varying amplitude and  $f_c(t)$  is the carrier frequency of the adjacent channel interferer. The microwave interferer is modeled as an AM-FM source based on the work presented in [85]. The AM-FM modulation is employed on a time-varying sinusoidal signal to generate the microwave interferer. The frequency of the microwave interferer is given by

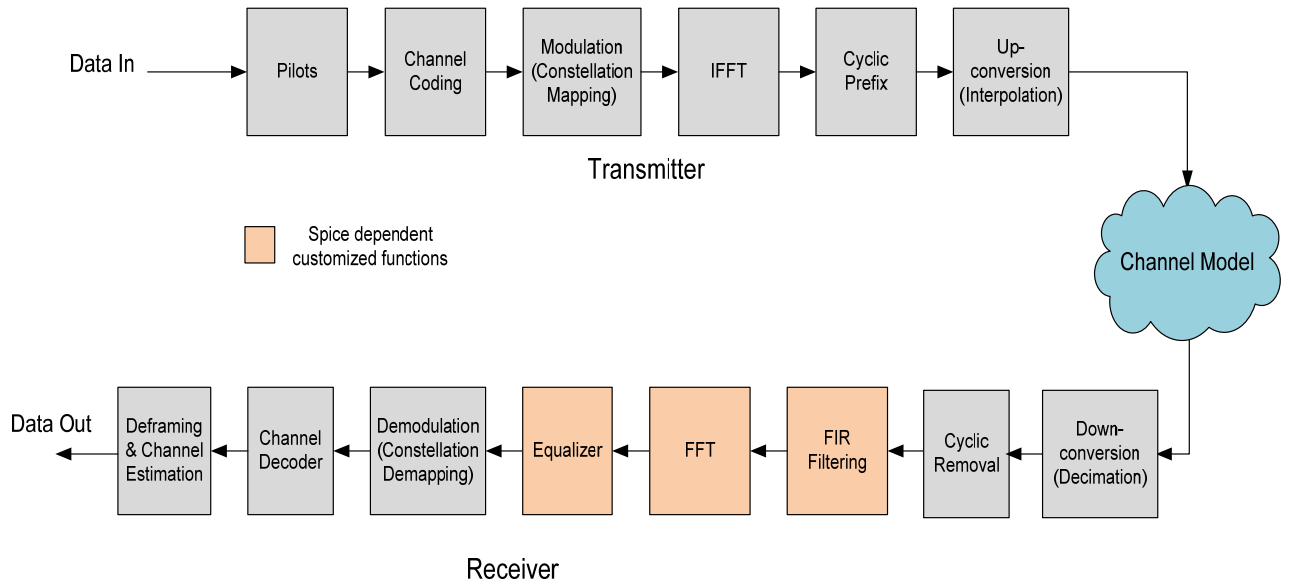
$$f_d(t) = f_o + f_w \sin\left(\frac{2\pi t}{T_w}\right),$$

where  $f_o$  is the initial interferer frequency,  $f_w$  is the maximum frequency wander, and  $T_w$  is the frequency wander period. For this work,  $f_o$  is 2.412GHz,  $f_w$  is 20MHz, and  $T_w$  is chosen as 20ms.

### 3.2.2 Baseband TRX Model

The OFDM based transceiver model (as shown in Figure 19) is implemented in MATLAB for simulation purposes. At the transmitter side, data encoding, modulation, and IFFT are implemented in floating point units. Channel modeling, as discussed in the previous section, is used to model 14 varying channel conditions from good to bad by changing propagation losses, interference, and multipath effects.

In the receiver demodulator, customized HSpice-dependent functions are implemented to model the FIR filter, FFT, and MMSE equalizer. For accurate performance modeling under voltage overscaling, modules are realized at bit-slice level in HSpice using 65 nm CMOS libraries. The path delay and power consumption measures from the bit-slice implementation of a module are used in Matlab function models to estimate the module performance under different voltage levels (1-Volt to 0.55-Volt in voltage steps of 1 mV). All the receiver modules are implemented in fixed point Q2.10 binary format and negative numbers are in 2's complement format. Power consumed in the system is estimated using a statistical power estimation technique (McPower) [86].



**Figure 19: OFDM transceiver model.**

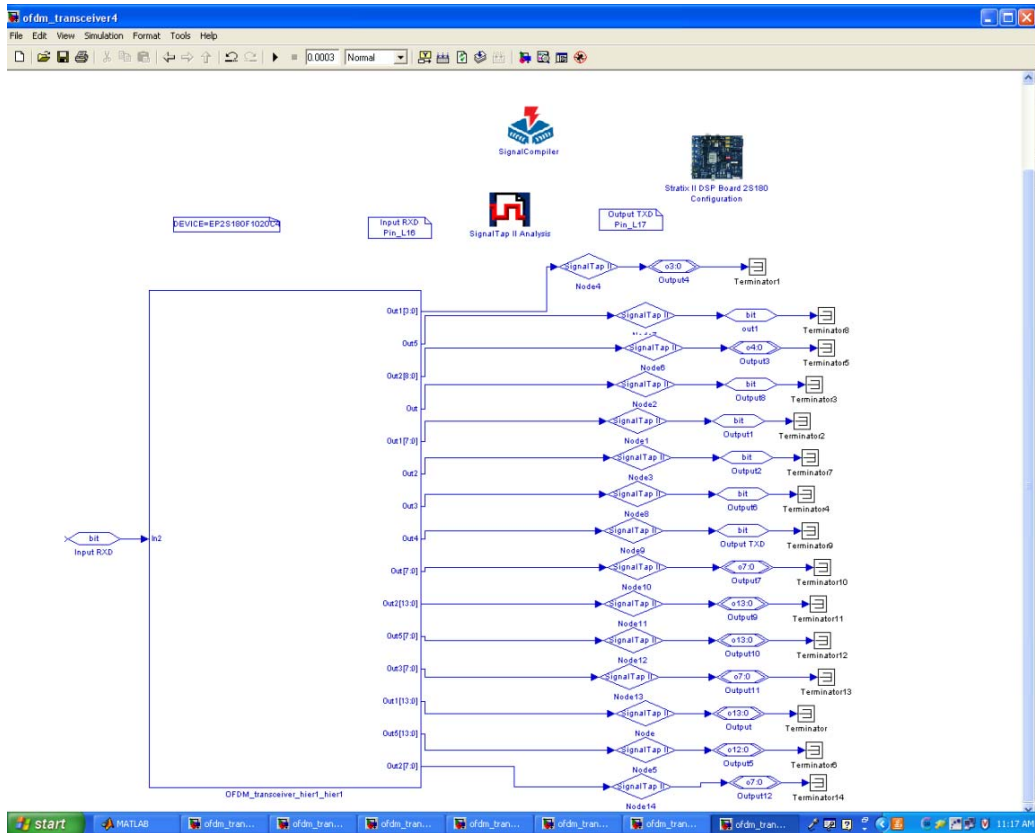
### 3.3 FPGA Implementation

The OFDM baseband TRX model, as shown in the simulation section, is also implemented in an Altera Stratix II DSP development board (see Figure 20) using Quartus II and DSP builder toolkit. Hardware implementation of the TRX model is performed to verify the simulation results. The Altera Stratix II DSP development kit provides high-speed, high-resolution analog

I/O's (DAC, ADC, VGA, Audio), digital I/O's (RS-232 serial port, Ethernet, etc.), expandable memory, and high number of DSP slices in Stratix II device for rapid prototyping of the OFDM baseband model in the FPGA. In our implementation, the DSP development board is connected to a personal computer (PC) through the serial port. Real-time data is captured in the PC using a webcam, which is then transmitted to the FPGA through the serial port. In the FPGA, data passes through the OFDM transmitter and is converted to an analog signal by DAC. DAC data is looped back through an ADC and is processed by the OFDM receiver implemented in the FPGA. The OFDM receiver data is then transferred back to the PC through the serial port and is displayed on the PC using a MATLAB GUI. A snapshot of the OFDM TRX implemented schematic in Simulink is shown in Figure 21.



**Figure 20: Altera Stratix II DSP development board.**



**Figure 21: The OFDM TRX implementation in Simulink.**

# **CHAPTER 4**

## **GUIDED PROBABILISTIC COMPENSATION FOR LOW-POWER FILTERS**

In many DSP applications (voice and image processing) several dBs of SNR loss can be tolerated without noticeable impact on the application level performance. For power optimization in such applications, voltage overscaling (VOS) can be used to operate the arithmetic circuitry at or marginally below the critical circuit path delay while incurring tolerable SNR loss due to the resulting periodic errors in computation. In this work, low-cost checksum codes are used for detection and compensation of intermittent errors due to voltage overscaling in linear digital filters. In traditional coding theory, diagnosis of errors is a key problem and incurs significant computation and latency cost. In the presented method, low-precision shadow latches are used to identify likely sources of errors because of voltage overscaling to avoid error diagnosis. This allows accurate error compensation with distance-2 checksum codes that are normally good only for error detection but not for correction. Very precise compensation is achieved by distributing the negative of the error value evenly across only the likely erroneous states. This is called guided probabilistic compensation, as compensation is not exact when errors occur simultaneously in more than one state. A feedback controller is used for dynamic voltage overscaling (DVOS) while keeping the error rate in the system within an acceptable range. It is shown that the low-cost error compensation allows significant power savings with minimal degradation in the system performance.

## 4.1 Overview

Over the last decade, the technology scaling has allowed large amounts of circuitry to be integrated into smaller areas in silicon resulting in soft error and DSM noise issues, process variability issues and concerns regarding overall power consumption. In particular, the supply voltage must be selected to accommodate worst case noise and process variability conditions, leading to larger than necessary power consumption for the *average* IC. Therefore, it is important to devise new methods for minimizing the design margins incorporated into digital logic to allow ultra low-power operation. In this work, we explore how digital filters can be designed to operate at or marginally below the circuit critical path delay to minimize power consumption. This scheme results in periodic errors that are corrected using low-cost checksum codes and is possible due to the inherent error tolerance of DSP algorithms. As a by-product, each circuit automatically adjusts to the minimum supply voltage necessary to maintain output signal quality (SNR) above a prescribed level using feedback control mechanisms.

There is a considerable work in literature that enables power savings by either dynamic or static voltage scaling because of quadratic relationship of system dynamic power with supply voltage. In practice, voltage scaling is limited by the timing requirements of the critical paths in the circuit. The reduction of supply voltage increases the circuit delay as given in equation 4. Therefore, voltage scaling techniques only reduce supply voltage up to the critical supply voltage ( $V_c$ ) level.  $V_c$  is the minimum supply voltage required to meet the timing limitations of the critical paths in a circuit. Further reduction in supply voltage can be achieved either by throughput reduction or by allowing errors in the system output. In [46][47], a low-power voltage overscaling technique for digital filters is presented, which allows the supply voltage to be reduced below the  $V_c$  level and mitigates the effects of timing violation errors using

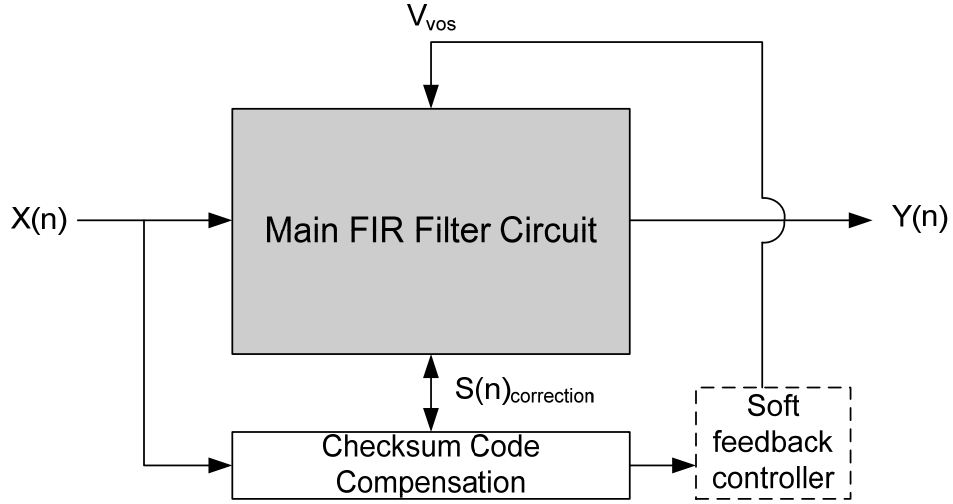
algorithmic noise tolerance (ANT). The ANT techniques include (a) prediction based output approximation and noise error cancellation [46], which uses a reduced length linear predictor to estimate the current output sample of the system such that it replaces the current output with predicted output in case of an error; (b) reduced precision redundancy [48], which compares the output of a reduced precision block with the main filter to detect and correct errors; (c) adaptive error cancellation [49], which tries to mitigate errors in a minimum mean square sense and (d) minimum power soft error correction [50], which uses a linear estimator followed by a maximum-likelihood detector to estimate and correct soft errors. In previous work, it is observed that when the circuit logic delay is marginally longer than the critical path delay of the circuit (due to supply voltage reduction), the resulting logic error rate increases marginally due to the fact that the circuit critical paths are excited infrequently by the applied stimulus [51]. As the supply voltage is decreased further, this error rate increases rapidly leading to a large deterioration in output signal quality.

In the proposed method, checksum codes are used for the first time to optimize the system power. One of the key limitations to widespread use of coding techniques for reliable on-chip computing is the data and circuit redundancy cost necessary for implementing the coding technique. Theoretically, a code of distance  $t+1$  is necessary to detect  $t$  errors and a code of distance  $2t+1$  is required to correct  $t$  errors. In the past, real number checksum codes have been used successfully for error detection in DSP applications. However, error correction is a harder problem and requires significant additional hardware and computation time for error recovery. This makes it difficult to implement error correction without having significant impact on system area overhead and throughput. In prior work [96], checksum codes have been used to probabilistically compensate for soft errors in linear digital filters with the objective of improving signal-to-noise ratio (SNR) in a system, while incurring minimum area and power



overheads. However, the performance of the probabilistic error compensation scheme is unsatisfactory for the errors induced by voltage overscaling, as voltage overscaling causes errors in the MSB's, which causes large magnitude errors at the system output. In this research, the supply voltage is dynamically overscaled to cause periodic errors in the DSP computation. The proposed method is implemented on a linear digital filter. Shadow latches are appended to the selected flip-flops in a system to detect the erroneous states under voltage overscaling. A mismatch between a flip-flop and its corresponding shadow-latch, flags an error associated with the system state. Since the erroneous states (or state) are known, the problem of error diagnosis is automatically resolved, allowing error compensation via the use of distance-two checksum-codes that are normally good only for system error detection but not for error correction. If more than one state is simultaneously erroneous, the negative of the checksum error value is distributed evenly across the detected erroneous states only, as opposed to all the system states. Since the error compensation procedure is guided by the shadow latches, the procedure is called *guided error compensation*. In addition, the system error rate is monitored and a feedback controller continuously adjusts the system supply voltage to maintain the error rate below a specified critical value to satisfy output signal quality (SNR) specifications.

The framework of the proposed dynamic voltage overscaling (DVOS) scheme is shown in Figure 22. While the transient errors affect the main filter block under DVOS, the checksum compensation block remains error free due to its reduced complexity. The magnitude and the frequency of errors in a filter depend on the component architecture (adder, multiplier, register) and the input statistics. To adjust the system performance with the changing signal quality requirements and the input statistics, a feedback controller is implemented. In the feedback mechanism, a proportional or a PID controller dynamically increases or decreases the voltage of the main filter block with high or low transient error rate in the system.



**Figure 22: Proposed linear checksum based voltage overscaling scheme.**

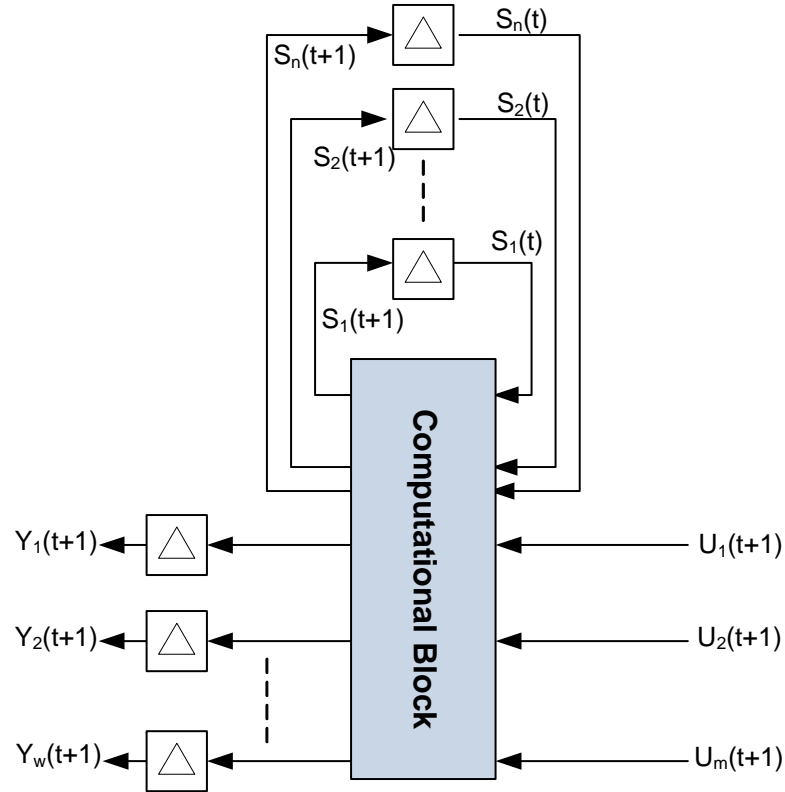
In the next section, state variable system representation, which can be used for digital filter implementation is explained. This is followed by the summary of error detection and compensation schemes. Next, shadow latches used for erroneous state diagnosis are presented and the complete guided probabilistic compensation methodology is explained.

## 4.2 State Variable System Representation

The digital state variable systems considered in this work are interconnections of adders, multipliers, shifters, and registers. The generic data flow in such a system is indicated in Figure 23. Let  $U_1, U_2, \dots, U_m$  be the primary inputs to the synchronous sequential block (computational block). For a system with  $S_1(t), S_2(t), \dots, S_n(t)$  states,  $n+m$  data words combine to produce  $w$  primary outputs as represented by  $Y_1, Y_2, \dots, Y_w$ . In a state variable system, next output  $Y(t+1)$  and the state vector  $S(t+1)$  at time  $t+1$  are related to the present state  $S(t)$  and the system input vector  $U(t)$  as given below:

$$\begin{aligned} S(t+1) &= As(t) + Bu(t) \\ Y(t+1) &= Cs(t) + Du(t) \end{aligned} \quad , \quad (8)$$

where  $A, B, C$  and  $D$  represent the arithmetic operations performed on  $m$  primary inputs  $U(t)$  and current states  $S(t)$  to generate  $w$  primary outputs  $Y(t)$ , and the next states  $S(t+1)$ .



**Figure 23: State variable system representation.**

In general, a computational block module, i.e., an operator can feed more than one state or one output of the system. Such an implementation in which the computational trees of different states and outputs are not disjoint is called shared implementation. An error in an operator in shared implementation can propagate to different states or outputs thus causing multiple errors. Challenges of the error detection and compensation in such a system are explained later in the chapter. In state variable system, soft errors can occur in computational block, system states and/or system outputs. An error in system output disappears after one clock cycle. However, an error in the system state or computational block stays in the system for several clock cycles.

Therefore, this work focuses on error detection and correction in computational block and system states only and a unified approach is presented to handle such errors.

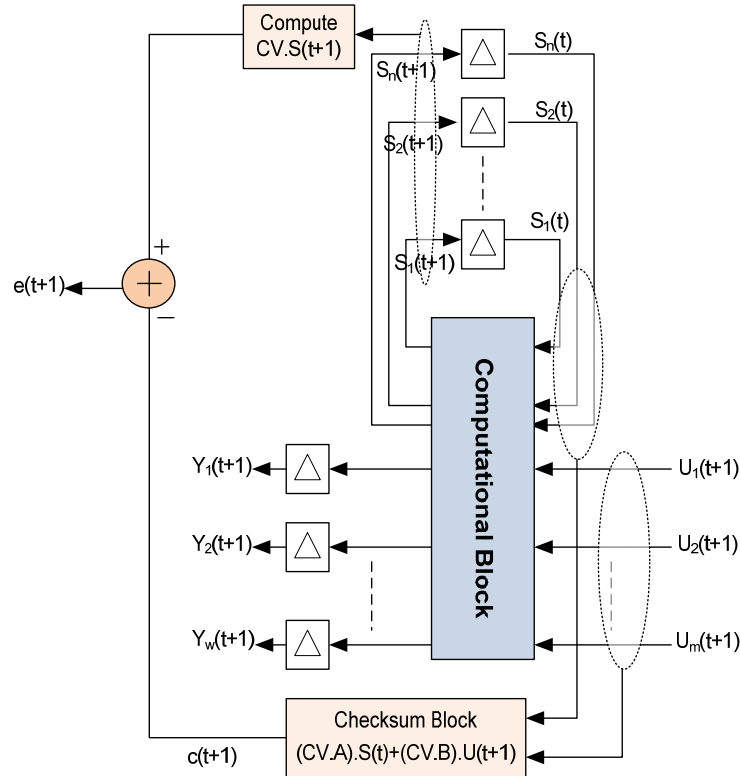
### 4.3 Linear Digital System and Checksum-based Error Detection

In state variable systems, real number codes [97] can be implemented to encode the state vector  $S(t)$ , using one or more check variables. These check variables can be used for error detection and compensation purposes. Each row  $i$  of the matrices  $A$  and  $B$  is scaled using a real value weight  $\alpha_i$  and the scaled rows of  $A$  and  $B$  are summed to generated vectors  $X$  and  $Y$ . Let the coding vector be the vector with relevant weights for each row i.e.  $CV = [\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n]$  for  $n$  number of rows. Then the encoded matrices  $A$  and  $B$  in the form of vectors  $X$  and  $Y$  are;  $X = CV.A$  and  $Y = CV.B$ . A check variable  $c$ , corresponding to each coding vector is computed as  $c(t+1) = X.S(t)^T + Y.U(t)^T$ . In case there is no error in the system then  $c(t+1) = CV.S(t+1)$ . Hence, an error signal  $e(t+1)$  is computed as  $e(t+1) = CV.S(t+1) - c(t+1)$ , which is zero for no error in the system. Figure 24 shows the matrix representation of the procedure.

Figure 25 shows the checksum implementation in a state variable system. In the presence of multiple erroneous states (caused by a shared operator feeding more than one state), the error signal might be zero. In [98], conditions for avoiding such an error aliasing are presented. A brief review is presented in next section.

$$\begin{aligned}
 [Y(t+1)] &= \underbrace{[C(1,1) \ C(1,2) \ \dots \ C(1,n)]}_{\text{State Output Matrix}} \begin{bmatrix} S_1(t) \\ S_2(t) \\ \vdots \\ S_n(t) \end{bmatrix} + \underbrace{\begin{bmatrix} D(1,1) \\ D(2,1) \\ \vdots \\ D(n,1) \end{bmatrix}}_{\text{Output Matrix}} [U(t)] \\
 \begin{bmatrix} S_1(t+1) \\ S_2(t+1) \\ \vdots \\ S_n(t+1) \end{bmatrix} &= \underbrace{\begin{bmatrix} A(1,1) & A(1,2) & \dots & A(1,n) \\ \vdots & \ddots & \ddots & \vdots \\ A(n,1) & A(n,2) & \dots & A(n,n) \end{bmatrix}}_{\text{State Matrix}} \begin{bmatrix} S_1(t) \\ S_2(t) \\ \vdots \\ S_n(t) \end{bmatrix} + \underbrace{\begin{bmatrix} B(1,1) \\ B(2,1) \\ \vdots \\ B(n,1) \end{bmatrix}}_{\text{Input Matrix}} [U(t)] \\
 \text{CV.S}(t+1) &\rightarrow \text{Error}(t+1) = \text{CV.S}(t+1) - c(t+1) \\
 [c(t+1)] &= \underbrace{[CV(1,1) \ CV(1,2) \ \dots \ CV(1,n)]}_X \begin{bmatrix} S_1(t) \\ S_2(t) \\ \vdots \\ S_n(t) \end{bmatrix} + \underbrace{[CV(n,1)]}_Y [U(t)]
 \end{aligned}$$

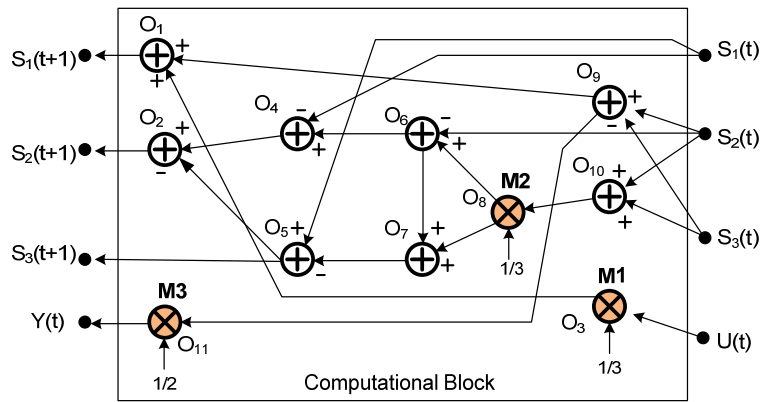
**Figure 24: Linear state matrix representation with a checksum code.**



**Figure 25: State variable form representation with checksum based error detection.**

### 4.3.1 Operator and State Gain – Gain Matrix

As mentioned earlier, a single operator may feed multiple system states or outputs in a shared hardware implementation. In such a scenario, the gain of an operator quantifies how an error in the operator affects different system states. To determine, how an error in an operator  $O_j$  affects the  $i^{\text{th}}$  state,  $S_i$ , first we find all the paths,  $P_i$ , from the output of  $O_j$  to  $S_i$ . For each such path, we define the gain,  $\Theta_i$ , to be the product of the gains of all the operators on that path. The gain of an adder (subtractor) is +1 from its “+” input and -1 from its “-” input. The gain of multiplier is the multiplication constant. Let  $g_{ij}$ , the total gain from  $O_j$  to  $S_i$  be  $\sum_{i=1:P} \Theta_i$ , where  $P$  is the number of paths existing from the output of  $O_j$  to  $S_i$ .  $g_{ij}$  effectively represents the amount by which an error  $\varepsilon_j$  at the output of operator  $O_j$  is scaled before being added to the value of state  $s_i$ . In another words, an error  $\varepsilon_j$  in  $O_j$  causes an error  $g_{ij} \times \varepsilon_j$  in  $s_i$ . For example for the system shown in Figure 26, the gain of path 8, 6, 4, 2, S<sub>2</sub> from  $O_{10}$  to  $S_2$  is  $(1/3)(+1)(+1)(+1) = 1/3$  and  $g_{2,10} = (1/3)(1)(1)(1) + (1/3)(1)(-1)(-1) + (1/3)(1)(1)(-1)(-1) = 1$ .



**Figure 26: Structure of a state-variable system with shared operators.**

Similar to the gain of an operator, gain of a state is defined as

$$\text{Gain of } S_i \text{ on } S_j = g_{ij} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

i.e. the effect of an error in a state,  $S_i$ , is zero for all other states except itself. The gain matrix  $GM$  is an  $n \times (k+n)$  matrix where  $n$  is the number of states and  $k$  is the number of operators involved in computing the system states. Let  $N = k+n$  and each column of the gain matrix represents the gain of an operator or a state. Without loss of generality, we construct the gain matrix such that the first  $k$  columns represent the operators gain and the last  $n$  columns represent the gain of states. Figure 27 shows the gain matrix corresponding to the example system of Figure 26. Operator 11 is not included in the gain matrix as it feeds directly to the system output and not the system states.

$$GM = \begin{array}{c} \begin{array}{cccccccccc} \leftarrow & O_1 & \dots & O_{10} & \rightarrow & S_1 & S_2 & S_3 \end{array} \\ \left\{ \begin{array}{cccccccccc|ccc} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -1 & 1 & 1 & 3 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -2 & 0 & -2/3 & 0 & 0 & 1 \end{array} \right\} \\ \begin{array}{cccccccccc} \text{Operators} & & & & & & & & & & \text{States} \end{array} \end{array}$$

**Figure 27: Gain Matrix corresponding to the system shown in Figure 26.**

It is shown in [97] that to guarantee an error being observed on the error signal i.e  $e(t+I)$ , the coding vector must be chosen such that all elements of the product  $CV \times GM$  are non-zeros. A non-zero value of the error signal indicates an error either in states,  $S(t+I)$ , in the check variable,  $c(t+I)$ , or in the error signal,  $e(t+I)$ . To differentiate between errors in the system states or checksum circuitry, two check variables can be used. If more than one check variable is used, it can be concluded that error is restricted to system states if and only if all the check variables are non-zero.

## 4.4 Guided Compensation for Low-Power Operation

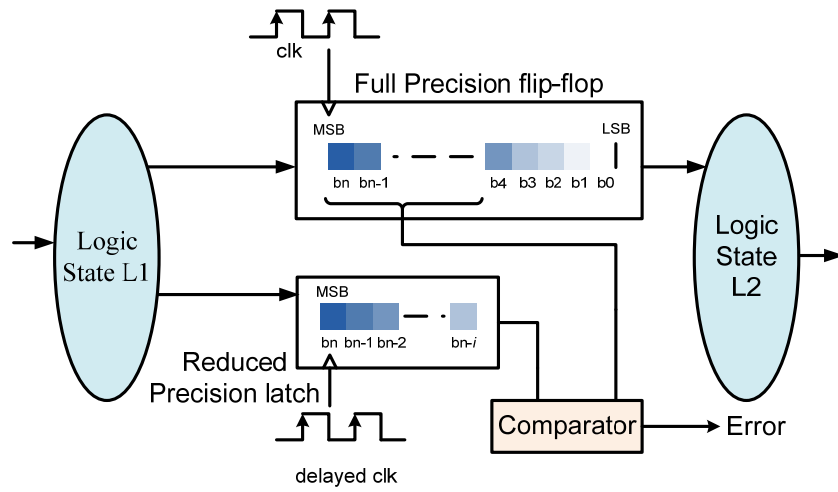
In this section we explain the GPC architecture for low-power operation of filters. The concept of shadow-latches [45] is used for diagnosis of erroneous states and system level error compensation is performed under the assumption that the shadow-latches clearly delineate the erroneous states from the error-free ones. Note that when errors due to voltage overscaling occur, they are detected by the checksum code employed (distance-2 codes are used). However, it is not possible to perform error compensation with these codes because distance-2 codes have only error detection but no error correction capability. The errors detected by the shadow latches point to likely sources of the errors (locations of the corresponding shadow latches, .i.e. error diagnosis). This information is then used to perform error compensation.

### 4.4.1 Shadow-Latches

For error detection in the system states we use low-precision shadow-latches that augment the full precision registers as shown in Figure 28. They are called “low-precision” latches since the set of latches used, cover most significant data bits only and not the lower significant bits (LSBs) of the data and operate at a delayed clock as compared to the main circuit flip-flops. In case of no error in the logical path for system state computation under voltage overscaling, the values in the MSB bits of the main register and shadow-latch will match. However, if the logical path violates the timing requirements under scaled supply voltage for an input, a comparator flags an error due to mismatch between the main flip-flop and corresponding shadow-latch values. To ensure that shadow-latches always latch the correct data, the operating voltage is constrained so that the worst case delay does not exceed the shadow-latch’s setup time. Also, to minimize the power cost because of added latches and comparator, we limit the number of MSB bits to be monitored. The registers (main flip-flops and shadow-latches) can enter meta-stable



state under VOS. To avoid such behavior of registers, we propose either implementation of meta-stable tolerant design of registers or operation of registers at nominal voltage. Operation of registers at nominal voltage does not have a major impact on power savings of the proposed scheme as the number of registers is low in an arithmetic component dominated circuit. However, separate supply voltage routing to the registers is necessary. In this work, meta-stable tolerant registers were used. Moreover, to reduce the cost of clock routing, delayed clocks are generated locally in the circuit. Only seven MSB bits are monitored in our filter implementation as we never reduced the supply voltage below a limit where more than seven MSB bits become erroneous.

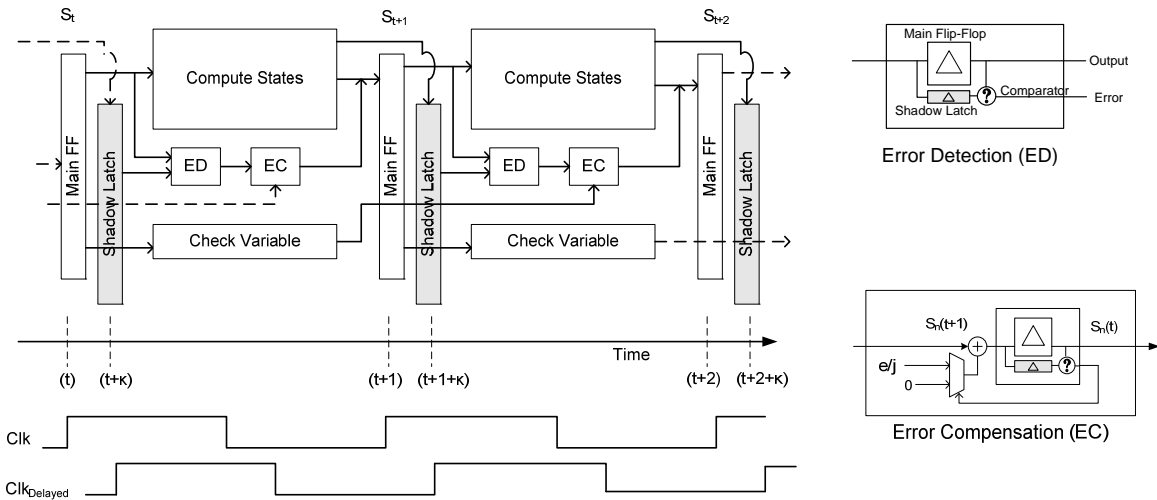


**Figure 28: Reduced precision shadow-latch for error monitoring in MSB bits.**

#### 4.4.2 Guided Error Compensation

The proposed guided probabilistic error compensation methodology is explained in this section with the help of Figure 29 and Figure 30. The GPC methodology detects and compensates for errors induced by voltage overscaling as described in the following steps:

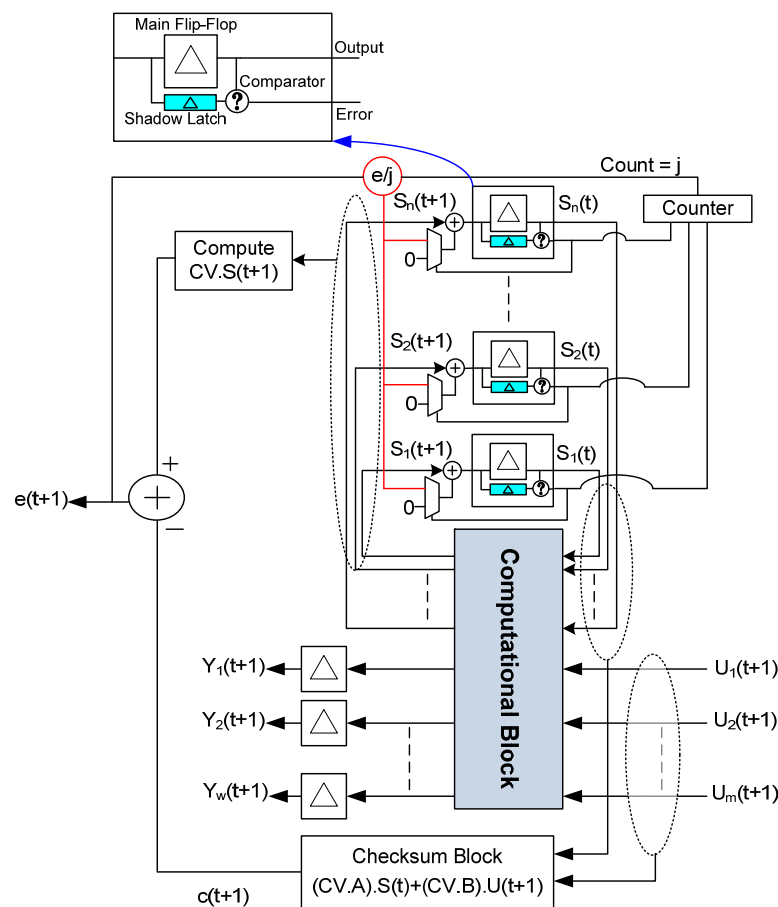
- Under VOS, if the timing requirements of some logic paths are violated in time  $(t, t+1)$  then one or more states will have erroneous data at time  $t+1$  (see Figure 29). The magnitude of system error  $e(t+1)$  is calculated using check variable.
- At time  $t+1+\kappa$ , where  $0 < \kappa < 0.5$ , one or more of the shadow-latches will flag on error due to mismatch of main flip-flop and shadow-latch values.
- Total number of erroneous states is calculated at time instance  $t+1+\kappa$  by a counter. We then divide the error in the system  $e(t+1)$  with the number of erroneous states. If the counter count is  $j$ , then the compensation value for the erroneous states is  $e(t+1)/j$ .
- The error signal of every state is used as the control signal of the multiplexer and selects between a zero or  $e(t+1)/j$  correction value. This correction value is added to the erroneous states at time  $t+2$ .



**Figure 29: Guided probabilistic compensation methodology. Error detection and compensation is performed in the next clock cycle.**

In case when only one state is erroneous then error value is compensated at only that state. If two states are found to be erroneous, then error value is divided by 2 and both erroneous states are compensated with the same error/2 value. No compensation value is added to the states that

were not flagged erroneous by the shadow-latches. The presented guided probabilistic error correction technique only compensates the error at the erroneous states with an overall goal of minimizing the system noise. The advantage of the proposed scheme is that there is no throughput penalty and area overhead is minimal in terms of reduced precision shadow-latches and reduced precision check variable. Error detection, error correction and check variable blocks compute in parallel with the actual computation. However, since the scheme relies on shadow-latches for diagnosis of erroneous states that operate on delayed clock, therefore error compensation is performed in the next clock cycle. This allows propagation of erroneous output for one clock cycle and increases system noise.



**Figure 30: GPC architecture implementation.**

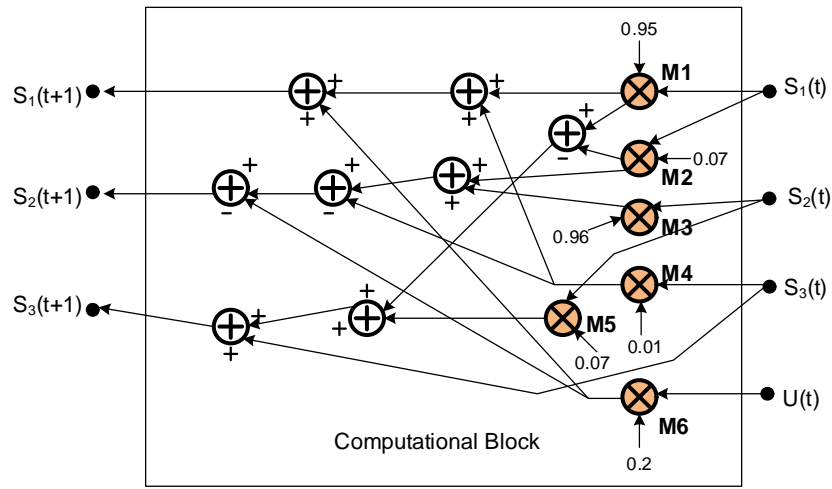
We reiterate that although some errors sneak through the error compensation scheme, the proposed method results in significant output SNR improvement as compared to no error correction, since critical paths in the circuit are not excited all the time [51] and an error in the system if not corrected stays in the system for many clock cycles. Low precision shadow-latches are used to monitor the MSB bits only therefore it is possible in some cases that error  $e(t+1)$  is indicated by the checksum codes but no shadow-latch flags an error due to the error occurring in the bits that were not monitored by the shadow-latches. In that scenario, we use traditional probabilistic compensation technique [96] to reduce the system noise. Figure 30 shows the localized probabilistic compensation architecture in detail.

#### 4.4.3 Guided Error Compensation for Shared Hardware Implementation

Guided error compensation method presented in the previous section works well when the computation trees in the computation block are mutually disjoint (i.e. there are no common adders and multipliers between different trees). However, in general, computation trees need not to be disjoint. Common arithmetic expressions in the linear computation trees can be shared to minimize the hardware required to implement a state variable system. Internal nodes of one computation tree can be tapped to feed the inputs to another tree, resulting in common operators between different trees. Such an implementation is called shared hardware implementation. Shared hardware implementation of state variable system (3-tap elliptical LPF) of equation(9) is given in Figure 31. Six multipliers and eight adders are used for this implementation.

$$\begin{bmatrix} S_1(t+1) \\ S_2(t+1) \\ S_3(t+1) \end{bmatrix} = \begin{bmatrix} 0.95 & 0 & 0.01 \\ 0.07 & 0.96 & -0.01 \\ 0.88 & 0.07 & 1 \end{bmatrix} \begin{bmatrix} S_1(t) \\ S_2(t) \\ S_3(t) \end{bmatrix} + \begin{bmatrix} 0.2 \\ -0.2 \\ 0 \end{bmatrix} [U(t)] \quad (9)$$

Now consider pipelined implementation of the state variable system of Figure 31. Let the clock period time of each pipeline stage equal to worst case time period of the multiplier. Each pipeline stage is appended with shadow latches to capture the time delayed value under voltage overscaling. In such a scenario, MSB error occurring in any multiplier under voltage overscaling will feed to multiple outputs. Therefore it is necessary to not only determine the number of erroneous multiplications but also the fan out of erroneous multipliers to primary outputs.



**Figure 31: Computational block implementation with shared hardware.**

We define this relationship by the fan out matrix “F”, as given below for a system of “m” states and “n-m” multipliers.

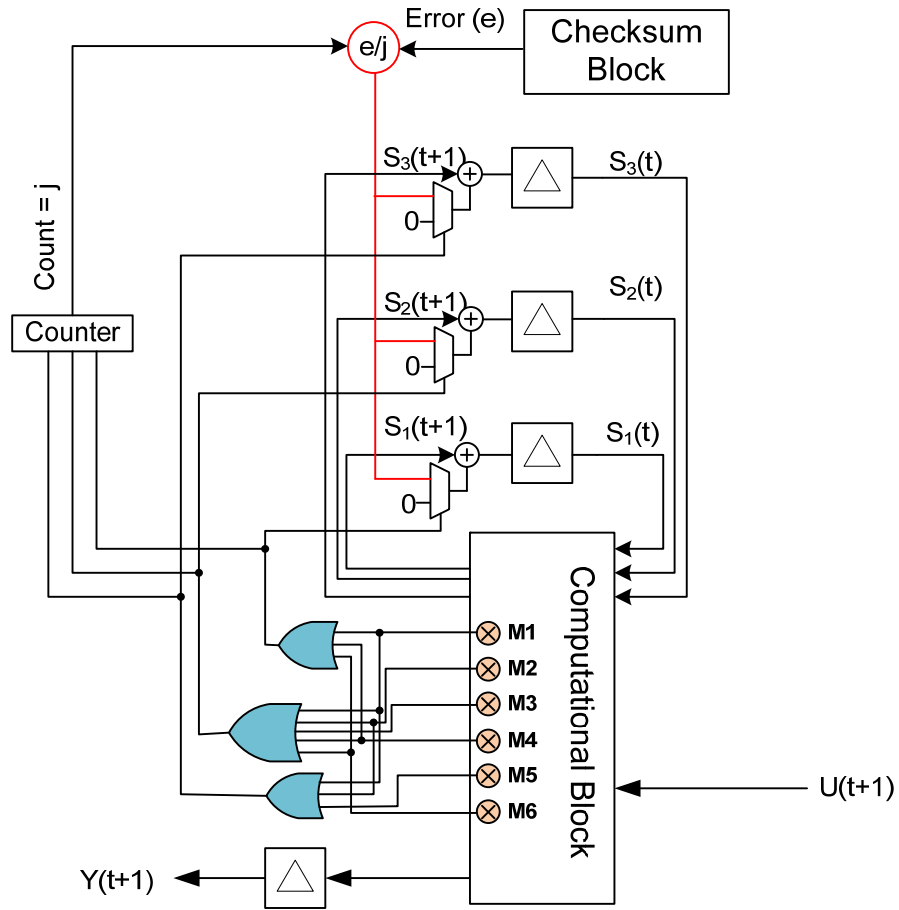
$$F = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & \dots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & \dots & \alpha_{2n} \\ \vdots & & & & \\ \alpha_{m1} & \alpha_{m2} & \alpha_{m3} & \dots & \alpha_{mn} \end{bmatrix} \quad (10)$$

In the fan out matrix F, number of rows represents the number of states in the system, and number of columns represents the sum of number of states and multipliers. For shared hardware implementation of state variable system of Figure 31, fan out matrix F is given below

$$F = \begin{matrix} & \begin{matrix} S_1 & S_2 & S_3 & M1 & M2 & M3 & M4 & M5 & M6 \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix}$$

States
Multipliers

To perform error compensation under shared hardware implementation entries of each row are “ORed”  $(\overset{n}{OR}\alpha_{1i}, \overset{n}{OR}\alpha_{2i} \dots \dots \dots \overset{n}{OR}\alpha_{mi})$  and fed into the enable pin of error compensation multiplexer (MUX) as shown in Figure 32. Only multipliers are shown in the computation block for the sake of clarity in the figure.



**Figure 32: Guided probabilistic compensation in shared-pipelined architecture.**

#### 4.4.4 Low Precision Checksum Effects

To reduce the overhead of the checksum circuitry, low precision checksum codes are implemented. We take advantage of the fact that VOS results in large magnitude errors as errors occur in the MSB bits. The checksum error in the system is calculated as

$$e(t+1) = CV \cdot S(t+1)^T - c(t+1).$$

In order to utilize low precision checksum codes, the checksum block producing  $c(t+1)$  and compute block producing  $CV \cdot S(t+1)^T$  as shown in Figure 30 will be of low precision. In such a case, quantization error ( $Error_{quant.}$ ) in the system is given as

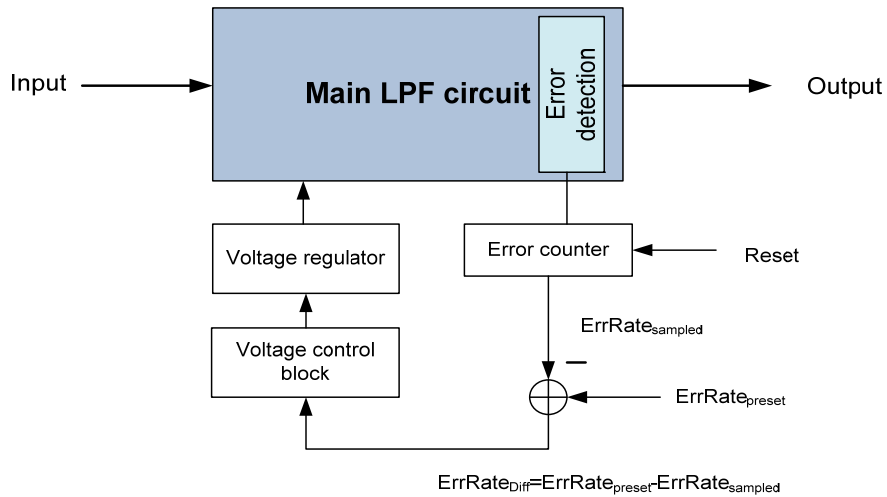
$$Error_{quant.}(t+1) = e_{org}(t+1) - e_{rp}(t+1).$$

In the above equation,  $e_{org}$  is the full precision and  $e_{rp}$  is the reduced precision calculation. The maximum of this error is dependent on the system input and the number of linear operations needed to produce the system output. Higher the number of operations (additions, multiplications) needed higher the quantization error will grow and vice versa. In real applications, a maximum bound can be placed on quantization error using the maximum of the system input and the linear operations of an output. Then in a low precision checksum system, any error less than the maximum bound will be ignored as quantization error. An error in the system will be detected and compensated if and only if it is greater than the maximum bound value. Under such a bound, it is possible to use low precision checksums without significant addition in system noise.

#### 4.4.5 Dynamic Supply Voltage Control

In this section, supply voltage control for the low-power operation of a filter is explained. Dynamic supply voltage control is necessary to adjust the system performance with changing signal quality requirements and input statistics. The feedback control system adjusts the supply

voltage according to the error rate in a system. Error rate is monitored in a system using a counter and it counts the number of errors detected by the error detection block over a certain period of time. If error rate is low then it means that computations in the circuit are completing too quickly (critical paths are not being excited) and/or the error correction is working well enough to lower the supply voltage further to save power. On the other hand, if error rate increases then it means that the filter components are not meeting the timing constraints and error correction is not sufficient to compensate for the number of occurring errors. In that case, feedback control increases the supply voltage to bring the error rate within acceptable range. The error rate in the system will be defined by the overall system performance requirements.



**Figure 33: Supply voltage control.**

Supply voltage control system used in this work is shown in Figure 33. The proposed approach strives to operate the system at a preset error rate ( $ErrRate_{preset}$ ). Error rate in the system is calculated using a counter, which is sampled after a certain time period. Counter is reset to zero after every measure of error rate. Sampled error rate ( $ErrRate_{sampled}$ ) in the system is compared to the preset error rate in a system to calculate the error rate difference ( $ErrRate_{Diff}$ ). This error rate difference value is then used to set the voltage of the main LPF. If the error rate



difference is negative then system is experiencing more number of errors that can be tolerated and therefore supply voltage is increased. Similarly, if the error rate difference is positive than it means the system voltage can be reduced further to take advantage of the low error rate. Voltage adaptation and consequently power optimization by such a control system is strongly dependent on the response time of voltage controller. A fast switching voltage controller can put system into oscillations. On the other hand, a conservative controller design will be slow to respond to changes in system environment. To ensure the stability of the voltage control system, the error rate sample period is set equal to minimum time required to switch between minimum voltage steps.

In this work, proportional (P), proportional-integral (PI) and proportional-integral-differential (PID) controllers are implemented using the Ziegler-Nichols method [91]. The objective of three different controller realizations was to observe the performance advantages and implementation overheads of the methods in our particular application. P-controller is easiest to implement and changes the system voltage in direct proportion to difference in required and sampled error rates. However, such a control system never settles to a steady system voltage even with fixed input statistics. In many practical implementations of filters such as in wireless transceivers, voltage control system can be implemented in the baseband DSP. The implementation of voltage control system in DSP software minimizes the hardware overhead of the proposed scheme and thus results in higher power savings. In this work it is assumed that voltage controller is implemented in soft-form and runs on system DSP.

## **4.5 Simulation Setup**

A low-pass 15-tap filter (LPF) is implemented that filters out frequencies above 70Hz. A sequence of 2000 input samples is considered with input signal sampled at 1 kHz.

Implementation technology is assumed to be 65nm, 1-Volt CMOS process. System precision is set to be 12-bit in Q4.8 fixed-point format. Ripple carry adders and array multipliers are assumed to be implemented in the filter and checksum calculation block. Negative numbers are represented in two's complement form. Minimum voltage step available for VOS is 1 mV. For delay estimation of the critical paths, circuit level simulations are performed in HSpice at varying voltage levels (1-Volt to 0.55-Volt in the voltage steps of 1 mV). These delay estimates are used to observe the path delay errors under VOS in a logic level simulator. The affect of VOS on delay of a full-adder is shown in Table 5.

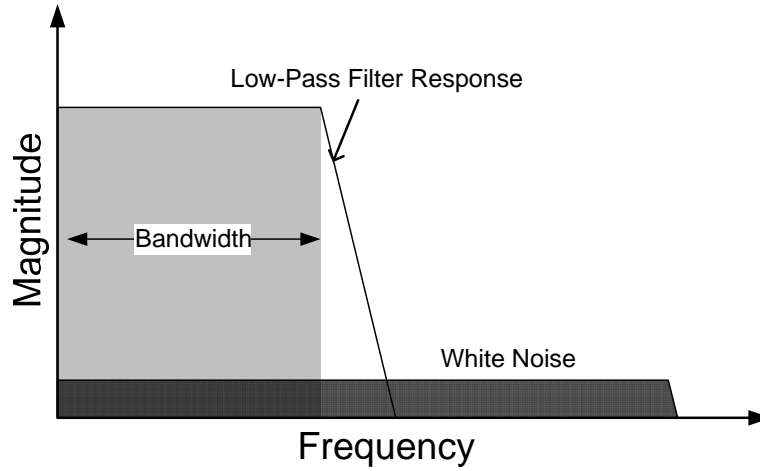
**Table 5: Change in Full-Adder delay with VOS (65nm CMOS technology).**

Vdd (V)	1	0.9	0.8	0.7	0.6	0.5
Delay(psec)	38.1	40.3	44.1	48.8	55.2	63.4

Power savings in the system under VOS are calculated as given below

$$P_{sav} = \frac{P_{org} - P_{gpc}}{P_{org}} .$$

Where,  $P_{org}$  is the power consumed in a LPF without VOS and  $P_{gpc}$  is the power consumed in a LPF fitted with error detection and compensation circuitry under VOS. Power consumed in the system is calculated using a statistical power estimation technique (McPower) [84].



**Figure 34: Frequency spectrum of a low-pass FIR filter.**

Ideal frequency response of a LPF is shown in Figure 34. For the 15-tap LPF used in simulations, growth in error rate and its corresponding affect on system SNR with VOS is shown in Table 6. It can be observed from the table that higher number of MSB bits become error prone with increasing VOS and thus effectively decrease system precision.

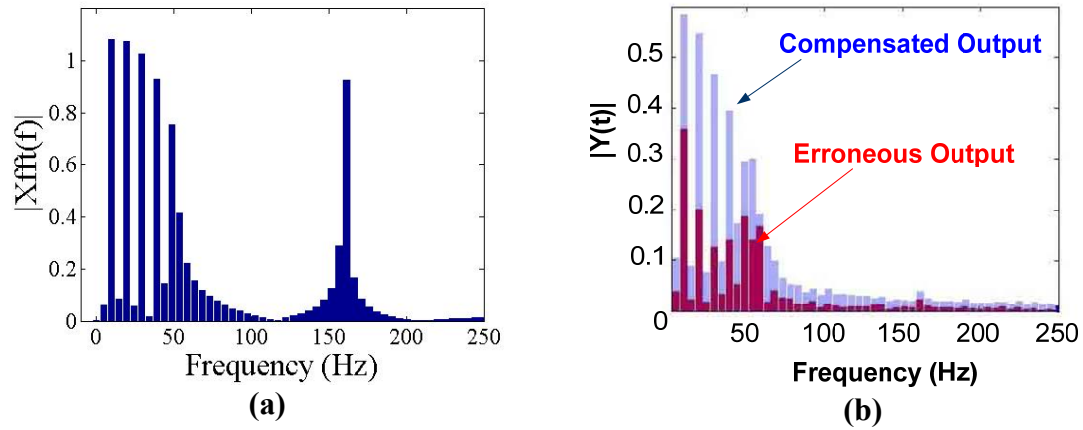
**Table 6: Increase in error rate with VOS in LPF.**

Vvos (Volt)	1	0.90	0.80	0.70	0.60
Sure Bits	12	10	8	7	6
Error rate	0%	2.3%	18.4%	37.7%	61.4%
Output SNR (dB)	23.1	-0.56	-6.8	-13.14	-16.1

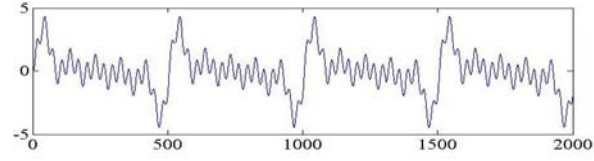
#### 4.5.1 Simulation Results at Fixed Voltages

Frequency response of the filter input is shown in Figure 35(a). Frequency spectrum of the LPF output, with and without GPC is shown in Figure 35 (b). Figure 36 (a) and (b) show LPF input and output waveforms in time domain with 2000 sample points. VOS to 0.85 volts results in erroneous output as shown in Figure 36 (c). As evident from Figure 36 (d), errors are of high

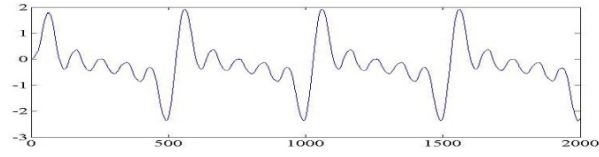
magnitude. LPF output with guided probabilistic compensation is shown in Figure 36 (e). GPC allows some errors to propagate to system output but still results in significant improvement in system output as shown in Figure 36 (e) and (f).



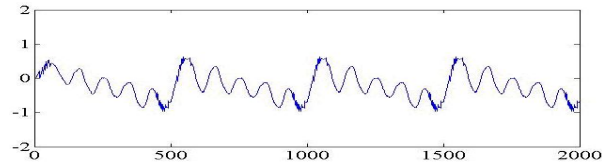
**Figure 35: (a) Frequency spectrum of input applied to the LPF. (b) Frequency spectrum of filter output with and without probabilistic compensation.**



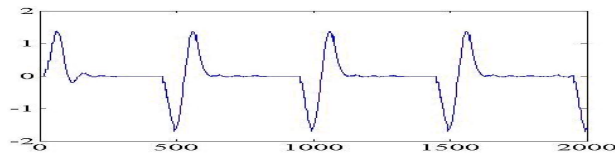
(a) Input signal  $x(n)$



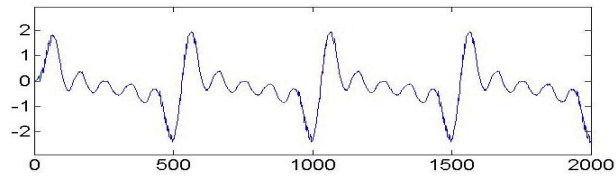
(b) Output signal  $y(n)$  at 1-volt



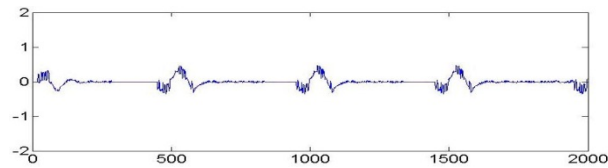
(c) Output signal  $\tilde{y}(n)$  at 0.85-volt



(d) Error signal,  $e(n) = y(n) - \tilde{y}(n)$



(e) GPC compensated output  $y_{\text{gpc}}(n)$

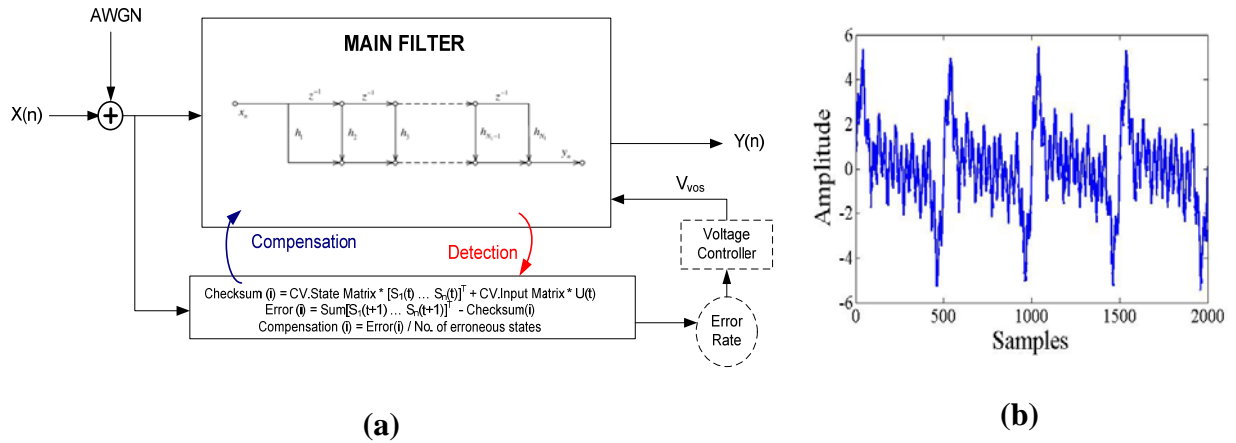


(f) GPC error signal,  $\tilde{e}(n) = y(n) - y_{\text{gpc}}(n)$ .

**Figure 36: LPF input, erroneous and compensated output under VOS in time domain (Y-axis “Amplitude” & X-axis “Sample”).**

### 4.5.2 Simulation Results with DVOS

Figure 37 (a) shows the simulation setup used for DVOS of LPF with the help of an error rate counter and voltage controller. Input signal shown in Figure 36 (a) with white noise added is used as input signal in this part of the simulations and is shown in Figure 37 (b). For voltage controller implementation, minimum voltage step is assumed to be 1mV and time required by voltage regulator to modulate the voltage step is 20 clock cycles. To ensure the stability of voltage control system, error rate is calculated over 20 clock cycles. Error counter is reset after 20 clock cycles in a P-controller. In PI and PID-controller implementations, an error sum accumulator is also used that is reset every 4000 clock cycles. Voltage controller changes the output voltage based on sampled error rate, preset error rate and controller implementation i.e. P, PI or PID. In this work, Ziegler-Nichols method is used as a starting point to determine the control parameter settings of (P, PI, PID) controller implementations and are shown in Table 7. Critical gain ( $K_c$ ) in this method is calculated by setting  $K_i$  and  $K_d$  to zero and by increasing  $K_p$  till the system output starts oscillating.  $P_c$  is the oscillation time period at critical gain ( $K_c$ ).

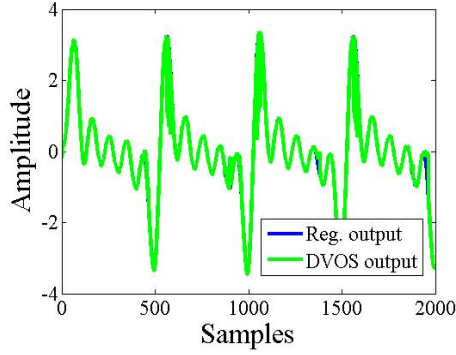


**Figure 37: (a) Simulation setup for DVOS (b) Input to LPF with added AWGN.**

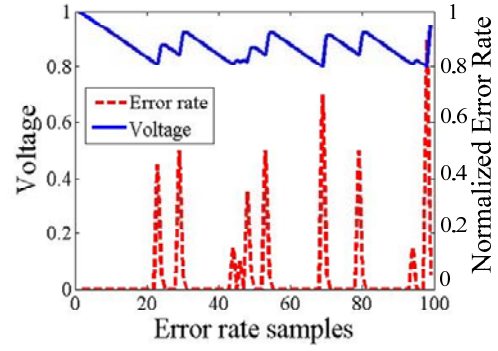
**Table 7: Control parameter settings of feedback controllers calculated using Ziegler-Nichols method [91].**

Critical gain; $K_c = 0.35$	PID Controller	PI Controller	P Controller
Oscillation time period; $P_c = 24$	Proportional control $K_p = 0.6K_c = 0.21$	Proportional control $K_p = 0.45K_c = 0.1575$	Proportional control $K_p = 0.5K_c = 0.175$
	Integral control $K_i = 2K_p/P_c = 0.0175$	Integral control $K_i = 1.2K_p/P_c = 0.0078$	N/A
	Differential control $K_d = K_p P_c / 8 = 0.63$	N/A	N/A

In the first voltage controller realization, a P-controller is implemented. Results of P-controller with preset error rates of 5% and 25% are shown in Figure 38 and Figure 39 respectively. LPF output at default voltage settings and with DVOS is shown in Part (a) of the figures. System voltage settings under DVOS and error rate as a function of time are shown in Part (b) of the figures. As explained earlier, settings of the preset error rate in a system has an important role in defining the overall power savings and the end system performance. Mean voltages of LPF at preset error rates of 5% and 25% are 0.83volts and 0.77volts and output SNR values are 20.175dB and 18.997dB respectively.

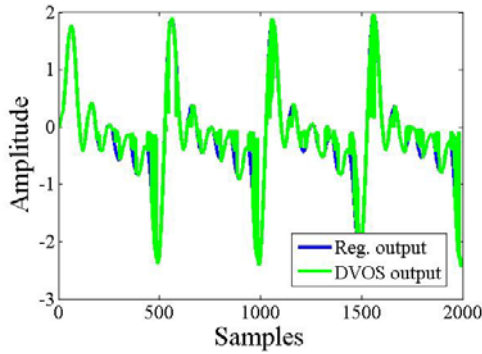


LPF output at regular voltage and with GPC-DVOS.

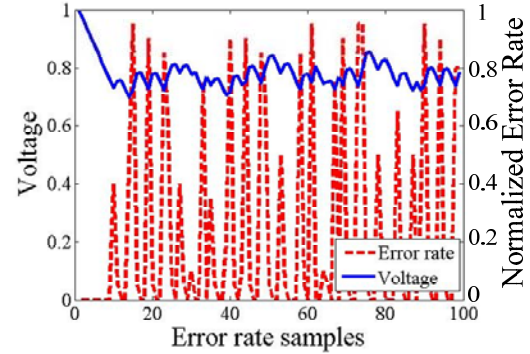


System voltage and error rate with DVOS as a function of time.

**Figure 38: Proportional controller with preset error rate of 5%.**



LPF output at regular voltage and with GPC-DVOS.



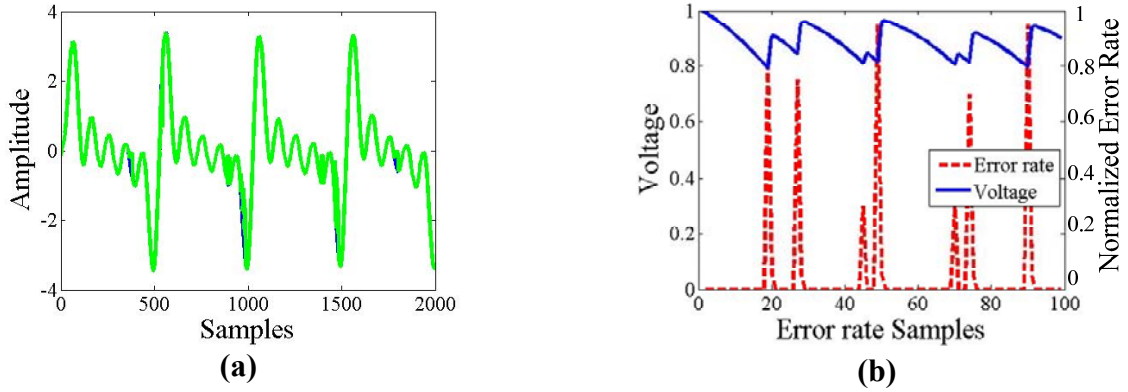
System voltage and error rate with DVOS as a function of time.

**Figure 39: Proportional controller with preset error rate of 25%.**

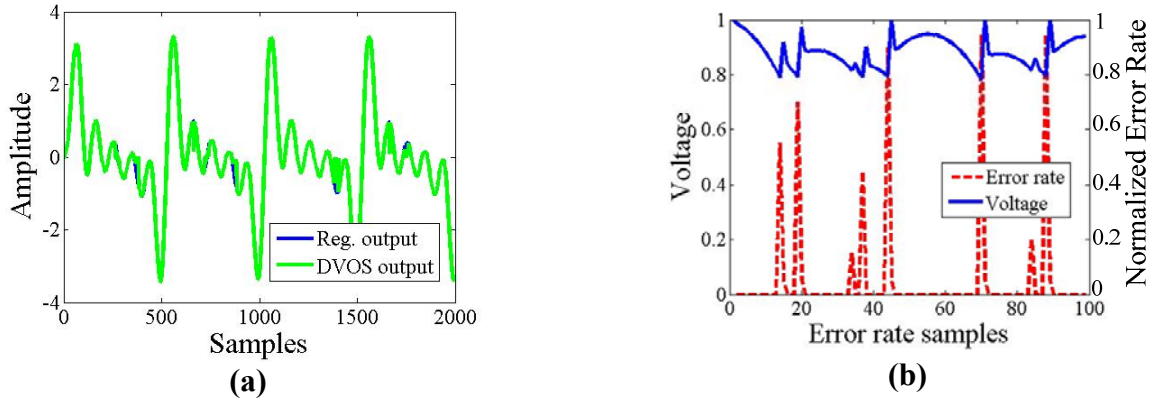
Figure 40 shows the output waveforms and corresponding system voltage and sampled error rates for a PI-controller implementation with a preset error rate of 5%. Mean voltage of LPF comes out to be 0.8893 volts with output SNR of 20.87dB. LPF output waveforms and corresponding voltage settings using a PID-controller for a preset error rate of 5% are shown in Figure 41 (a) and (b) respectively. Mean voltage for the PID-controller is 0.8992 volts with output SNR of 20.67dB. The control parameters of PI and PID-controllers are set according to the values given in



Table 7. It is observed through similar simulation results, which are not presented here to avoid repetition of similar data that P-controller performs reasonably well for our LPF application. The reason being, LPF fitted with GPC can tolerate high error rate as majority of errors are compensated in filter by compensation circuitry. Therefore, voltage oscillations with sampled error rate changes in a P-controller implementation do not have a major impact on LPF performance. Moreover, implementation of P-controller is straight forward with minimum overhead cost as compared to PI and PID-controllers.



**Figure 40: PI-controller, preset error rate of 5% (a) LPF regular and DVOS output (b) Voltage and sampled error rate of LPF.**

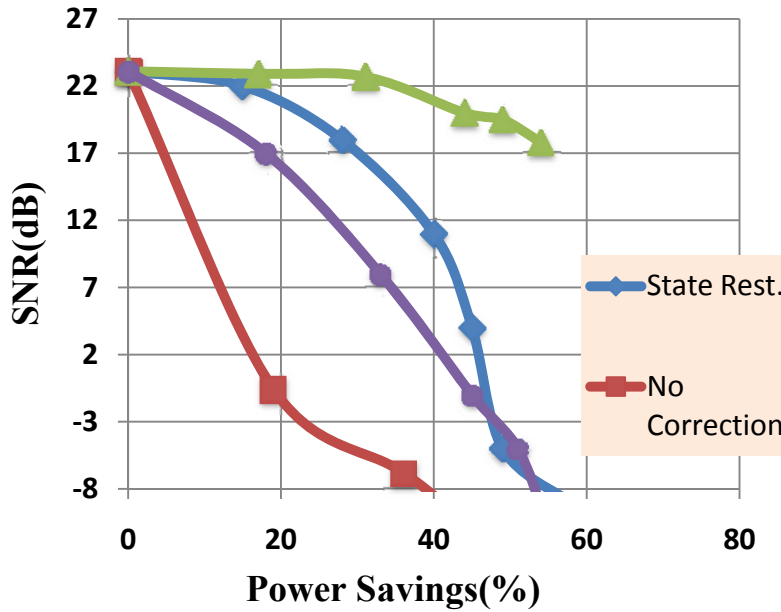


**Figure 41: PID-controller, preset error rate of 5% (a) LPF regular and DVOS output (b) Voltage and sampled error rate of LPF.**

### 4.5.3 Power Savings and Area Overhead

Power savings achieved by GPC scheme is plotted against uncorrected output, probabilistic compensation and state-restoration scheme in Figure 42. The power overhead of error detection

and correction units is incorporated while calculating the power efficiency of these schemes. Power savings up to 55% are achievable using the proposed scheme with approximately 5dB loss in system performance. The state-restoration scheme [96], in its simplest form restores the system states  $S_i(t)$  to previous states  $S_i(t-1)$  whenever an error is detected by the checksum block. The overhead of scheme is higher as compared to GPC as full precision flip-flops are used to store the previous states. However, no separate clock generation is required. It is stated in [96] that *state-restoration scheme performs well when errors are of high magnitude but occur in small bursts*. Under DVOS, errors are of high magnitude but they usually occur in long bursts before system voltage can recover to higher value. Therefore, state-restoration scheme fails to perform well under DVOS. Probabilistic error compensation technique also performs poorly in compensating for errors occurring under voltage overscaling.



**Figure 42: Power savings achieved with guided probabilistic compensation.**

For LPF fitted with a GPC scheme power estimates are calculated using mean voltage value for a given preset error rate in the voltage controller. LPF power is estimated using the mean operating voltage of circuit and statistical Monte-Carlo Power technique. SNR degradation in the system

with increasing preset error rate is captured by logic level simulations. Preset error rate in the system is gradually increased to find the power savings versus SNR relationship as plotted in Figure 42. Area overheads are calculated by implementing a regular LPF, LPF with GPC and a LPF with state restoration in Verilog HDL and synthesizing circuits using Synopsys design Compiler in a 65nm library. GPC scheme results in approximately 12% overhead as compared to with no-correction implementation method (see Table 8).

**Table 8: A 15-tap LPF implemented in 65nm technology.**

Implementation Method	Area ( $\mu\text{m}^2$ )
No Correction	35722.64
Prob. Compensation	39651.78
GPC Scheme	40187.45
State Restoration	43443.33

## 4.6 Concluding Remarks

In this chapter, a guided probabilistic error compensation technique for low-power digital filters is presented. The scheme uses checksum codes for error detection under voltage overscaling. Low precision shadow-latches are used for guided probabilistic compensation. Voltage overscaling is controlled by a voltage controller based on the error rate in the circuit. The feedback mechanism makes sure that the presented scheme can work with both, a highly correlated and uncorrelated input signal. No training phase is needed and system adapts dynamically to save power while keeping system performance within acceptable range. It is

shown that the technique results in significant power savings without considerable affect on system performance.

# CHAPTER 5

## SOFT ERROR MITIGATION AND LOW-POWER OPERATION OF NON-LINEAR FILTERS

As explained in Chapter 1, soft errors are a major reliability concern in Nano-CMOS. There is a considerable work in literature on soft error mitigation techniques. In this work, linear checksum codes are used for the first time for not only error detection but also probabilistic error compensation in non-linear filters. Non-linear filters are used extensively in DSP applications and are present in many mission critical systems. The state variable representation as explained in previous chapter is used for non-linear filters as well. In the following, first, *error detection* in non-linear filters enabled by the concept of time-freeze linearization is explained. Second, probabilistic *error compensation* scheme for soft error mitigation in non-linear filters is presented. Third, application of *guided probabilistic compensation* method for low-power operation in non-linear filters is explained. Finally, simulation setup and experimental results are presented.

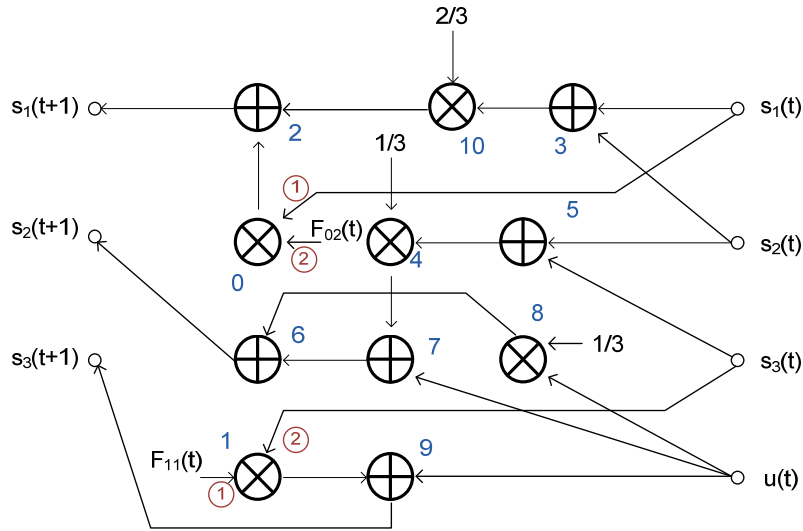
### 5.1 Non-linear Circuits – Checksum based Error Detection

Non-linear circuits considered in this work are interconnections of adders, multipliers and registers. However, in contrast to linear circuits, system states at time  $t+1$  are calculated as a weighted sum and non-linear functions of the system states and the inputs at time  $t$ . The non-linear function might be  $(S_i(t))^k$ , for some integer  $k$ . An example of a non-linear digital circuit is shown in Figure 43. The given circuit is described by equations as given below

$$S_1(t+1) = 2/3S_1(t) + 2/3S_2(t) + f_1(t),$$



one time frame to the next. In time-freeze linearization, all non constant multipliers and their inputs are labeled as  $ip_{i1}$  and  $ip_{i2}$ ,  $0 < i < \mu - 1$ , where  $\mu$  is the number of non-constant multipliers. For example, in Figure 43, there are two non-constant multipliers i.e.  $\mu = 2$ . “ $ip_{01}$ ,  $ip_{02}$ ” and “ $ip_{11}$ ,  $ip_{12}$ ” are the inputs of the non-constant multipliers 0 and 1 respectively. Next, a set  $I$  is defined such that it comprises of at least one input of all non-constant multipliers. In case of  $\mu$  non-constant multipliers in the circuit there are  $2^\mu$  possible ways to formulate set  $I$ . At any time instance  $t$ , the value  $F_{ij}(t)$  available at input  $ip_{ij}$  ( $j=1$  or  $j=2$ ,  $ip_{ij} \in I$ ) of the non-constant multiplier, is considered to be a constant coefficient and is multiplied by the data at the other input of the multiplier. For the example circuit, we may choose a set  $I = \{ip_{02}, ip_{11}\}$ . Then the function value at input two of multiplier 0 is  $F_{02}(t) = 1/3(S_2(t) + S_3(t))$  and input one of multiplier 1 is  $F_{11}(t) = 1/3S_1(t)(S_2(t) + S_3(t))$ .



**Figure 44: Time-freeze linearized circuit.**

The time-freeze linearized circuit of Figure 43 is shown in Figure 44. From the time-freeze linearized circuit data flow graph, it is possible to represent the non-linear circuit function in the form of a *linearized state equation* (LSE). Thus making it possible to apply concurrent error detection techniques of linear digital filters (as explained in last chapter) on non-linear filters

represented in LSE form. The linearized state equation and gain matrix (GM) of the example circuit are given as

$$\begin{bmatrix} S_1(t+1) \\ S_2(t+1) \\ S_3(t+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 2/3+F_{02}(t) & 2/3 & 0 \\ 0 & 1/3 & 1/3 \\ 0 & 0 & F_{11}(t) \end{bmatrix}}_A \times \begin{bmatrix} S_1(t) \\ S_2(t) \\ S_3(t) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 4/3 \\ 1 \end{bmatrix}}_B \times [U(t)]$$

$$GM = \left\{ \underbrace{\begin{bmatrix} 1 & 0 & 1 & 2/3 & S_1(t) & 1/3S_1(t) & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1/3 & 1 & 1 & 1 & 0 & 0 \\ S_3(t) & 1 & 0 & 0 & S_1(t)S_3(t) & 1/3S_1(t)S_3(t) & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{\text{Operators 0 to 9}} \right\}$$

where matrix A and GM are considered constant during a single time frame, although they change over time. Assuming  $CV = [1 \ 1 \ 1]$ , the modified LSE is given below:

$$\begin{bmatrix} S_1(t+1) \\ S_2(t+1) \\ S_3(t+1) \\ c(t+1) \end{bmatrix} = \begin{bmatrix} 2/3+F_{02}(t) & 2/3 & 0 \\ 0 & 1/3 & 1/3 \\ 0 & 0 & F_{11}(t) \\ 2/3+F_{02}(t) & 1 & 1/3+F_{11}(t) \end{bmatrix} \times \begin{bmatrix} S_1(t) \\ S_2(t) \\ S_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 4/3 \\ 1 \\ 7/3 \end{bmatrix} \times [U(t)]$$

The presence of a soft error in the system is detected by a non-zero value of  $e(t+1)=CV.s(t+1)^T-c(t+1)$ , as in the linear digital systems. As mentioned earlier, all single operator faults can be detected if and only if all the elements of the product  $CV.GM$  are non-zero. However, in the case of non-linear filters, the gain matrix GM is a function of time. Since coding vector cannot be changed in the system at runtime,  $CV.GM(t)=0$  condition may result in cases where it is not possible to detect the errors in a system.



## 5.2 Non-linear Circuits - Probabilistic Soft error Compensation

The objective of probabilistic error compensation is to improve the signal-to-noise quality ratio (SNR) in non-linear systems. The proposed compensation method does not *correct errors* as there is no error diagnosis; however, the error effects are mitigated and SNR is improved significantly in the system. In our error modeling approach, if an error occurs in the time period (t, t+1), then state vector S(t+1) has wrong values for one or more of the system states and results in a non-zero error vector, which is detected during the time period (t+1, t+2). For a single fault, error vector goes to zero in the next clock cycle. Whenever, error vector is non-zero, magnitude of error and coding vector information is used to compensate the error in the system. In a digital filter, soft errors may affect system states (registers) or computational units (adders/multipliers, also called *operators*). These errors can propagate to primary system output(s) and thus affect the system SNR. System SNR calculation is performed as explained next.

Let Y be the system output signal with no errors in system and Y<sub>err</sub> be the output signal in the presence of a soft error. Then error at the system output at the  $i^{\text{th}}$  time instant is given by

$$\text{error}(i) = Y(i) - Y_{\text{err}}(i).$$

If a total of T time steps are considered, the SNR of a system is defined in terms of its output signal and error power as given below:

$$\text{SignalPower} = \sum_{i=1}^T Y(i)^2, \quad \text{ErrorPower} = \sum_{i=1}^T \text{error}(i)^2, \quad \text{SNR} = 10 \log_{10} \left( \frac{\text{SignalPower}}{\text{ErrorPower}} \right).$$

Let EV be the  $n \times 1$  error vector that represents the error in the state-variable values whenever an error occurs in the system. Then in the presence of an error, system states can be represented as

$$S_{\text{err}}(t+1) = S_{\text{good}}(t+1) + \text{EV}. \quad (11)$$

In non-linear circuits, error in the state values has time dependence. After  $k$  cycles, error in the system states will be

$$S_{\text{err}}(t+1+k) = S_{\text{good}}(t+1+k) + A(t)^k E V. \quad (12)$$

If no further errors occur, then error in the system will die down in  $m$  cycles as  $A^m \rightarrow 0$ .  $A(t)^k$  is a time dependent matrix because of non-linear functions in the system. If an error  $\varepsilon$  occurs at the output value of an operator, its effect on final states and output of the system depends on the interconnection topology of the non-linear digital filter. For an error,  $\varepsilon$ , affecting the output of the  $j^{\text{th}}$  module (state or operator), the error in the state  $S_i$  is  $g_{i,j}(t) \times \varepsilon$  (gain of a module may be or may not be a time dependent function, as shown in previous section). However, for the sake of clarity, we will write error in the state  $S_i$  as  $g_{i,j}(t) \times \varepsilon$ . The error detected by the checksum codes also depends upon the coding vector  $CV = [\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n]$  and is given as

$$e(t+1) = \left( \sum_{i=1}^n \alpha_i g_{i,j}(t) \right) \times \varepsilon, \quad (13)$$

where  $\alpha_i$  is the  $i^{\text{th}}$  element of the coding vector, and  $n$  is the number of states in the system. In the case of a known erroneous module, error at a state  $S_i$  in terms of the system error is given as

$$e_{s_i} \mid_{\text{module}_j} = g_{i,j}(t) \times \varepsilon = \frac{g_{i,j}(t) \times e(t+1)}{\left( \sum_{i=1}^n \alpha_i g_{i,j}(t) \right)}. \quad (14)$$

Above equation can be rewritten as

$$e_{s_i} \mid_{\text{module}_j} = \gamma_{i,j}(t) \times e(t+1), \quad (15)$$

where  $\gamma_{i,j}(t)$  is given as  $\gamma_{i,j}(t) = \frac{g_{i,j}(t)}{\left( \sum_{i=1}^n \alpha_i g_{i,j}(t) \right)}$ .

Let  $E_j(t)$  be the erroneous state vector when the  $j^{\text{th}}$  module is in error. The expanded form of  $E_j(t)$  is given below

$$E_j(t) = \begin{bmatrix} \gamma_{1j}(t) \\ \gamma_{2j}(t) \\ \vdots \\ \gamma_{nj}(t) \end{bmatrix} \times e(t+1) = \gamma_{ij}(t) \times e(t+1) \quad . \quad (16)$$

It is imperative to mention here that although the individual gain of some of the system modules maybe *time independent*, however in a non-linear circuit,  $E_j(t)$  is always a time dependent function (as some modules will always have a time dependent gain). Also,  $E_j(t)$  represents the erroneous state vector whenever a system module is in error, and EV represents the error magnitude in the system states. The goal of error compensation scheme is to determine an *error compensation vector*  $\Delta_{n \times 1}$  for compensating the error (EV) in the states such that after compensation, error in the states is  $EV - \Delta$ . Error in the system states and the system output after k cycles is  $A^k(EV - \Delta)$  and  $CA^k(EV - \Delta)$ , respectively.

To improve the system SNR with probabilistic compensation, average system output noise power is to be minimized. The system output noise in the presence of a compensation vector  $\Delta$  is given as

$$AverageNoise = \sum_{i=1}^n \sum_{k=0}^m w_i (CA^k (E_i(t) - \Delta))^2 \quad , \quad (17)$$

where  $w_i$  is the probability of the  $i^{th}$  module being erroneous. A solution to the minimization problem, assuming  $\sum_{k=0..m} CA^k \neq 0$ , is given as

$$\Delta = \sum_{i=1}^n w_i E_i(t) \quad . \quad (18)$$

Above equation can be rewritten using equation 7 and the resulting compensation vector is given below

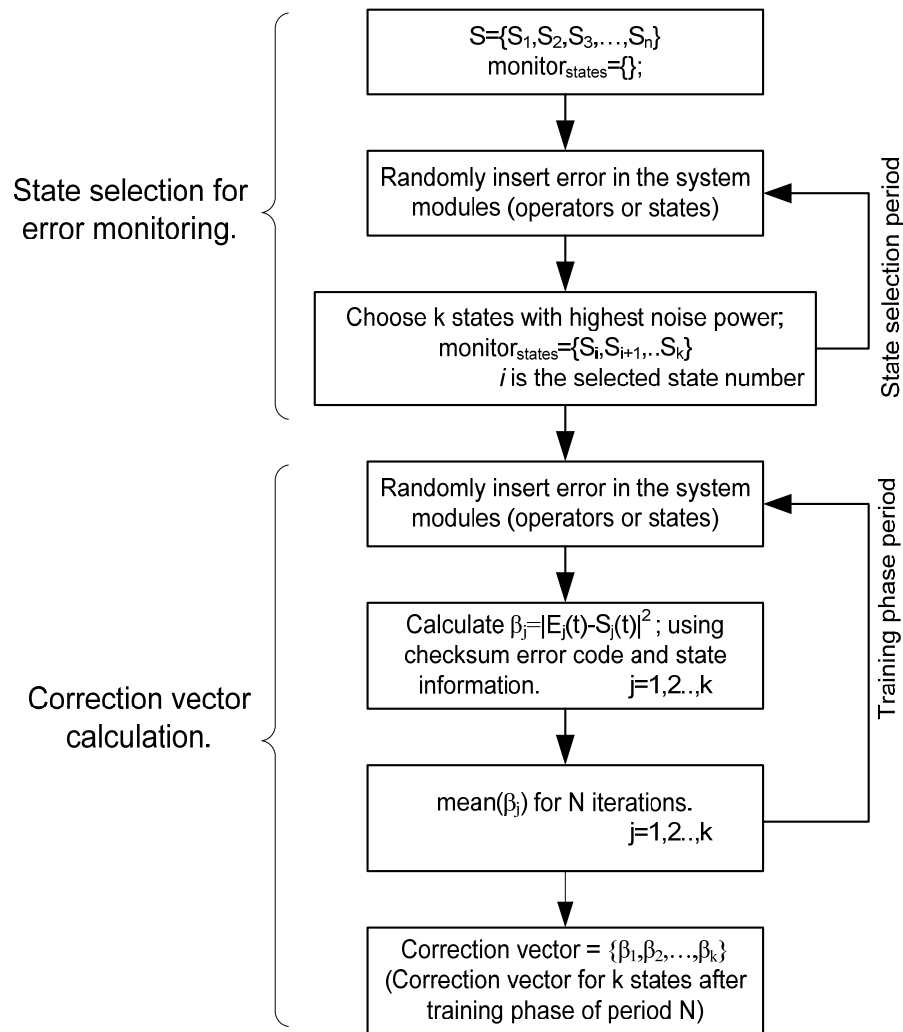
$$\begin{aligned}\Delta &= \left( \sum_{i=1}^n w_i [\gamma_{1,i}(t) \dots \gamma_{n,i}(t)]^T \right) \times e(t+1) \\ &= [\beta_1, \beta_2, \dots, \beta_n] \times e(t+1) \quad ,\end{aligned}\tag{19}$$

where  $\beta_j = \sum_{i=1}^n w_i \times \gamma_{j,i}(t)$  and is called the *correction vector*. To perform system error compensation, system error  $e(t+1)$  is multiplied with the correction vector  $[\beta_1, \beta_2, \dots, \beta_N]$  resulting in the *compensation vector*  $\Delta$ , which is then subtracted from the system state vector. The correction vector is calculated prior to the system implementation. However, as it is a time-dependent vector in non-linear circuits, the optimal value of correction vector changes as a function of time. Since, it is not possible to calculate the correction vector at every time instance during the run-time operation, the non-linear circuit is passed through a training phase to calculate a constant quasi-optimal correction vector, as explained in the next section.

### 5.2.1 Training Phase – Correction Vector Calculation

To find the constant correction vector for a non-linear circuit we pass the system through a training phase as shown in Figure 45. Since filters are used in specific applications, their input statistics can be characterized. For a given input, the *training* phase is broken down into *critical-state selection* and *correction vector* approximation. We define the training phase period,  $N$ , as the number of times errors are inserted into the system. During the training phase, errors are randomly injected into the system operators and states, while the filter is excited by a stimulus similar to what the filter would experience during real-time operation. In response to the injected errors, error signal  $e(t+1)$  and the erroneous state values are observed. The system states with higher-noise power are selected to be monitored and compensated at run time, as shown in Figure 45. After selection of  $k$  states to be monitored, random errors are injected again in the system. Then the correction vector  $\beta_j$  is calculated as the mean squared

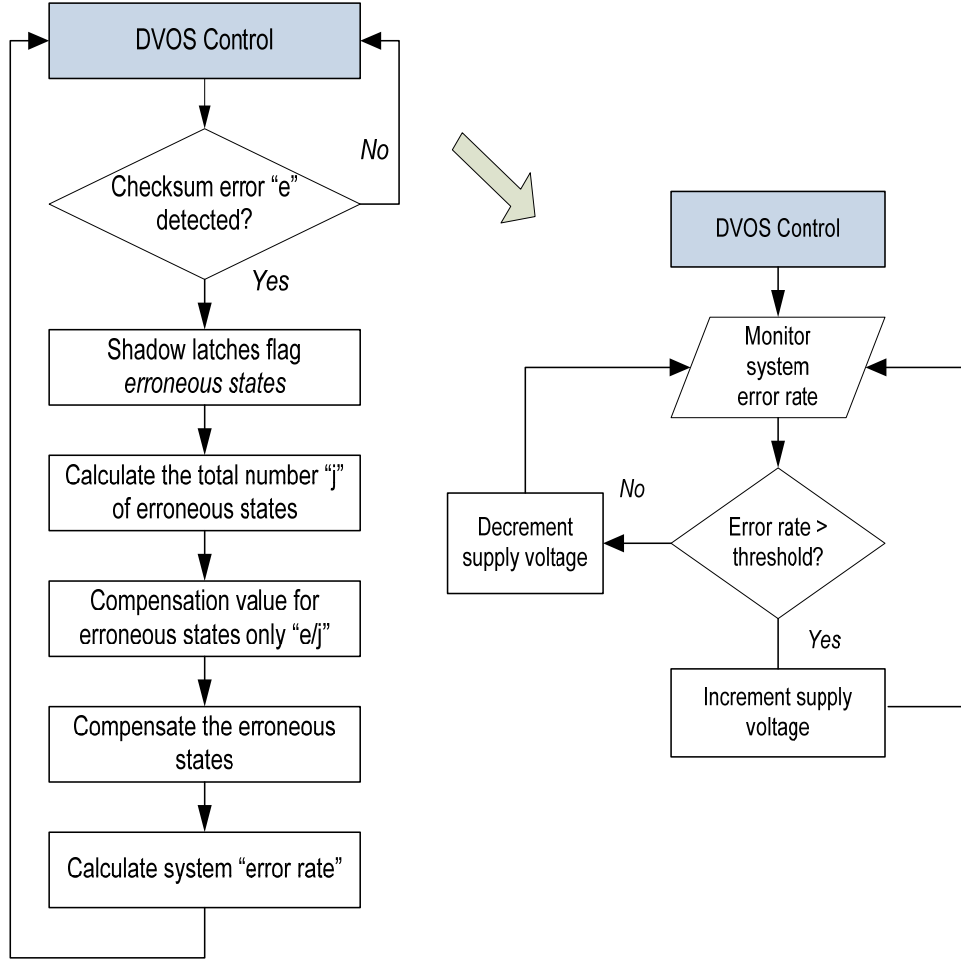
value  $\beta_j = \frac{1}{N} \sum_{i=1}^N |E_j(t) - S_j(t)|^2$  for  $j=1,2,..,k$  and length of the training period  $N$ . There is one-time *training period time cost* to calculate the error compensation vector for a given circuit. It is observed that the selective monitoring of the system states enable better probabilistic error compensation. As mentioned above, system states that do not contribute significantly to the system noise are not monitored. The corresponding elements for such states are set equal to zero in the coding vector CV.



**Figure 45: Training phase - Compensation vector selection.**

### 5.3 Guided Probabilistic Error Compensation for Low-Power

The guided probabilistic compensation (GPC) scheme for low-power operation of linear digital filters is presented in last chapter. In this work, the application of GPC scheme is extended to non-linear digital filters. Typically, transient errors induced by voltage overscaling are of high magnitude and occur in bursts. Under such a noisy environment, better error compensation is needed in the circuit than probabilistic error compensation. “*Guided probabilistic compensation*” methodology utilizes the diagnosis properties of shadow latches (as explained in previous chapter) and system error value from the checksum codes to perform error compensation in the circuit. The proposed GPC framework for non-linear filters is summarized in Figure 46. For the system supply voltage adjustment, the system controller continuously monitor the system error rate; supply voltage is increased if the error rate increases above the allowable threshold level and supply voltage is decreased whenever the error rate is below the error rate threshold. Simultaneously, system error magnitude is approximated by the checksum code and possible erroneous states are flagged by the shadow latches. Compensation is performed only on the erroneous states and error rate is updated.



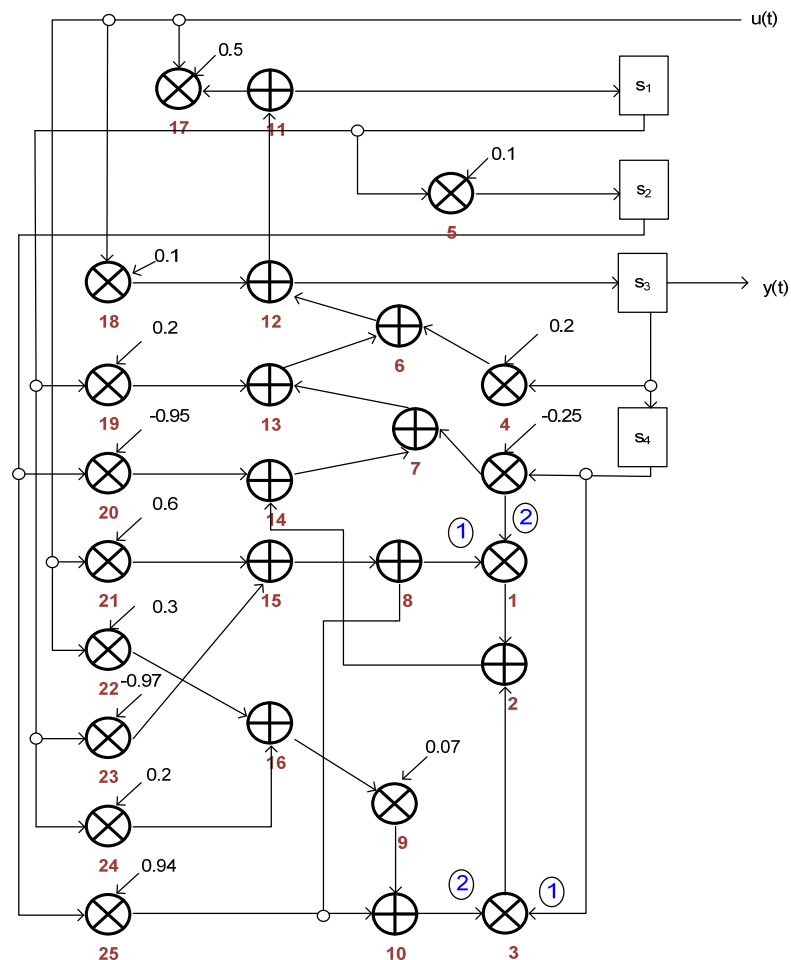
**Figure 46: Guided probabilistic error compensation with DVOS control.**

## 5.4 Evaluation

### 5.4.1 Simulation setup

A non-linear filter as shown in Figure 47 is implemented along with the error detection and the error compensation modules in MATLAB. Soft errors are modeled by modifying the magnitude of the filter state values according to the gain of the faulty operator for each state. The system precision is set to be 12-bit and in Q4.8 fixed-point format; negative numbers are represented in 2's complement form. For emulating the VOS effects on system performance, customized filter modules (multipliers, adders, etc.) are implemented using HSPICE bit-slice data. As an example, the critical path delays of the multiplier are estimated from the module

implementation and circuit level simulations in HSPICE at varying voltage levels (1-Volt to 0.55-Volt in voltage steps of 1 mV). These delay estimates are used to model path delay errors under VOS in the MATLAB modules. 65 nm, 1-volt CMOS libraries are used for circuit level simulations of the modules in HSPICE. For precise area and power estimates, the equivalent circuit is implemented in Verilog HDL and is synthesized using Synopsys Design Compiler. Power is estimated in Synopsys based on the average operating voltage of the system for a given input and threshold error-rate. System performance is characterized by SNR as given in equation 2. In this work, proportional controller is implemented using the Ziegler-Nichols method [91].

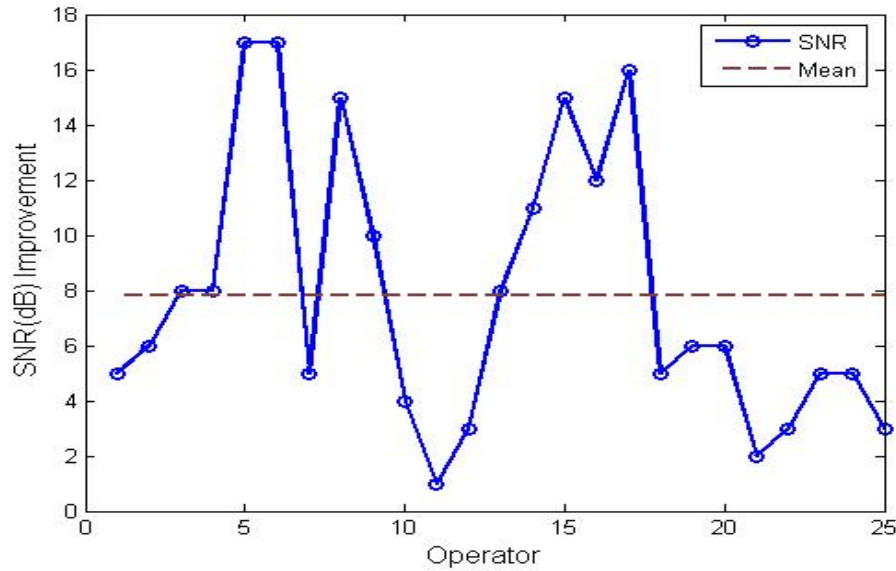


**Figure 47: Non-linear circuit used for simulation results.**

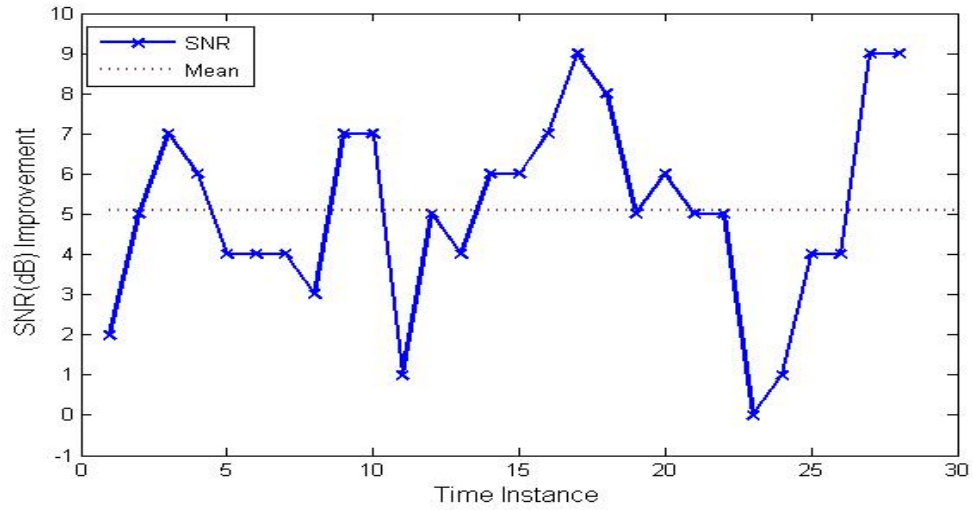


### 5.4.2 Probabilistic Compensation

Figure 48 shows the SNR improvement in the non-linear circuit (Figure 47) under soft errors by using probabilistic error compensation. The soft errors are injected in one operator at a time and the corresponding SNR improvement using probabilistic error compensation is calculated. The injected soft errors are of varying magnitude with a mean error magnitude value of 1 and variance of 0.5. In this experiment, 8 dB is the mean SNR improvement for the soft errors randomly injected in the system. Because of the non-linear functions, the SNR improvement at the system output, for the soft errors in an operator varies over time. In Figure 49, soft errors are injected in operator 1 at different time instances and the SNR improvement because of the probabilistic compensation is calculated. The average SNR improvement for the soft errors in the operator 1 is approximately 5 dB.



**Figure 48: SNR gain by probabilistic error compensation on different operators.**



**Figure 49: SNR gain at Operator 1 on different time instances.**

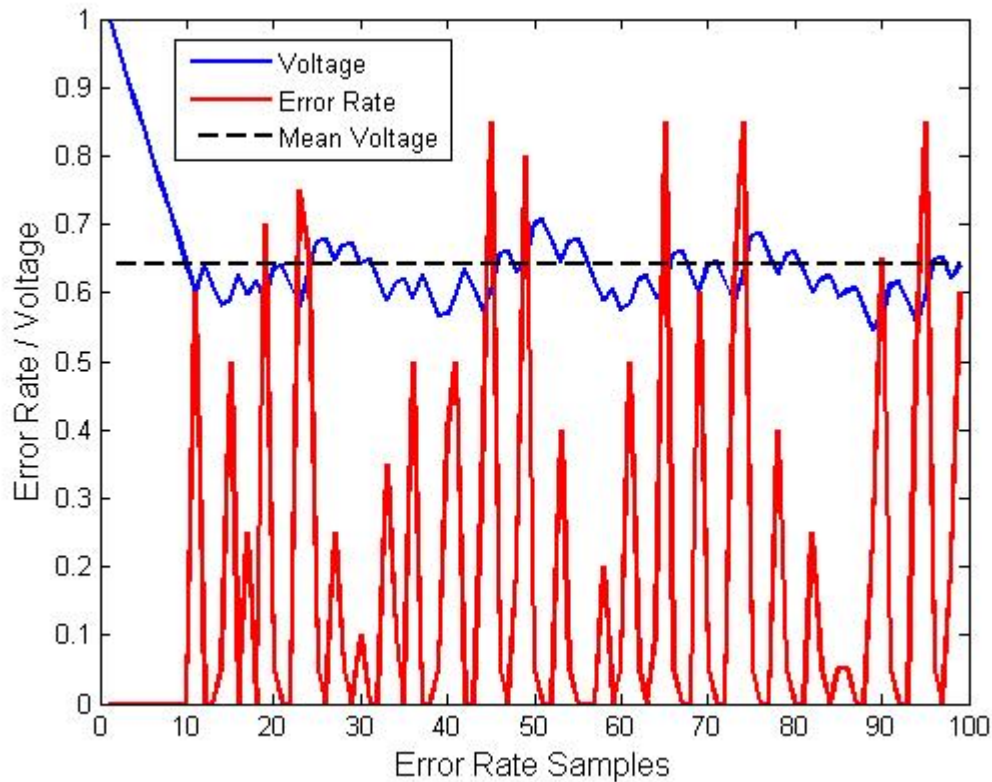
Table 9 summarizes the effects of soft errors with increasing input sampling frequency on the system SNR in the presence of probabilistic error compensation and state restoration scheme. The state restoration scheme is a simple error compensation scheme in which all system states are restored to their respective last values, whenever an error is detected in the system. It is observed that the probabilistic error compensation scheme works better than the state restoration scheme, especially at lower frequencies.

**Table 9: Effect of sampling frequency on system SNR.**

Sampling Frequency (x Input freq.)	No Correction SNR (dB)	Probabilistic Compensation SNR (dB)	State Restoration SNR (dB)
4x	57.1	68.4	35.8
8x	66.9	74.5	48.4
16x	73.3	82.6	65.9
32x	80.3	89.9	83.1

### 5.4.3 Guided Probabilistic Compensation

The guided probabilistic compensation scheme as shown in Figure 22 is implemented with the non-linear circuit shown in Figure 47. For the voltage controller implementation, the minimum voltage step is assumed to be 1mV and the switching frequency between two discrete voltage levels is assumed to be 5MHz, i.e., 200nsec. To ensure the stability of the voltage control system, error-rate is calculated over 400nsec. The voltage controller changes the output voltage based on sampled error-rate, preset error-rate, and proportional constant value in the controller. In this work, Ziegler-Nichols method [22] is used to determine the parameter settings of the proportional controller. The P-controller is implemented with the preset error-rate of 25%. The system voltage adjusted by the voltage controller and the calculated error-rate is plotted in the Figure 50 and the mean voltage is approximated at 0.64 volts.

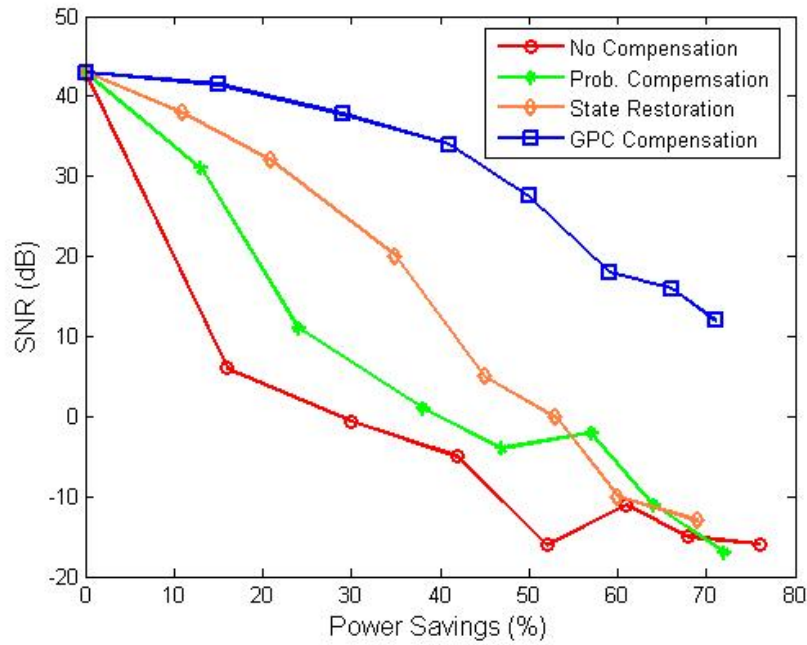


**Figure 50: System error rate and voltage values under GPC scheme.**

Power savings achieved by GPC scheme is plotted against uncorrected output, probabilistic compensation and state-restoration scheme in Figure 51. The power overhead of the error detection and correction units is incorporated while calculating the power efficiency of these schemes. Power savings are calculated using the following relationship

$$Power\ Savings = \frac{Power_{org.} - Power_{DVOS}}{Power_{org.}},$$

where  $Power_{org.}$  and  $Power_{DVOS}$  are the system power without and with a error detection/compensation scheme respectively. Power savings of greater than 40% are observed using the proposed GPC scheme with approximately 6dB loss in the system performance. It is stated in [96] that the state restoration scheme performs well when errors are of high magnitude but occur in small bursts. Under DVOS, errors are of high magnitude but they usually occur in long bursts before the system voltage can recover to a higher value. Therefore the state restoration scheme fails to perform well under DVOS. The probabilistic error compensation technique also performs poorly in compensating for errors occurring under voltage overscaling. For the non-linear filter, fitted with a GPC scheme, power estimates are calculated using mean voltage value for a given preset error-rate in the voltage controller. Filter power is estimated using the mean operating voltage of the circuit and the statistical Monte-Carlo power technique. The SNR degradation in the system with increasing preset error-rate is captured by logic-level simulations.



**Figure 51: System SNR and power savings using GPC scheme.**

Table 10 summarizes the implementation area for the different schemes for the non-linear filter. The non-linear filter along with different error detection/compensation schemes is implemented in Verilog HDL and synthesized using Synopsys design Compiler in a 65 nm library. GPC scheme results in approximately 20% area overhead as compared to with no correction implementation method. Area overheads in the GPC scheme and the state restoration scheme are comparable; however, as shown above, GPC scheme performs better in a DVOS environment.

**Table 10: Area overhead of the proposed schemes.**

Technique	Area ( $\mu\text{m}^2$ )
No Correction	30950.80
Probabilistic Compensation	37140.96
GPC Scheme	37912.72
State Restoration	37743.54

## 5.5 Concluding Remarks

In this chapter, the use of linearized checksum codes for soft error mitigation and guided probabilistic compensation is summarized. The concept of time-freeze linearization makes it possible to extend checksum based concurrent error detection to non-linear circuits. A methodology is presented for the best correction vector estimation at design-time, which is then used for probabilistic error compensation at run-time. A guided probabilistic error compensation scheme is also presented that allows low-power operation of non-linear filters by DVOS and continuous monitoring of system error-rate. It is shown with experimental data that the probabilistic compensation technique results in significant SNR improvement in case of soft errors and the guided probabilistic compensation scheme allows considerable power savings. The area overhead of the proposed schemes is minimal compared to the module duplication and TMR schemes.

# **CHAPTER 6**

## **CHANNEL AND VARIATION ADAPTIVE LOW-POWER BASEBAND PROCESSING**

As technology scales below the 45nm CMOS technology node, RF front ends and baseband processors will need to be aggressively overdesigned to work reliably under worst case channel (environment) conditions as well as worst case manufacturing variations. In this chapter, novel techniques for low-power variation-tolerant OFDM baseband receiver are presented. The core idea behind these techniques is to degrade receiver performance to conserve power under good operating conditions while satisfying the minimum required system performance requirements. Receiver performance is degraded by reducing the supply voltage ( $V_{dd}$ ) and wordlength ( $W$ ) precision of different receiver modules in a fashion. A power control law is devised to select the  $V_{dd}$  and  $W$  settings of different baseband modules under different channel conditions in such a way that results in optimal power savings with minimal impact on baseband performance. Different methods are presented in this work for channel and variation aware adaptive low-power baseband processing. In the first method, system operating loci are calculated for varying channel conditions and process variations and stored in the system at design time. At run time, path oscillation timing tests (POTTs) are used to determine the system process variations and hence the “best” operating locus. An error vector magnitude (EVM) driven feedback loop is used to dynamically modulate the wordlength/power consumption of each module as dictated by this locus to minimize power and modulate baseband SNR across a range of channel conditions. In the second method, a dual feedback based design approach is proposed that allows the baseband unit of a wireless OFDM system to adapt dynamically to channel conditions as well as

manufacturing process variations without the need of pre-calculated and stored system loci. Two nested feedback control loops are used; the first allows the baseband SNR to increase when channel conditions are good and vice versa by modulating the system wordlength, the second control loop modulates the system supply voltage in response to the changing wordlength precision. Both feedback loops are designed to allow the processor to operate at the minimum power consumption without exceeding a specified overall bit error rate across process variability conditions and dynamically changing noise conditions. In the third method, the knowledge of the end application is used to determine the system performance requirements at a given time, thus saving power when performance requirements are not high. Image transmission across a wireless link is studied for this method with end application requirements of edge detection and centroid location on the received data. It is assumed that the image quality metric is computed by the image acquisition system and transmitted along with the image data.

## **6.1 Prior Work**

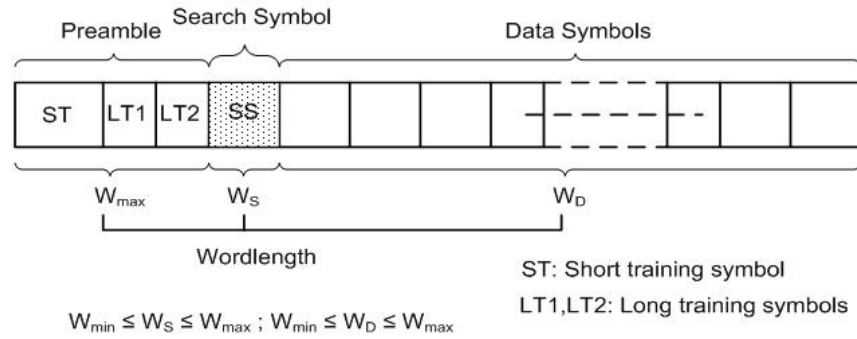
Low-power design is a very well-studied field and it is not possible to summarize all the power savings techniques in the given space. Therefore, in this section, only the relevant power savings methodologies are presented.

### **6.1.1 Tuneable Wordlength**

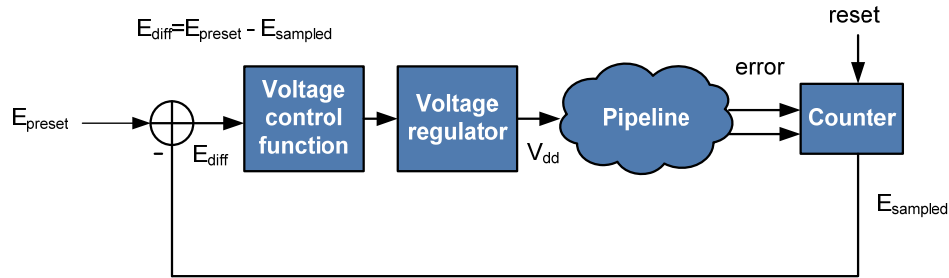
Determining the optimal wordlength for digital systems is difficult because of inconclusive tradeoffs between hardware overhead and system performance. Different simulation based techniques [36]-[39] find the optimal wordlength for DSP systems according to predetermined system-level performance metrics. Often, the resulting digital circuits are implemented with large wordlength values to avoid calculation errors due to lack of dynamic range. A method for dynamic tuning of wordlength under changing environment conditions was



proposed in [40] for an OFDM baseband receiver. In the proposed technique, the OFDM demodulator measures the error vector magnitude (EVM) of demodulated symbols and tunes the system wordlength to satisfy the end signal quality requirements. The structure of an OFDM packet with preamble, search symbol and data symbols is shown in Figure 52. The preamble in the OFDM packet consists of a short training symbol “ST” and long training symbols “LT1 and LT2”. The ST symbol is used to synchronize the packet and the LT symbols are used for channel estimation.



**Figure 52: OFDM packet with preamble and added search symbol.**



**Figure 53: Supply voltage controller based on system preset and sampled error rate.**

Within the receiver, channel estimation is performed using a signal of maximum wordlength  $W_{max}$ . Using this channel metric, the system performance for a signal of wordlength  $W_S$  is estimated.. If the estimated system quality is sufficient, the receiver sets the signal wordlength value to  $W_D$  such that  $W_D \leq W_S$ , else the wordlength value is set to  $W_D > W_S$ . The demodulator thus receives data symbols at wordlength  $W_D$ . In this work, signal quality is determined by EVM and

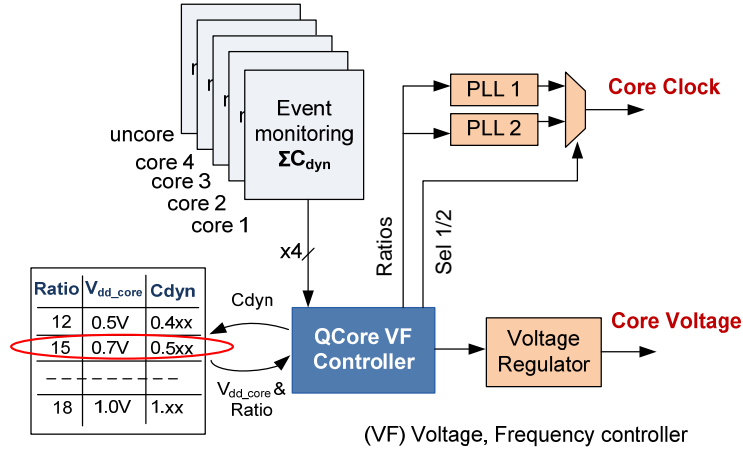
it is shown that EVM has strong correlation with system bit error rate (BER). The proposed scheme relies on the use of *gated clocks* for wordlength adaptation and power savings. However, the scheme requires significant modifications to the circuit architecture and nontrivial routing resources.

### **6.1.2 Dynamic Voltage Scaling in a Pipeline Architecture**

Dynamic voltage scaling (DVS) is a very effective power savings technique because of the quadratic relationship of voltage with power. To obtain maximum power savings, the supply voltage is scaled to a critical value below which correct operation of the circuit cannot be guaranteed. The critical supply voltage level is chosen such that delay requirements of critical paths are still met even under worst case environment and process variations. In the Razor [45] framework, a DVS technique is presented that pushes the limits of critical supply voltage with the help of error detection and correction. The core idea of the Razor technique is to modulate the supply voltage while monitoring system error rate, thus taking advantage of data dependence on circuit delay and eliminating the need for voltage margins. The technique uses a double latched pipeline operating on a regular clock and a delayed clock. The delay between the two clocks is set such that the latch operating on the delayed clock always latches the correct value under voltage overscaling. A metastability-tolerant comparator is used to match the values of both latches and flags an error in case of a mismatch. Whenever an error is flagged, error recovery mechanisms restore the correct data value in the pipeline from backup devices. It is shown that the method results in significant power savings with little impact on system performance. A feedback based voltage controller as shown in Figure 53 is implemented that modulates the supply voltage of the system based on the observed error rate due to voltage overscaling.

### 6.1.3 Power-Frequency Management

Dynamic voltage and frequency scaling is used in many novel processor designs. In [42], a new design for power-frequency management of a Quad-core Itanium processor is introduced based on digital activity sensing and discrete voltage-frequency pairs (see Figure 54). In every processor core, approximately 120 architectural events are monitored and weighted to represent the relative amount of capacitance being switched by each event. A central control system accumulates these events to estimate current switching activity and makes a decision every 6 $\mu$ sec to choose a voltage-frequency pair from the lookup table for the core. This lookup table is stored in the system pre-production along with associated capacitance thresholds. However, the same lookup table is used for all the cores in a die and is not fine-tuned to take into account inter and intra-die process variations.

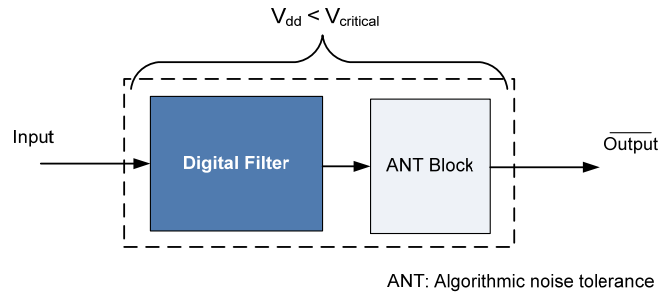


**Figure 54: Switching activity based voltage-frequency controller in Itanium Quad-Core processor.**

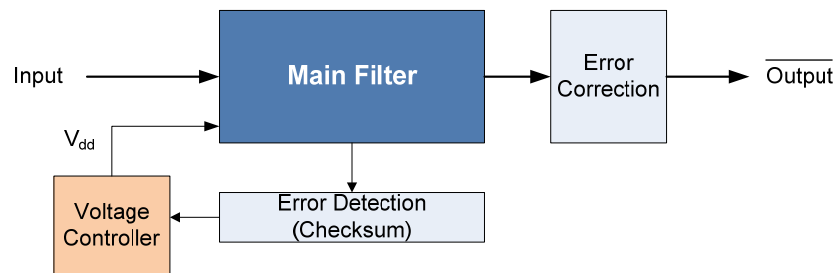
### 6.1.4 Voltage Overscaling and Algorithmic Noise Tolerance

Different techniques can be found in literature that combines VOS with ANT for low-power operation of digital filters [46][47]. The key idea is to scale the supply voltage below the critical voltage to save power and perform algorithmic error correction to minimize the impact of delay constrained errors on overall system performance (see Figure 55). The effectiveness of such a

scheme largely depends on the following: the magnitude and frequency of errors (architecture and data dependent); error detection and correction capability; and error-control overhead in terms of power and area. Many ANT techniques such as error cancellation, reduced-precision redundancy, and linear-predictor based algorithms are present to improve the end signal quality [48]. VOS schemes combined with ANT result in significant power savings with minimum impact on system performance. However many of these schemes operate only on a pre-calibrated VOS level as they lack real time feedback for modulating supply voltage with changing input statistics and system performance requirements. In our prior work, we proposed a checksum based technique for dynamic voltage overscaling for low-power operation of digital filters, which relies on system error rate to modulate the supply voltage as shown in Figure 56.



**Figure 55: Voltage overscaling for low-power filter operation.**



**Figure 56: Supply voltage feedback control**

## 6.2 Motivation and Overview

Power consumption and process variations are major concerns in highly scaled and functionally complex devices. In wireless baseband receivers, the noise performance of specific signal processing algorithms for signal demodulation and symbol decoding can be traded off for power under good channel conditions. In operating conditions where the worst case channel is seen infrequently, significant power can be saved by reducing the performance of the baseband signal processing algorithms (trade off performance for power) when channel conditions are not worst-case. The amount of power that can be saved depends however, on the speed of the underlying logic circuitry, i.e. the process parameters corresponding to the manufactured unit. Hence, for minimum power operation, knowledge of both channel conditions as well as logic speed (process parameters) is necessary.

In wireless communication systems, existing power management schemes [87]-[89] rely on channel quality metrics to modulate the data communication rate (radio-link control). These metrics are derived from the analysis of received “pilot symbols” embedded in each packet of transmitted data. Pilot symbols are short signal sequences that are transmitted prior to the data in each packet to enable the receiver to characterize the channel, and to calibrate the receiver for current channel conditions (noise, attenuation, etc.). A range of channel conditions can be accommodated for each specified data rate of communication that the wireless system can support. If the channel conditions become inferior to the worst channel quality for a specific data rate, then the wireless system switches to the next *lower* possible data communication rate. On the other hand, if the channel condition improves significantly, then the data rate is increased to the next *higher* data rate possible. For any given data rate, the wordlength of the baseband signal processing system is always (statically) set in such a way that the system bit error rate (BER) is

less than the maximum allowed BER value for the wireless communication protocol being used for communication. This wordlength therefore corresponds to the worst channel quality that the data rate can accommodate. Given the fact that most of the time in which the mobile device operates at that data rate, it will not be working in a worst case environment; the effective wordlength can be reduced when the channel is not worst-case, saving power while keeping the system BER within quality of service (QoS) requirements. A simple example of the above is when a wireless device is used close to a transmission tower vs. far from any transponder. In the following, channel driven adaptation techniques are first presented followed by the application driven low-power technique.

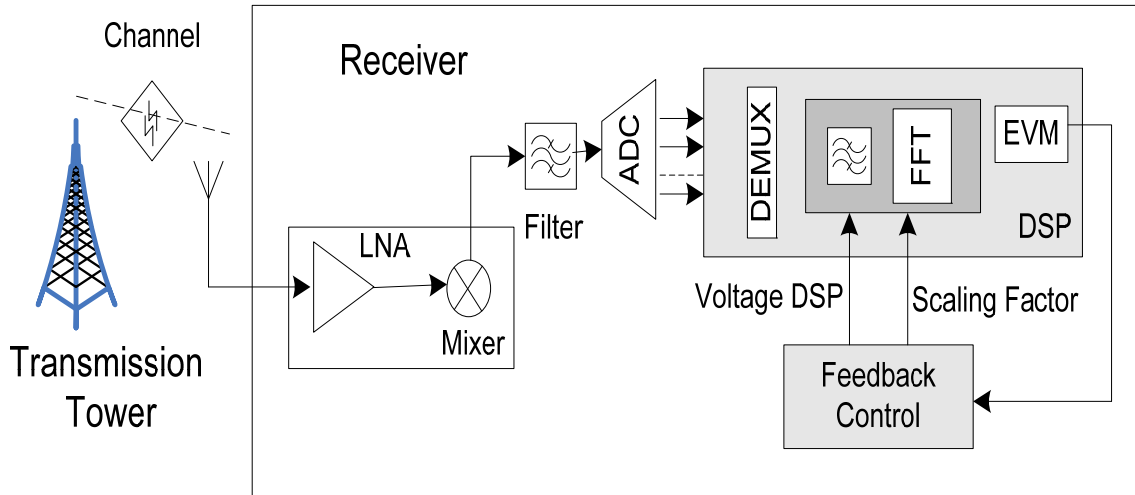
## **6.3 Channel Driven Adaptation Techniques for Low-Power**

In the following two techniques are presented for channel driven variation-tolerant and low-power operation of baseband receiver. In the first presented technique, design time analysis is performed to calculate the operating conditions of the baseband modules under changing channel conditions. The resultant operating conditions for a given process variation in the device is termed as the *operating locus* of the device. In the second technique, dual nested control loops are used to determine the operating conditions of the baseband modules at run-time, therefore the technique requires no design-time analysis, storage of loci, and delay tests to determine the process variation in the device.

### **6.3.1 Locus Based Channel and Variation Tolerant Low-Power Processing**

In order to adjust the baseband receiver operating conditions ( $V_{dd}$ ,  $W$ , etc), a signal quality metric is needed that quantifies the cumulative sum of quality of transmission, channel quality degradation and quality of signal reception. Such a metric, called the *adaptation metric* is

computed by the baseband signal processor in real-time (online). When this adaptation metric has a “high” value, the quality of signal processing in the baseband DSP can be traded off (degraded) for power consumption within specific limits and vice versa. In our proposed technique, signal is degraded by scaling down the input (dropping LSB bits) and correspondingly adjusting the supply voltage. The approach exploits the fact that the critical circuit path lengths of the underlying arithmetic units (multipliers and adders) are reduced by input data scaling. This reduction in circuit critical path length allows the correct operation of the arithmetic units at lower supply voltage without incurring additional bit errors. Since data scaling causes LSB bits drop, it results in graceful system-level performance degradation. This dynamic  $W/V_{dd}$  adjustment is done to save power while keeping the adaptation metric of the demodulated signal below a specified upper limit.



**Figure 57: Block diagram of adaptation metric based receiver architecture.**

In the presented method, EVM of the demodulated signal is used as the adaptation metric. EVM is chosen as an adaptation metric because of its strong statistical correlation with system BER under fading channels as well as AWGN channels [40]. Our own studies indicate a strong correlation between EVM and BER (see Figure 58). The feedback system of Figure 57 generates

the control signals for adaptive signal scaling in the baseband DSP. Such a system always strives to operate at the lowest power consumption levels (lower but acceptable performance) for any specified data rate through adaptive control of DSP wordlength thus saving significant power whenever the channel quality is not worst-case.

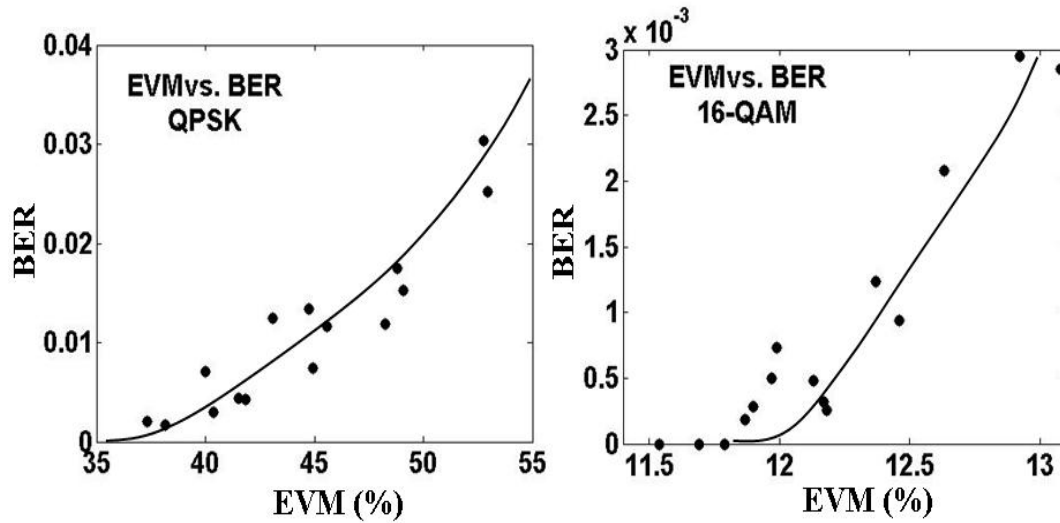
### 6.3.1.1 Adaptation Metric – EVM

Traditionally, the performance of a communication system is defined in terms of bit error rate (BER) of the received data. Typical BER values for wireless systems are of order  $10^{-3}$ - $10^{-4}$  and accurate BER measurement is possible only for a large number of transmitted symbols. Hence, we use the error vector magnitude (EVM) of the received signal for characterizing system performance as it can be computed across a few frames of transmitted data and exhibits strong correlation with BER. EVM is given by the sum of the vector differences between the received and ideal signal constellation points and is computed as

$$EVM = \sqrt{\frac{1}{N} \frac{\sum_{i=1}^N \|y_i - x_i\|^2}{\|y_{\max}\|^2}} \quad (20)$$

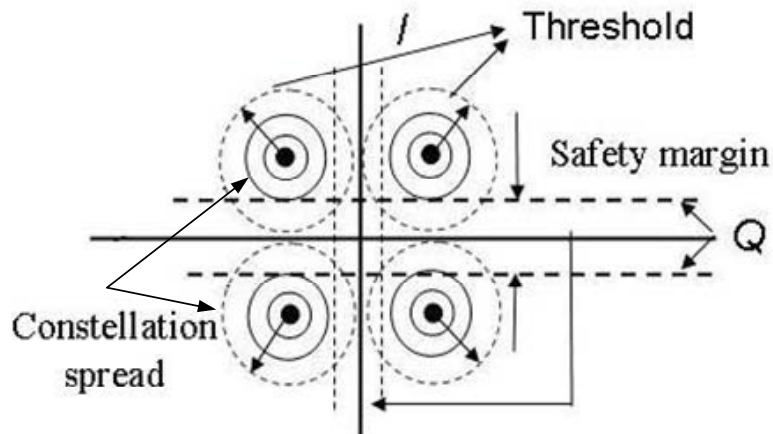
where  $y_i$  and  $x_i$  are the received and ideal complex modulated data ( $I+jQ$ ),  $y_{\max}$  is the outermost data point in the constellation, and  $N$  is the number of data points used for computation. In case of quadrature phase shift keying (QPSK),  $x_i$  gives one of “1+j”, “1-j”, “-1-j” and “-1+j” as known data. System-level simulations were performed to determine the correlation between the EVM and BER under different channel conditions and system non-idealities, and are plotted in Figure 58.





**Figure 58: EVM vs. BER relationship [90].**

The QPSK and QAM-16 modulation schemes were employed for evaluation purposes and about  $10^5$  bits were transmitted and received. The different values of EVM and BER are obtained by perturbing channel conditions. From the plots it is observed that in general an increase in EVM is associated with an increase in BER, and vice versa. An upper bound on the BER specification can then be translated into an upper bound on EVM. For example, if BER bound is set at  $5 \times 10^{-4}$ , the corresponding mean EVM bound for the QPSK can be approximated to be about 35%. A graphical illustration of the QPSK encoded symbols is shown in Figure 59. As the channel conditions and receiver performance degrade, the constellation points for each symbol lie inside circles of increasing size, corresponding to increasing EVM. When the circles cross the horizontal and vertical constellation boundaries, the received symbols are decoded incorrectly and bit errors occur. The objective of the EVM based feedback control is to force the receiver to operate just within the boundary of this threshold value (largest circle) irrespective of channel conditions.



**Figure 59: QPSK constellation spread with varying channel conditions.**

#### **6.3.1.2 Input Signal Scaling and Voltage Adjustment**

Dynamic voltage scaling benefits from the margins in the circuit design, workload and latency constraints and allows power to be traded for the quality in the circuit. Due to the quadratic relationship between power and supply voltage, significant power savings can be achieved if the circuit operates at lower supply voltages. If we define “*critical voltage*” as the voltage required for the correct operation of the circuit then in LSB-first architectures, reducing supply voltage lower than the critical supply voltage degrades the circuit performance catastrophically because of errors in the MSB bits. This drastic drop in circuit performance impedes efforts to try to operate the circuit below the critical supply voltage levels. If gradual performance degradation can be achieved with voltage scaling below the critical supply voltage, then the DSP circuit can be made to operate at an acceptable performance level but with much lower power consumption. In the following, we explain how signal scaling reduces the chances of critical path excitation in DSP arithmetic circuits.

Any downward signal scaling by a factor that is a power of 2 causes the data to be shifted right by an appropriate number of bits. As long as sign extension is performed correctly in 2’s

complement arithmetic, the effect is to reduce the circuit critical path by the number of least significant bits that is “shifted out” of the arithmetic computation. This has two implications: (a) the supply voltage can now be reduced by an amount that is proportional to the reduction in the circuit critical path length (the critical path corresponding to a given scaling factor is called the *active critical path* for that scaling factor) and (b) the performance of the DSP circuit is degraded gracefully since only the least significant bits of DSP computation are eliminated to trade off performance for lower power consumption.

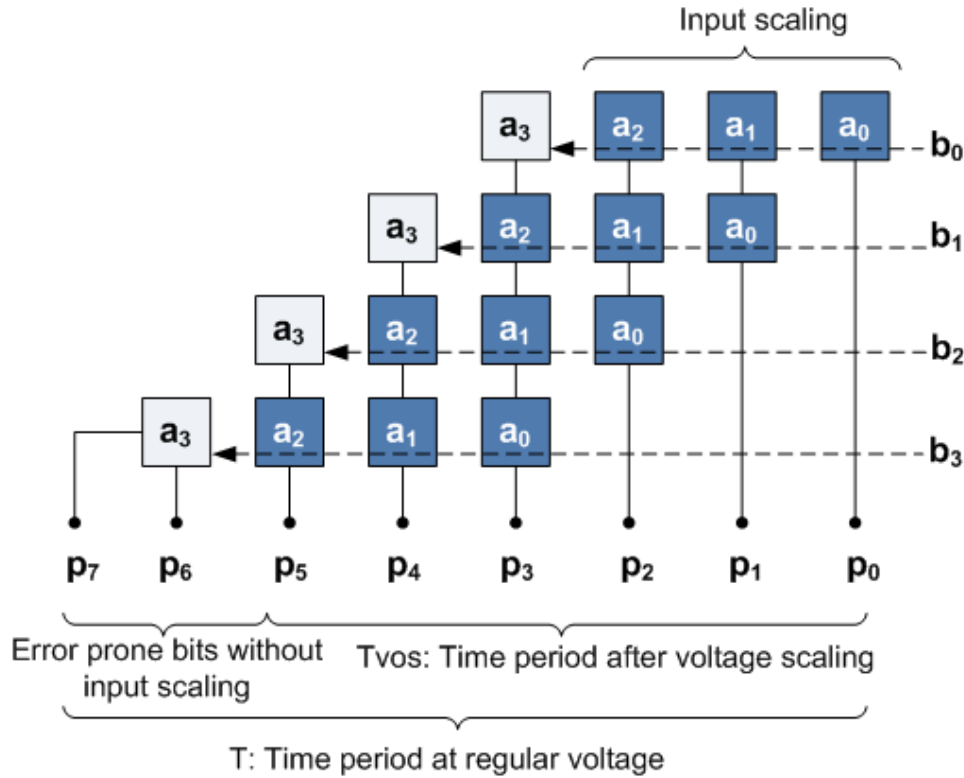
For a simple illustration of the proposed concept, consider multiplication of two 4-bit binary numbers, as shown in the Figure 60. The critical path of such a multiplication is  $2^{n-1}$ , where  $n$  is the number of multiplier/multiplicand bits. Now, if multiplicand  $b$  is scaled by a factor 2, i.e., right shift of one bit, the active critical path of such scaled multiplication will reduce to  $2^{n-1}-1$ . Clearly, there is a reduction in the critical path length of the multiplier with input scaling.

0110	a		0110	a	
0110	b	—Scaling→	0011	b/2	
<div style="display: flex; justify-content: space-between;"> <div style="text-align: right;">             0000              0110              0110              0000           </div> <div style="text-align: left;">             0100100           </div> </div>			<div style="display: flex; justify-content: space-between;"> <div style="text-align: right;">             0110              0110              0000              0000           </div> <div style="text-align: left;">             0010010           </div> </div>		Results in Reduced length Critical Path

**Figure 60: Critical path reduction with signal scaling.**

The application of the concept is explained with the array multiplier example as shown in Figure 61. Let  $T$  be the time period required to meet the critical path constraint of an array multiplier for a prescribed supply voltage. As the supply voltage is reduced, MSB bits incur errors due to increasing circuit delay. However, scaling the input (multiplicand) in the array multiplier

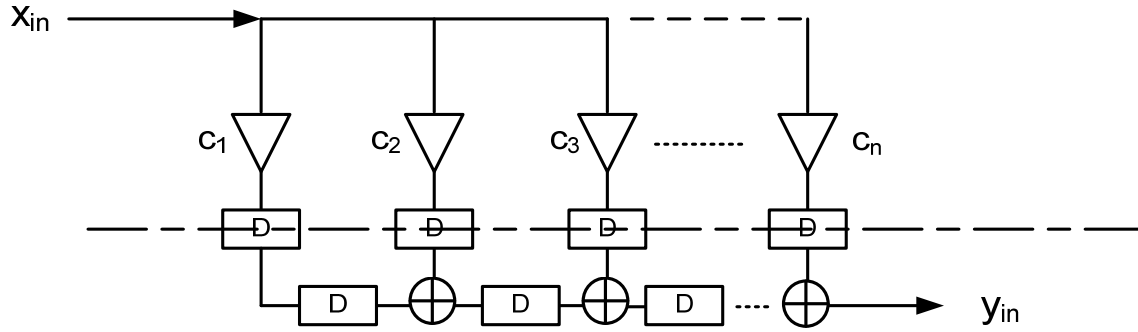
decreases the effective circuit critical path length, allowing the MSBs to be computed correctly at the cost of increased round-off/truncation errors. Consequently, no MSB errors occur while performing wordlength (W) and corresponding supply voltage ( $V_{dd}$ ) scaling. This concept holds in Booth multipliers as well (most commonly implemented multiplier architecture in DSP) as the number of partial products that need to be added to produce the correct output will reduce by signal scaling.



**Figure 61: Input (Voltage) scaling for low-power operation in an array multiplier.**

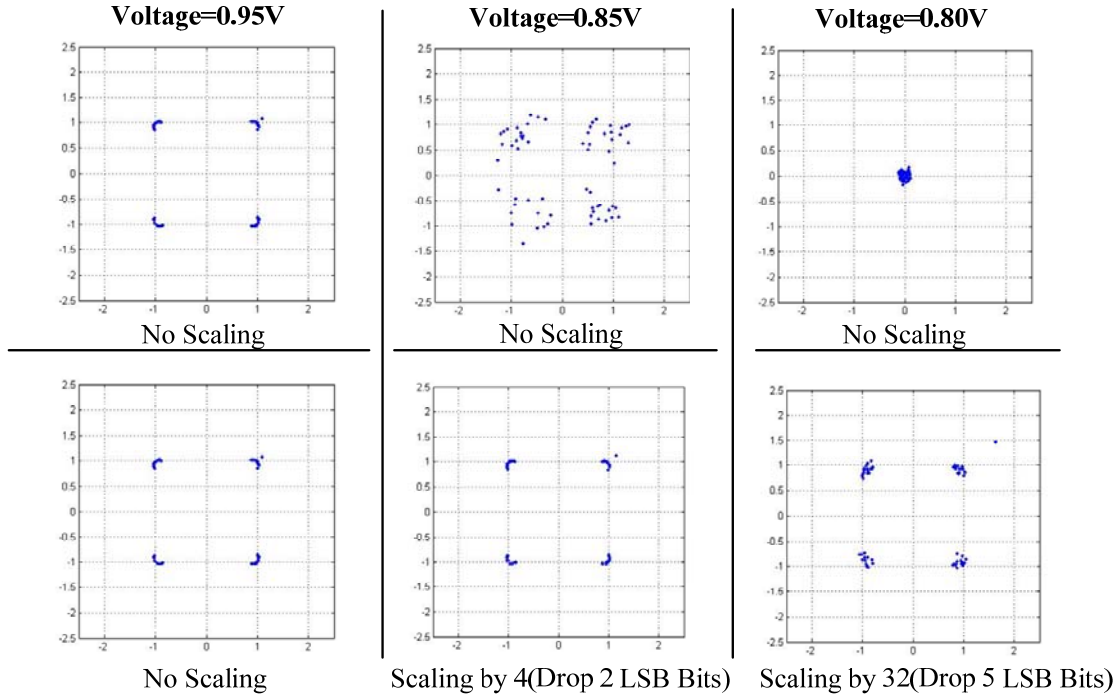
Figure 62 shows an  $n$ -tap FIR filter implemented in transpose-form. The critical path of the pipelined FIR filter is  $T_M$ , where  $T_M$  is the multiplier time delay. The filter coefficients  $C_n$  are fixed in such a filter. The input to the filter is scaled down by a factor of  $2^\alpha$  (shifting right and dropping  $\alpha$  LSB bits), therefore reducing the active critical path lengths in the filter. This allows lowering the operating voltage for the circuit. Since, input scaling by  $2^\alpha$  reduces the active critical path by  $\alpha$  bits, the supply voltage can only be lowered till we do not exceed the timing

requirements of the new active critical path, as this will ensure that no MSB bits get corrupted because of the reduced supply voltage. The data out of the filter is scaled up by the same scaling factor  $2^a$ . The system performance degradation resulting from the dropping of LSB bits is determined using the EVM of the demodulated signal. If signal quality is good (EVM is low), the noise performance of the filter is intentionally degraded by increasing input signal scaling. If the received signal quality is poor (large EVM), the EVM block starts up-scaling the input and adjusts the system supply voltage accordingly.

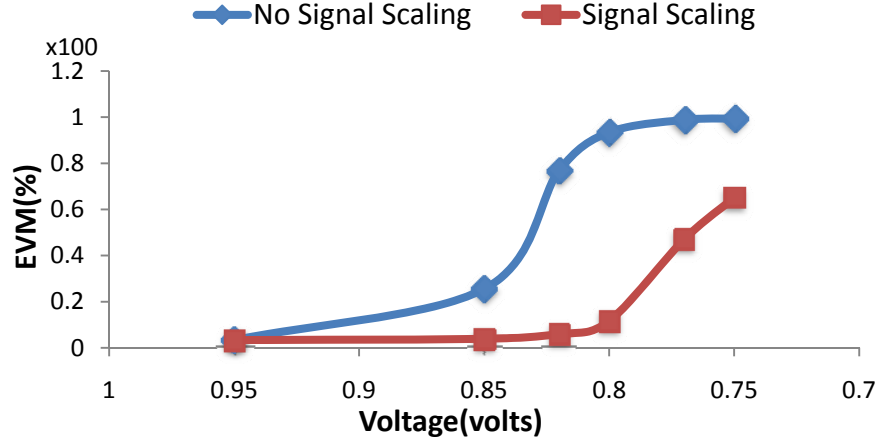


**Figure 62: Transposed form pipelined FIR filter.**

Figure 63 compares the spread in the QPSK constellation points of an OFDM modulated signal with simple voltage scaling and joint W- $V_{dd}$  scaling in a 16-bit 9-tap LPF. The corresponding performance degradation is shown in Figure 64.



**Figure 63: (Top Row) QPSK constellation points with voltage scaling alone. (Bottom Row) Constellation points with combined W- $V_{dd}$  scaling.**



**Figure 64: EVM degradation with voltage scaling and combined W- $V_{dd}$  scaling.**

Independent input scaling and voltage control of receiver modules ensures fine control of EVM degradation and allows better circuit power savings. Abrupt degradation in signal quality can result in corrupted DSP data if the feedback compensation system does not react rapidly. However, in wireless communication systems, protocol level robust design techniques ensure

that only correct data is retained for application level processing. In the worst case scenario, a packet error can result in retransmission of data.

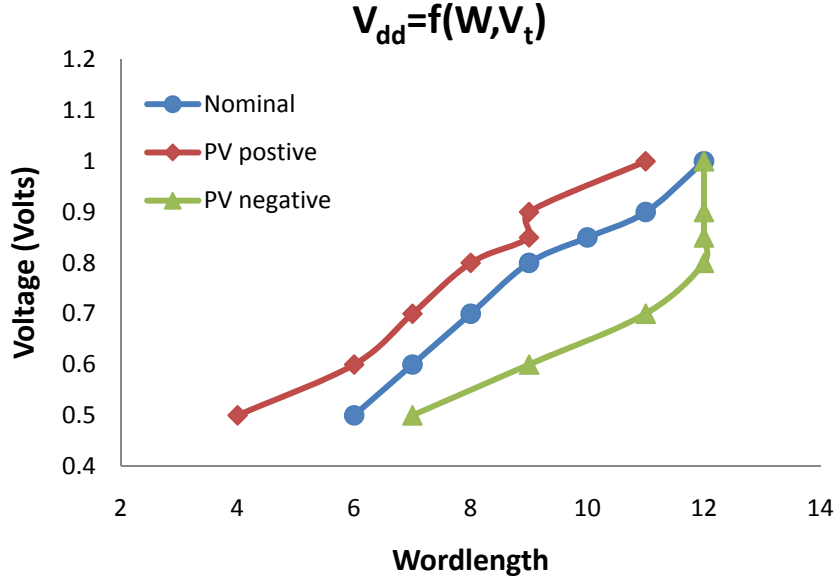
### **6.3.1.3 Signal Scaling and Supply Voltage Control**

The implemented feedback control continuously monitors the quality of the demodulated signal by measuring EVM. In order to operate the receiver within the prescribed limits of bit error rate, the receiver operates within a given range of EVM values determined by extensive simulation of the wireless transmission protocol. If the EVM value is below the prescribed threshold, the feedback control degrades system performance by input scaling. For every scaling level, the filter is set to a lower voltage level determined by a lookup table that is obtained via prior simulation and calibration experiments. Since, we know the effect of input scaling on the length of the active critical circuit paths and the effect of voltage scaling on the delay characteristics of the critical paths, a lookup table is constructed that determines different supply voltage levels as per the timing requirements of the requisite active critical paths. The advantage of such a lookup table, having wordlength and corresponding voltage entries is that it ensures that no MSB bits get corrupted due to signal scaling and supply voltage modulation.

### **6.3.1.4 Effects of Process Variations**

Silicon process variations cause the performance characteristics (delay, power, etc) of manufactured devices to vary from their nominal values thus changing the wordlength and supply voltage relationship as defined in previous section. For example, if the delay of the circuit increases, then a higher voltage level is necessary to meet the timing requirements for a particular wordlength. Therefore, we redefine the supply voltage of each module as a function of wordlength and its estimated  $V_t$  value (obtained from a critical path delay test performed post-manufacture). For the critical path of 12-bit adder, supply voltage is shown as a function of

wordlength and  $V_t$  variation in Figure 65. The nominal curve depicts the wordlength and voltage relationship without any process variation in the device. For the  $V_t$  variation of 20%, PV\_positive and PV\_negative curves (as shown in the figure) represent the change in  $V_{dd}$  and W relationship for the positive and negative variation respectively.



**Figure 65: Supply voltage as a function of W and  $V_t$ .**

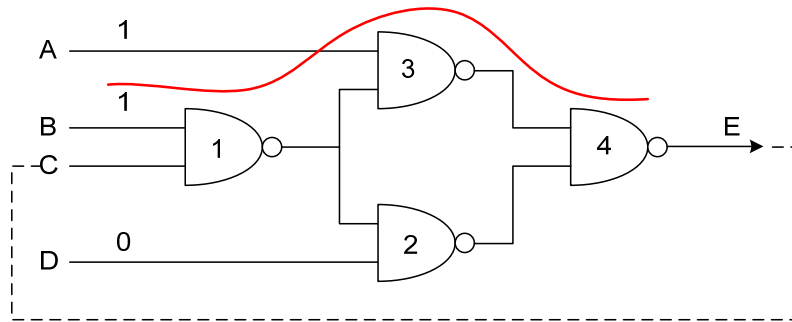
Under process variations, faster modules have higher leakage power dissipation and vice versa. Hence, for system power optimization it makes sense to reduce the supply voltage of a faster module at a *faster* rate than the supply voltage of a slower module for the same effective reduction of wordlength. The process variability in the circuit is predicted using POTTs as a onetime procedure as explained in next section.

### 6.3.1.5 Path Oscillation Timing Tests (POTTs)

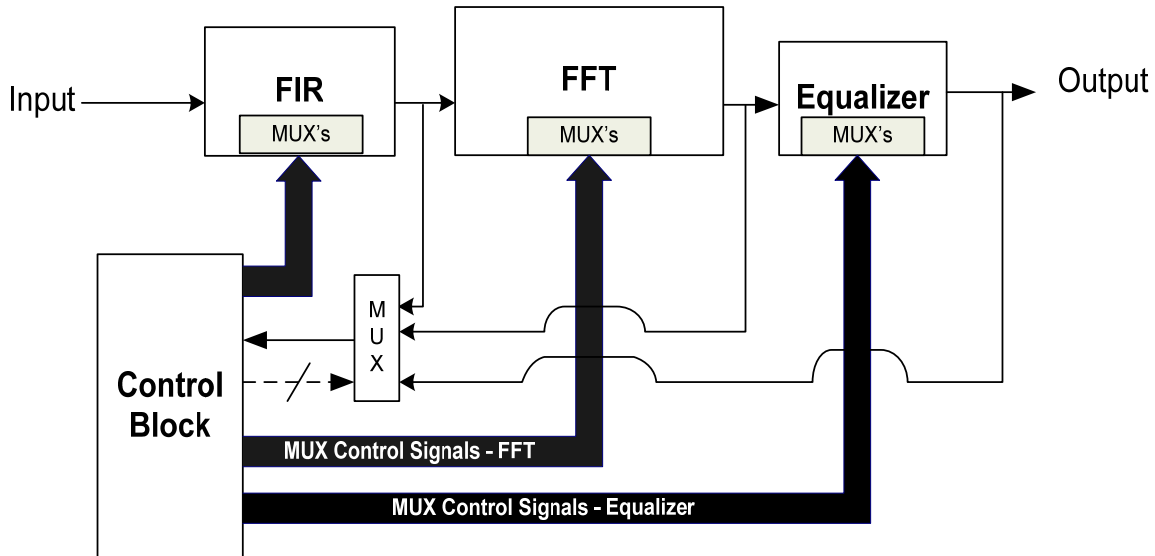
A delay test technique is used to predict the process variation in the system. The technique proposed in [79] consists of sensitizing a path in the circuit under test and incorporating it into a ring oscillator to test for the delay of the path and stuck-at faults. Assuming no stuck-at faults in the system and exploiting the delay test measurements, we compare the delay test information to



the nominal delay of the path to estimate the process variation. To approximate the intra-die variations, the tests are independently applied to the critical paths of the modules and it is made sure that there are an odd number of inverters in the target path to guarantee oscillations. The oscillation frequency of the path is measured through a counter and is used to predict the path delay. Under process variations, faster modules have higher leakage power dissipation and vice versa. Hence, it is possible to estimate the leakage power dissipation of a module by determining its critical path delay. In fact, the *relative* delay values of two or more different modules can be used to estimate their *relative leakage power values* as well.



**Figure 66: Path sensitization for oscillation test.**



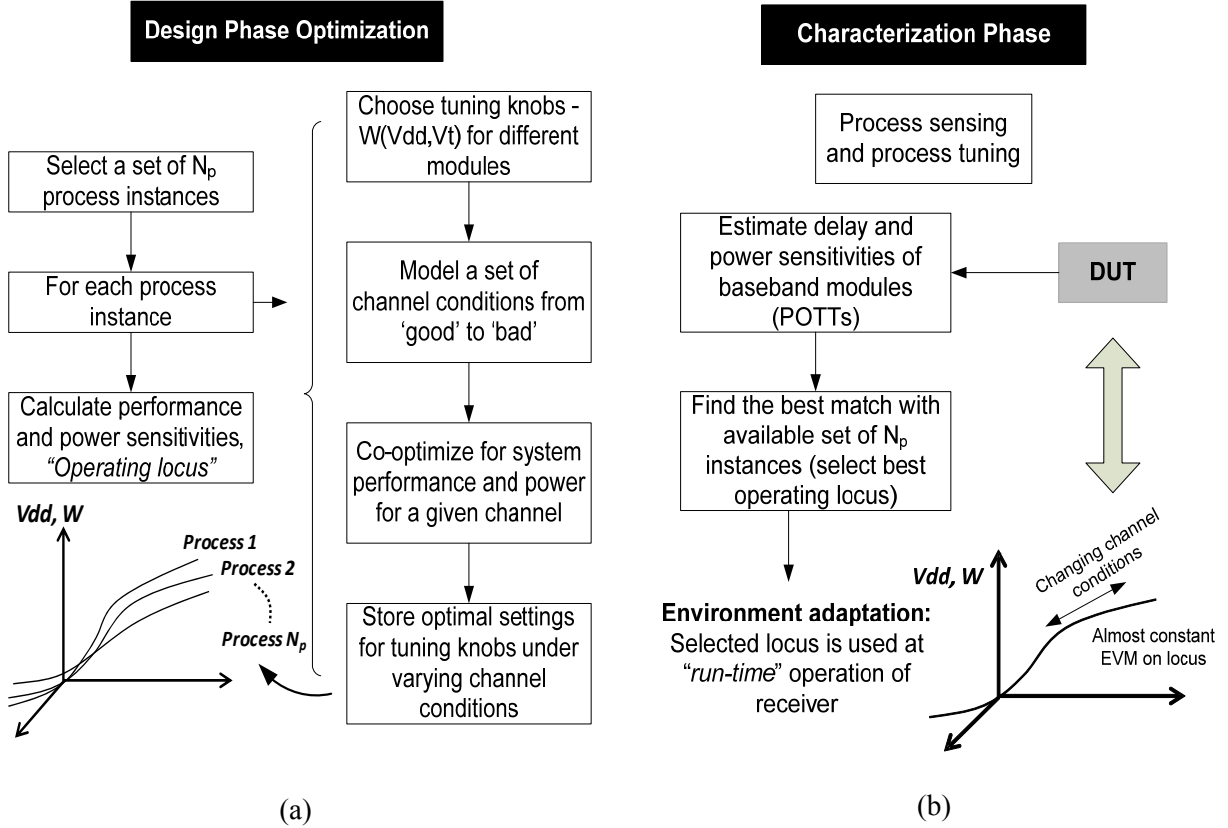
**Figure 67: Block level representation of POTTs scheme.**

The test method is demonstrated using the simple logic circuit shown in Figure 66. To sensitize a path, off path inputs of all gates directly connected in the path should be set to non-controlling values by properly setting the primary inputs. In the example circuit, path Gate1-Gate3-Gate4 is sensitized by setting the circuit inputs as  $ABCD = [11E0]$  and with non-inverting feedback from E to C. A block diagram of the POTTs implementation in the complete system is shown in the Figure 67. The MUXs in the modules are required to sensitize the critical paths, to be tested by disconnecting the inputs and applying the required input pattern. The outer MUX is used to complete the feedback loop. In our implementation, it is assumed that only one path is tested at a given time. A control block is used for measuring the oscillation frequency and controlling the input patterns for path test and module MUXs.

#### 6.3.1.6 Loci Based Operation: System Design and Characterization Phase

Design phase optimization procedure is summarized in Figure 68 (a). During design phase, extensive simulation and test results are used to calculate system *performance locus*. The locus provides optimal settings of the control parameters ( $V_{dd}, W$ ) for different receiver modules under varying channel conditions. For a set of system process variations, multiple loci are calculated and stored. A *low-power control algorithm* is used to calculate the control parameters for individual modules in such a way that for a minimum degradation in overall system performance, maximum savings in system power consumption are achieved for a given channel condition. Iterations of the control algorithm on a set of channel conditions result in performance locus. During system characterization phase, one time post-silicon timing test is run on the device to estimate the process variations. It is assumed that a fixed threshold voltage can be associated with each module under intra-die process variations. The timing test information is

used to map the device to one of the multiple loci stored in the system. That particular locus is then always used by the device for the run-time operation as shown in Figure 68 (b).



**Figure 68: (a) System design phase. (b) System characterization phase.**

In the following system design and characterization phase algorithm is summarized.

#### Design Phase:

1. Assume  $K$  modules to be optimized in the receiver.
2. Define a set of  $P$  process variations obtained by statistical sampling experiments driven by intra-die process variability statistics.
3. Define a set  $C$  with varying channel conditions from good to bad.
4. Calculate and populate  $W(V_{dd}, V_t)$  lookup tables for  $K$  modules under  $P$  process variations.

5. For K modules and a set of P process variations, build a set  $N_p$  with  $P!/(P-K)!$  combinations.

Each combination of set  $N_p$  corresponds to different across module variations in the baseband receiver.

6. For every entry of set C:

***Function low\_power\_control\_law( )***{

Set wordlengths and voltages of modules to a set of default “starting” values. For the current set of voltages and wordlengths calculate the gradient matrix (M);

$$M = \begin{bmatrix} \frac{\partial P}{\partial W_{FIR}} & \frac{\partial P}{\partial W_{FFT}} & \frac{\partial P}{\partial W_{Eq.}} \\ \frac{\partial EVM}{\partial W_{FIR}} & \frac{\partial EVM}{\partial W_{FFT}} & \frac{\partial EVM}{\partial W_{Eq.}} \\ \frac{\partial P}{\partial EVM_{FIR}} & \frac{\partial P}{\partial EVM_{FFT}} & \frac{\partial P}{\partial EVM_{Eq.}} \end{bmatrix}$$

Use the matrix M to find the wordlengths of the modules (FIR, FFT, Equalizer) that result in *minimum power consumption* while ensuring the *system EVM is less than or equal to the maximum allowable EVM* for prescribed maximum BER value. This is done using gradient search with a cost function that minimizes system level power while maximizing EVM to the maximum allowed limit. Save final wordlengths and corresponding supply voltages of the individual modules (FIR filter, FFT and Equalizer).

}

***Optimal locus = Results in k-dimensional locus determined by above control parameter values for varying channel conditions (entries of set C).***

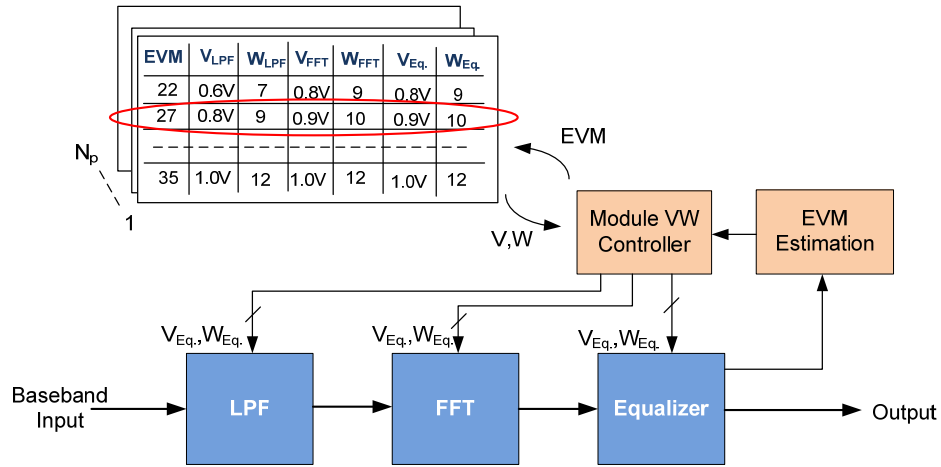
7. Repeat Step 6 for every entry of set  $N_p$ . Each invocation results in a locus that is stored in a lookup table. Hence, the final output is a set of loci, *each locus corresponding to a K-*

*dimensional set of POTTs results, called the **process vector PV**, one for each of the K baseband modules.*

**Characterization Phase:**

8. Run one time post-silicon POTTs on the device to determine process variation effects.
9. To accommodate process variations, the obtained vector of POTTs results is matched against the *process vector* corresponding to each locus in the set of stored loci. The locus corresponding to the *least squares timing vector match* is selected as the locus of choice for the baseband processor concerned.
10. This locus of control parameters is then used at run-time of the device to modulate baseband module noise to minimize power across varying channel conditions.
11. Maintain system performance above defined EVM value to meet the end application performance requirements.

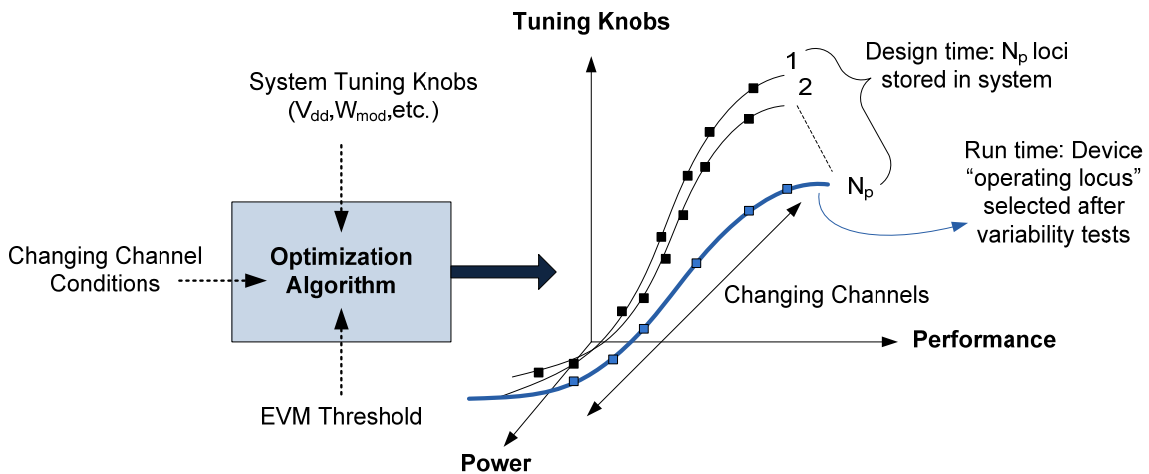
Figure 69 shows the graphical representation of a channel driven locus based system that modulates the settings of the control parameters according to the current channel conditions and process variations in the system. In this work, we calculated the power savings in two type of systems; first, which allows voltage and wordlength adjustment on individual modules; second, the system that allows only one voltage setting for the whole baseband receiver.



**Figure 69: Signal quality based module-level “voltage, wordlength” scaling.**

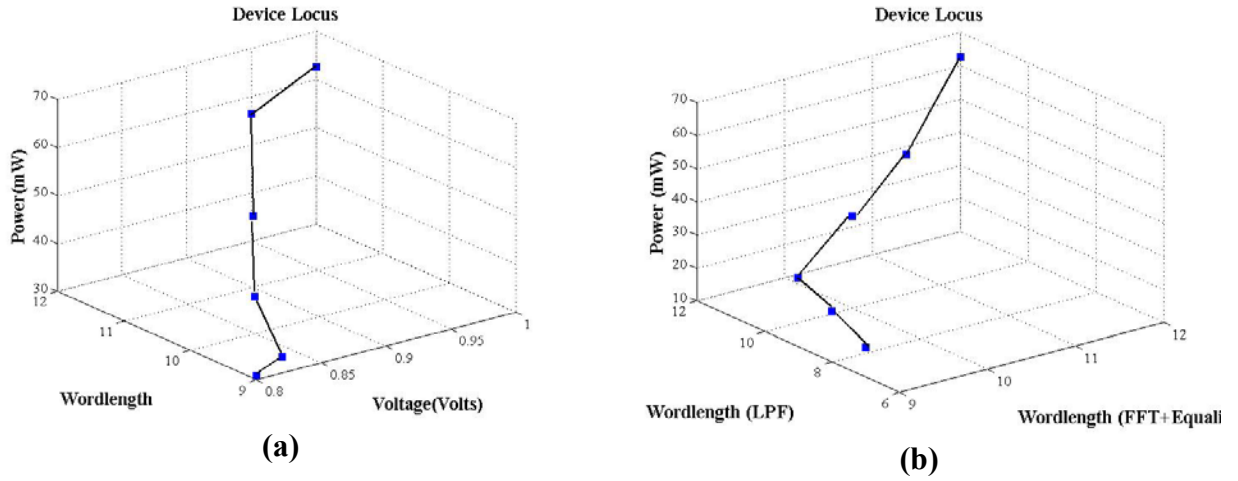
### 6.3.1.7 Evaluation

System model as explained in Chapter 3 is used for the experimental results in this section. The design time loci calculation and run-time device operation on a given locus can be summarized as in Figure 70. The optimization algorithm calculates the tuning knob values based on the channel conditions and the given EVM threshold, resulting in a locus. The algorithm is repeated for a number of intra-die process variations resulting in multiple loci. After device characterization one locus is selected to be used for run-time operation.



**Figure 70: Optimal power locus for a wireless receiver under changing channel conditions.**

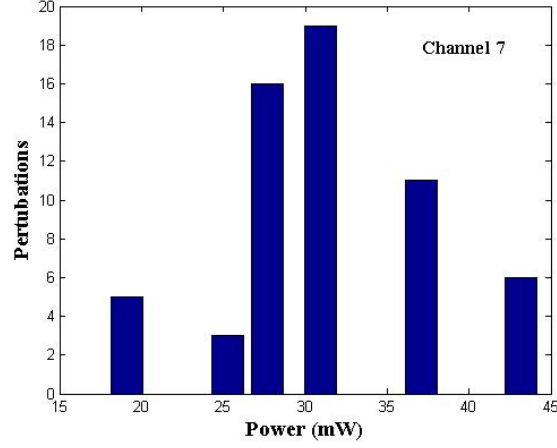
The device locus of a nominal device, i.e., no process variations and with symmetric voltage/wordlength scaling on the baseband modules is shown in Figure 71(a). Also, the device locus with independent voltage/wordlength control on modules is shown in Figure 71(b). It is obvious from the figures that more power is saved with independent voltage/wordlength control of the modules, as different modules have different impact on system performance. The loci shown in the figures are calculated for QPSK modulation that can tolerate EVM up to 35%. Under various channel conditions, EVM values are almost constant on the locus operating points.



**Figure 71: Device locus for run-time operation (a) symmetric voltage/wordlength modulation of modules (b) independent voltage/wordlength scaling on baseband modules.**

Under process variations supply voltage relationship with wordlength changes thus affecting the total power consumed in the circuit. Figure 72 shows the power distribution of the baseband receiver for a given channel with process variations of up to 20% in the modules. A set of 60 instances are perturbed for the LPF, FFT and equalizer. The application of the control law results in optimal  $V_{dd}/W$  settings for these modules that result in optimal performance efficiency. The amount of the power savings is dependent upon the process spread in the device, therefore

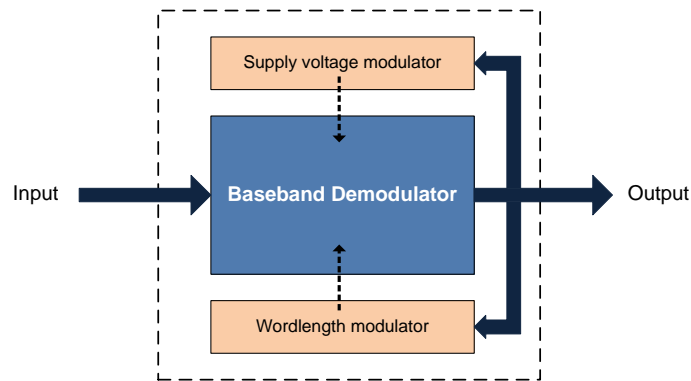
determination of the process variation in a device enables setting the optimal control parameter values and thus resulting in optimal power savings.



**Figure 72: Power distribution of the OFDM baseband receiver under 20% process variations.**

### 6.3.2 Dual Nested Loop Architecture

A dual feedback mechanism is proposed that tailors the operation of each device to changing channel conditions (*dynamic*) and its manufacturing process parameters (device specific and *static*) for minimum power operation (see Figure 73). The objective of the proposed real-time feedback system is to allow fine-grain power control at the system level using the proposed closed loop dual feedback as opposed to open loop control using lookup tables (loci-based operation) that requires extensive calibration and guard banding.



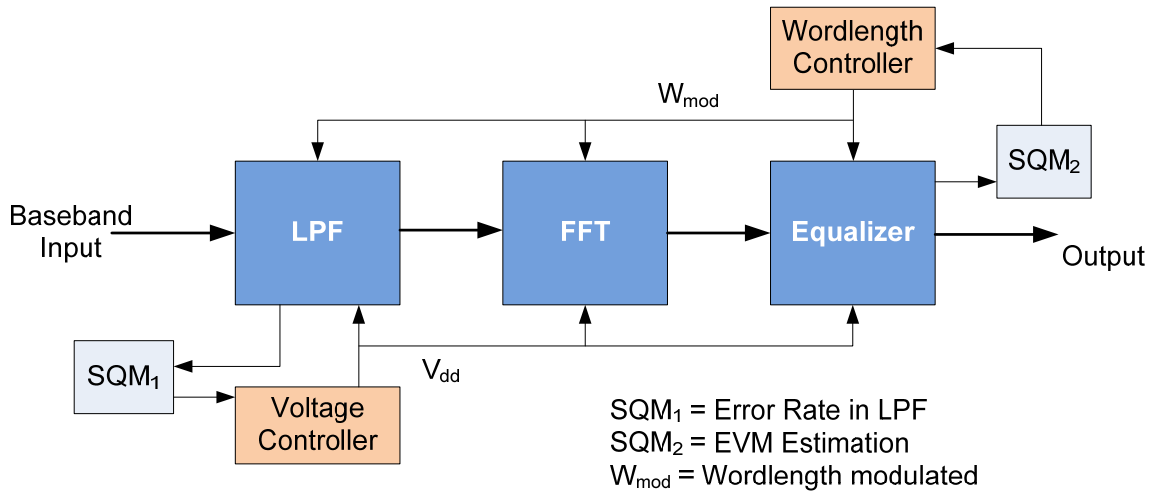
**Figure 73: Proposed real time feedback architecture.**



### 6.3.2.1 Proposed Power Control Methodology: Overview

The loci based adaptive baseband demodulator architecture presented in Section 6.3.1 results in significant power savings without compromising the end signal quality requirements. However, the architecture performance depends on the granularity with which channel conditions are quantized and how accurately the operating locus is calculated. Therefore, savings in power consumption are higher with a finer channel quantization granularity.

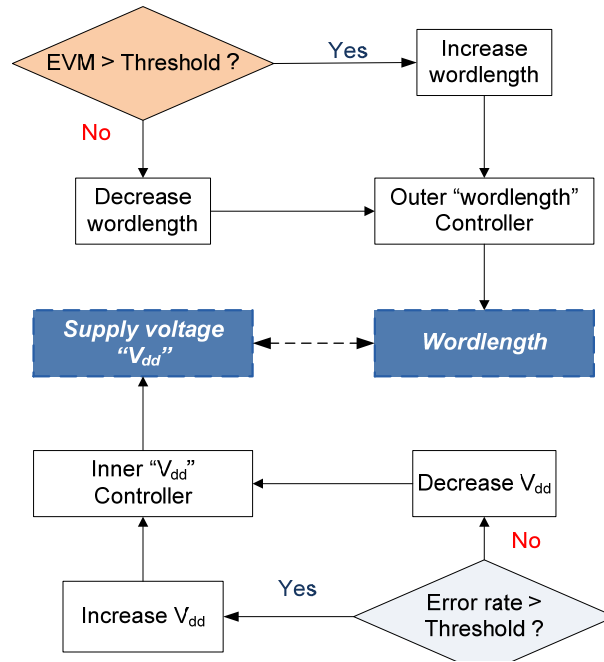
In this work, a real-time dual feedback mechanism is presented (as shown in Figure 74) that performs performance vs. power tradeoff under process variability and changing wireless channel conditions without the need to store tables of pre-computed loci corresponding to different process conditions. Another benefit of the scheme is that it is not necessary to assess the process for the device under test (DUT) concerned using on-chip tests (e.g. path based oscillation test - POTTs) or the use of on-chip sensors.



**Figure 74: Proposed real-time dual feedback control architecture.**

The core idea is to use feedback loops, which modulate system wordlength  $W$  and supply voltage  $V_{dd}$  independently in real-time but in a nested fashion. The EVM value is computed across several frames of data (about 1024 received symbols) in real-time, is updated dynamically

and is computed with latency in the order of 10s of milliseconds. The EVM value is used as input to an “outer loop” controller that determines the wordlength of the processor. Note that lower wordlength results in lower signal SNR and thereby a higher EVM value. The *EVM* adaptation metric represents the cumulative sum of the quality of transmission, the channel quality degradation and the quality of signal reception. It is shown in previous sections that EVM has a strong correlation with BER. As the wordlength changes, an “inner control loop” adapts the supply voltage to the current selected wordlength. This is done by decreasing the supply voltage for reduced wordlength (and vice versa) until errors occur in the most significant bits of computation (MSB errors). These errors are then compensated accurately using checksum codes applied to the underlying DSP algorithms. The nested loop framework is shown in Figure 75. To allow computation to proceed without loss of throughput, some inaccuracy in the compensation process is allowed. Since a higher error rate has an adverse effect on EVM, the inner loop control mechanism is designed to always maintain a low predetermined MSB error rate. In this manner, the “outer” and “inner” control loops interact with each other to always allow the lowest power operation for any channel condition. The use of two nested feedback loops to adjust  $V_{dd}$  and  $W$  of the individual blocks at run-time, without any pre-calibrated lookup tables makes this scheme independent of process variations as well.



**Figure 75: Dual nested loop control strategy.**

Input signal scaling based voltage scaling is a very useful technique for graceful degradation of system performance in exchange for considerable power savings. Since we know the effect of input scaling on the length of the active critical circuit paths and the effect of voltage scaling on the delay characteristics of the critical paths, a lookup table is constructed that gives the supply voltage values corresponding to different values of the wordlength  $W$ . As  $W$  is reduced, the circuit critical path is reduced and hence the multiplier/adder can be operated at a lower supply voltage level saving power. The use of such a lookup table, with  $W$  and corresponding  $V_{dd}$  entries ensures that no MSB bits are corrupted due to signal scaling and supply voltage modulation. However, process variations cause the performance characteristics (delay, power, etc) of the manufactured devices to vary from their nominal values. If the delay of the circuit changes due to process variations, the entries of the lookup table as described in the previous section are no longer optimal from a power consumption perspective. For example, if the delay of the circuit increases, then a higher voltage level is necessary to meet the timing requirements

for a particular wordlength  $W$ . Therefore under process variations, the  $W$  and  $V_{dd}$  values stored in the lookup table are incorrect and may result in MSB errors. Hence, the lookup tables *must contain the worst-case supply voltage value* for each entry of the wordlength  $W$  if process variations are handled via the lookup table approach. This generally results in larger than necessary power consumption for devices with nominal process values. On the other hand, multiple lookup tables (loci as shown in Figure 69) may be constructed for devices with different delay. However, this requires the incorporation of on-chip hardware (sensors) for delay measurement and incurs extra design cost.

To avoid the use of the lookup tables, the supporting calibration mechanisms and on-chip hardware (sensors) we propose instead to incorporate a simple error checking mechanism into the hardware that lets the supply voltage adapt dynamically to the wordlength  $W$  using closed loop feedback control as follows. First signal scaling is performed to make the effective wordlength equal to the selected value of  $W$ . Then, closed loop feedback control is used to reduce the supply voltage until MSB errors occur. These errors are compensated and the supply voltage is marginally increased and the process is repeated causing the supply voltage to throttle around a value just large enough to ensure that the MSB error rate is below a prescribed value.

### **6.3.2.2 Guided Probabilistic Compensation for Low-Power Digital-Filters**

As presented in Chapter 4, VOS combined with ANT can be used in digital filters for low-power operation. Error detection is performed using checksum codes applied to state variable representations of digital filters. In the GPC architecture, every state is double latched with a regular and a delayed clock. Latches operating with the delayed clock are called shadow latches and are used for flagging an erroneous state. Whenever an error is detected by the checksum codes, error magnitude is divided by the number of erroneous states and the resulting value is

applied only to the erroneous states as the error compensation. In case, only one state is erroneous, the error value is used to compensate only that erroneous state. If two states are found to be erroneous, then the error value is divided by 2 and both erroneous states are compensated with the same (error/2) value. No compensation is performed for the states that are not flagged erroneous by the shadow latches. The GPC technique compensates the errors only in the erroneous states with the overall goal of system noise minimization. GPC is explained in detail in Chapter 4.

### **6.3.2.3 Signal Quality Metrics**

Two signal quality metrics are needed in order to modulate system wordlength and supply voltage independently for maximum power savings. *EVM as explained in Section 6.3.1.1* is used to modulate the system wordlength. System error rate is chosen as the second signal quality metric and is explained below.

### **6.3.2.4 Error Rate Metric for Modulating Supply Voltage**

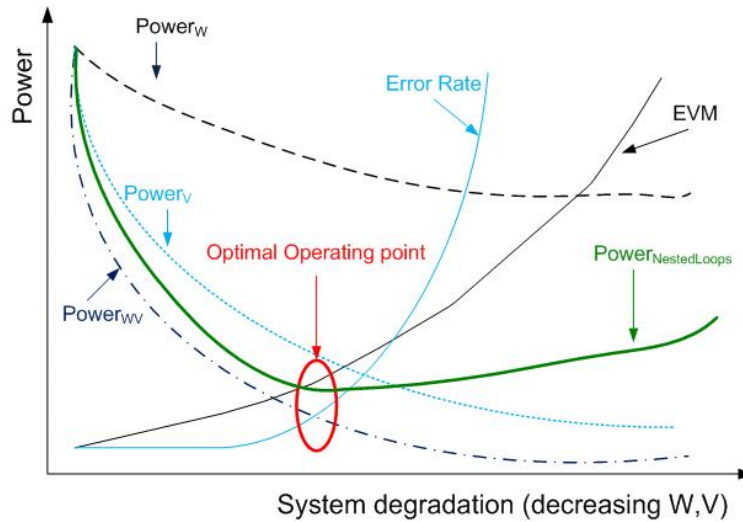
Errors in the system are monitored for a certain time period using a counter to calculate error rate. The counter is reset to zero after every counting period. If the error rate is low then the computations in the circuit are completing too quickly (critical paths are not being excited) and/or the error correction is working well enough to allow the supply voltage to be reduced further to save power. On the other hand, if the error rate is high then the filter components are not meeting the timing constraints and it is possible that the larger numbers of errors are not compensated accurately. In this case, feedback control increases the supply voltage to bring the error rate to a lower value that allows accurate error compensation.

### 6.3.2.5 Nested Loop Architecture

In the proposed *dual nested loop architecture*, two signal quality metrics are used to control wordlength and supply voltage of the DSP modules. At the higher level, EVM driven feedback control is used in an outer control loop to control the wordlength of the baseband filter. The generated wordlength value is passed to an inner control loop that adjusts the supply voltage of the filter in response to changing wordlength requirements in such a way that the MSB error rate of the filter is always below a prescribed value and all errors are compensated as accurately as possible. It is assumed that same arithmetic modules are used in all the baseband functions. Therefore, the error rate under voltage overscaling can be monitored in only one module (LPF) and the system voltage can be adjusted accordingly across all baseband functions. In case, different critical paths exist in different modules, the error rate should be calculated independently for all modules and their supply voltages adjusted according to their individual error rates. For a given system, the maximum allowable EVM value and error rate is defined for the outer and inner control loops, respectively. At run-time, the feedback control loops independently strive to operate the system at the defined signal quality limits, thus saving considerable system power under good channel conditions. The key advantages of having two independent control loops working within defined constraints are: (1) there is no need to calculate and store loci (lookup tables for  $W$  and  $V_{dd}$ ) at design time, (2) no testing is necessary to determine the effects of process variations on different arithmetic modules and (3) nested loops can automatically adapt to changes in  $V_{dd}$  and  $W$  relationship under process variations.

The relationship between baseband  $W$ ,  $V_{dd}$ , error rate, EVM and system power is qualitatively illustrated in Figure 76. As  $W$  and  $V_{dd}$  are scaled together in a nested loop, system performance depicted by error rate and EVM starts degrading. As  $V_{dd}$  is scaled fewer critical paths complete within the stated clock period and results in increasing error rate.  $V_{dd}$  scaling

below a certain critical level results in exponential rise in system error rate. Because error-compensation is performed just in the LPF in our implemented architecture, therefore we do not want to operate the system in the region of high error rate. Wordlength scaling results in power savings as lesser switching activity occurs in the MSB bits of the circuit but also causes an increase in system EVM. From BER vs. EVM relationship there is a maximum EVM limit for a given modulation scheme (QPSK, QAM-16, etc.). As a result, there is a tradeoff between power savings achieved in the system with  $W$ - $V_{dd}$  scaling and system performance. Power consumed by the compensation circuitry increases with increasing error rate thus negating some of the power savings achieved from voltage/wordlength scaling. Therefore, power savings achieved by the nested loop architecture is dependent upon limits of error rate and EVM defined in the system.



**Figure 76: The qualitative relationship between supply voltage, wordlength, EVM and error rate. The optimal operating point of nested loops is defined by the quality requirements of end signal.**

In any feedback system, stability is a main concern. To make sure that the proposed dual loop structure is stable, the minimum period for changing the wordlength equals to the time required for supply voltage step change. Also, checksum error rate value is set to a low value. This ensures that supply voltage loop follows the wordlength loop and supply voltage will be lowered

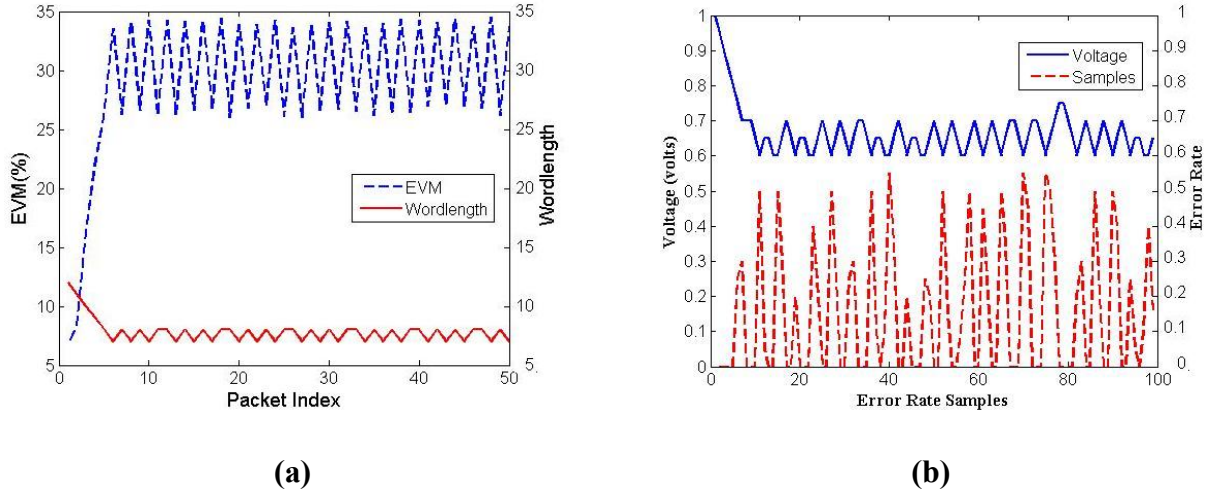
only if some bits have been dropped by the outer (wordlength) loop. Moreover, in case of sudden adverse change in channel conditions, wordlength and supply voltage is restored to the highest values to ensure that system BER does not increase more than the maximum allowed limit.

#### **6.3.2.6 Evaluation**

The OFDM transceiver model as described in Chapter 3 is used for simulation results. At the transmitter side, data encoding, data modulation (QPSK, QAM16) and IFFT are implemented in the floating point units. Different values of white noise, interference and multipath fading effects are added to model 14 different channel conditions (good to bad). In the receiver demodulator, a 15-tap LPF is implemented along with 128-point FFT and MMSE equalizer. For delay estimates of critical paths, circuit level simulations are performed in HSPICE at varying voltage levels (1-Volt to 0.55-Volt in voltage steps of 1 mV). These delay estimates are used to observe path delay errors under VOS in a logic level simulator. Power estimates are based on average voltage and wordlength levels in the system. In this work, proportional controller [91] is implemented to modulate the system wordlength and supply voltage according to their respective signal quality metrics.

Figure 77 shows the wordlength and voltage scaling performed by the control loops on QPSK modulated data at a given channel condition. Outer loop modulates the system wordlength (input scaling) with preset EVM value of 35%. Following the outer loop, voltage is scaled in the system by inner loop with a preset error rate of 10%. The control loops result in the system average wordlength of 7 bits and average voltage of 0.65 volts. It is worth mentioning here that the level of voltage scaling achieved in this experiment is not possible without the input scaling in the outer loop. For all the other experiments in the section, preset settings of control loops are limited at EVM of 30% and error rate of 5%.





**Figure 77: Fixed Channel (a) Outer loop - Wordlength modulation under EVM constraint. (b) Inner loop – Voltage modulation under Error rate constraint.**

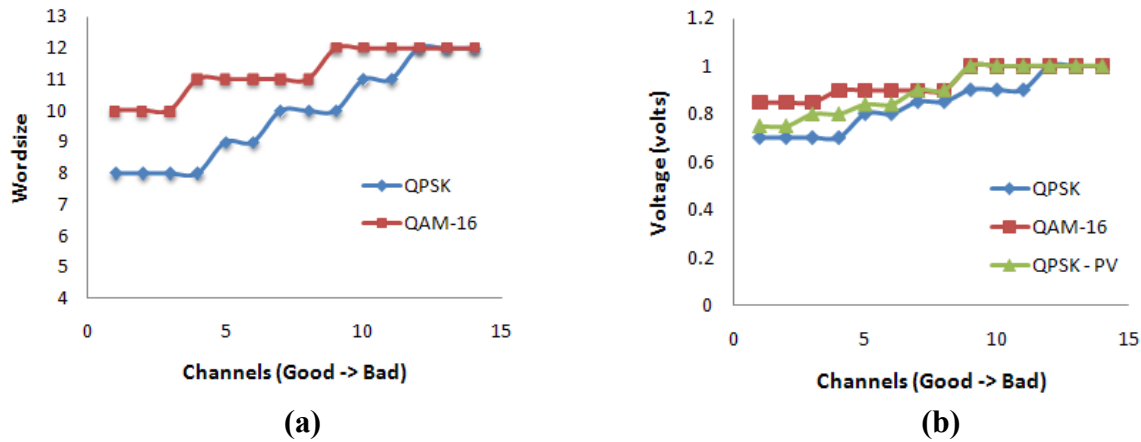
Power consumed in the system at different voltage levels (along with corresponding wordlength scaling) is summarized in Table 11. More power is consumed in the LPF as it is a 1/4x decimation filter i.e. runs at four times a higher clock as compared to rest of the demodulator circuit.

**Table 11: Power consumption in baseband demodulator under voltage scaling.**

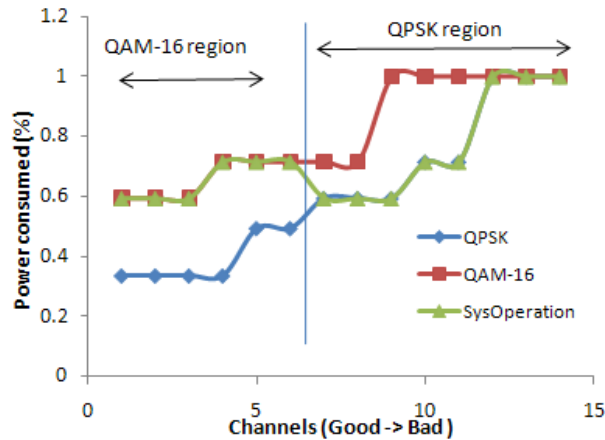
Voltage (volts)	LPF (mW)	FFT (mW)	Equalizer (mW)	Total (mW)
1	32.31	22.15	8.23	62.69
0.90	24.16	15.32	5.34	44.82
0.80	16.43	10.77	3.72	30.92
0.70	10.8	7.68	2.51	20.99
0.60	8.9	5.42	1.96	16.28

Figure 78 shows average system wordlength and supply voltage values for QPSK and QAM-16 modulation under varying channel conditions in the proposed architecture. Under process variations the nested loop architecture stabilizes at different voltage setting of modules (Figure

78(b)). This is an inherent benefit of nested loop architecture, i.e. system requires no onboard sensors or tests to adapt according to process variations. Power savings achieved by the proposed architecture for QPSK or QAM-16 under varying channel conditions is shown in Figure 79. In wireless communication, different data rates are used in the system depending upon the channel conditions. Higher data rate modulation scheme such as QAM-16 is used in very good channels. Therefore, SysOperation curve is a better representative of system power savings under varying channel conditions. It is evident from the graph that scheme results up to 40% power savings without any impact on system performance.



**Figure 78: System wordlength and voltage for QPSK and QAM-16 modulation, for different channel conditions. Process variations result in different voltage settings.**

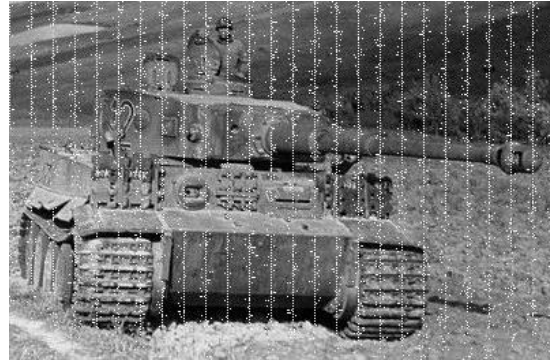


**Figure 79: Average power consumption in the baseband demodulator with the proposed nested loop architecture.**

Figure 80 shows an image received through the adaptive demodulator under different preset values. The greater margin in preset values results in higher power savings but degrades the system performance. To compare the results of the nested loop architecture with multiple loci scheme, 100 process perturbed instances were considered. Process variation of up to 20% was injected in the system. Difference in power consumed by the nested loop architecture and optimal loci ( $\Delta p$ ) is calculated and is plotted in an histogram (Figure 81). It can be noticed that dual nested loop architecture consumes slightly more power than the locus based architecture in most of the process instances. This is because of wordlength and voltage values oscillations in the nested loop architecture that may result in higher power depending upon modeling of channel conditions and feedback controller design. However, as compared to locus based technique, neither system loci are required to be stored nor timing tests (or chip sensors) are required to be performed to determine the process variation in the device. This power comparison does not consider the power consumed in the baseband DUT operations performed in the loci technique.

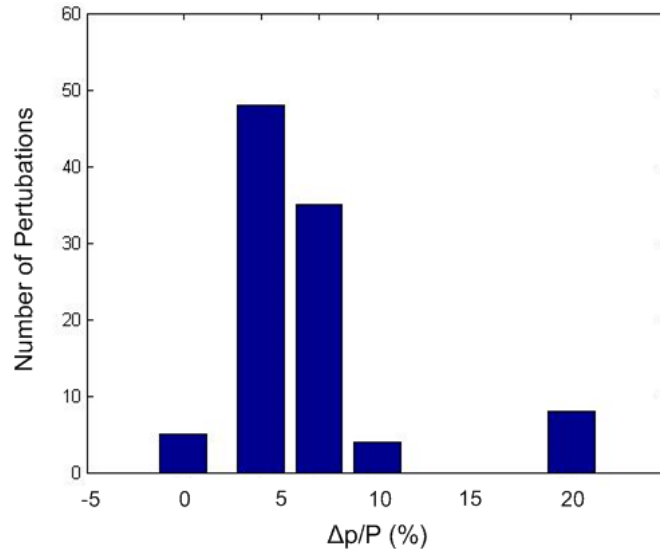


(a)



(b)

**Figure 80: (a) Image received at preset EVM=30% and error rate=2%. (b) Image received with EVM=35% and error rate=10%.**



**Figure 81: Power histogram under process variations.**

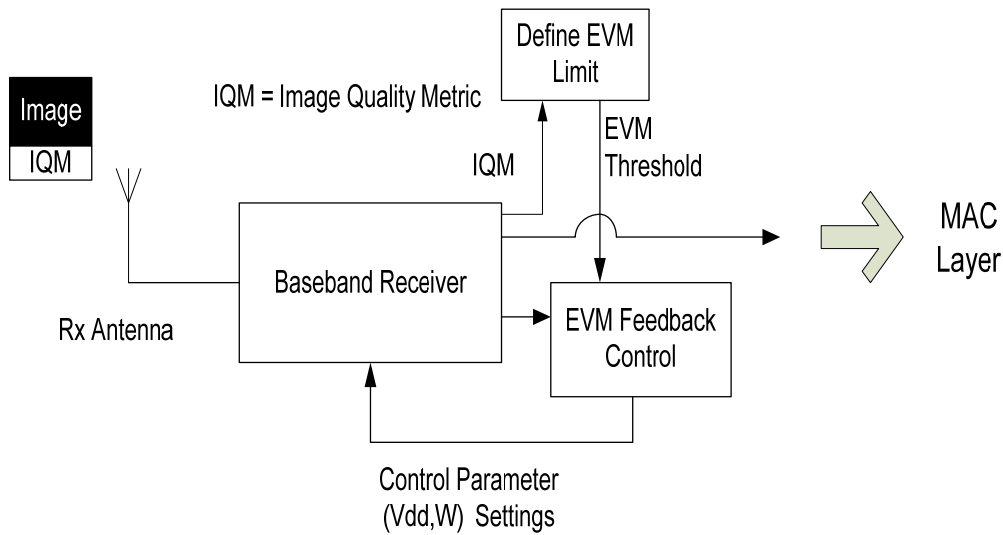
The proposed dual real-time feedback based design approach allows baseband demodulator of a wireless OFDM system to adapt dynamically to channel conditions as well as manufacturing process variations, with an end objective of low-power operation. It is shown with experimental results that the proposed technique can result in power savings up to 40%. An implementation of a better feedback control system can result in even improved power savings.

## **6.4 Application Driven Channel and Variation Tolerant Low-Power Baseband Design**

In this section, an application driven channel and variation tolerant low-power baseband processing technique is presented. The core idea is that the baseband processor power can be optimized by operating it at its lowest acceptable performance limit according to the requirements of the end application. Since performance is traded-off for power, lower the processor performance, higher will be the power savings. In this work, image transmission across a wireless link is studied and edge detection/centroid location is used as the end application. For power optimization, a good image is degraded more by the baseband processor as compared to a

bad image (before being fed to the edge detection/centroid location algorithm), as higher bit errors can be tolerated by the edge detection algorithm for a “good” image as compared to a “bad” one. In this work, image quality is defined in terms of image noise and contrast.

A key concept here is that the processor operates at the minimum power only when it is performing “just enough” computation to correctly perform edge detection and centroid location irrespective of the transmitted image quality and wireless channel condition. To throttle the processor around this “just enough” performance level, real-time feedback control is necessary. The proposed framework comprises of one feed forward and one feedback control loop (see Figure 82).



**Figure 82: Application driven power savings methodology.**

In the considered edge detection and image centroid applications, images are transmitted along with an image quality metric (IQM). Image contrast is used as the IQM and defines the quality of an image. When IQM is high, baseband processor performance can be degraded more than for an image corresponding to low IQM. We assume that the IQM value is computed at the camera source itself and transmitted to the receiver along the image data packets. This is used in a feedforward control configuration to control the processor performance level as shown in Figure

82. While IQM captures image quality, EVM of the received constellation is used in an inner control loop to jointly control the baseband processor performance (supply voltage, wordlength precision) to conserve power. Note that EVM is a function of the wireless channel baseband signal processing quality and is used as feedback metric (Figure 82).

### 6.4.1 Feed Forward - Image Quality Metric

As mentioned in the previous section, IQM is used as a feed forward control for setting up the EVM threshold value in the receiver. IQM is calculated in the transmitter and is sent in the first packet of an image. For computational ease, we use a simple image-contrast based metric, *the standard deviation of image pixel intensities*. The standard deviation of an image is calculated as shown below:

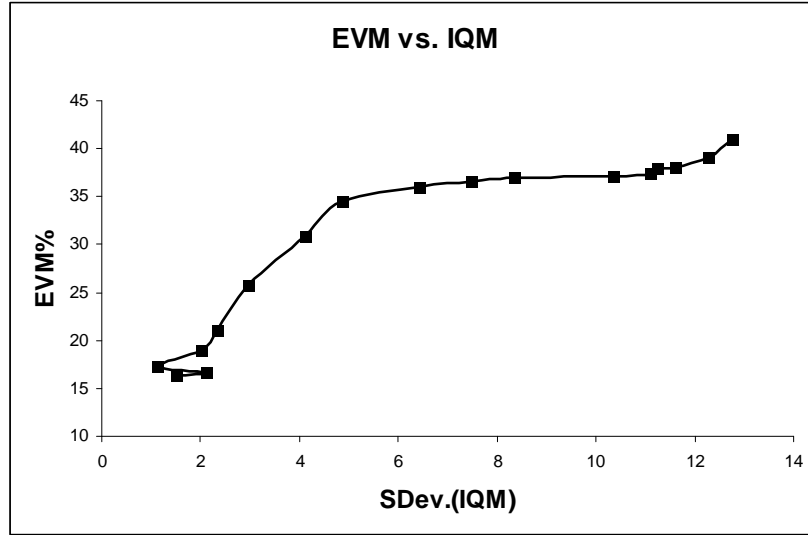
$$\text{Image standard deviation} = \sqrt{\frac{1}{M * N} \sum_{i=1}^M \sum_{j=1}^N (x(i, j) - \mu)^2}$$

$$\text{where } \mu = \frac{1}{M * N} \left( \sum_{i=1}^M \sum_{j=1}^N x(i, j) \right)$$

M= No of rows in the image.

N = No of columns in the image.

For EVM versus image quality relationship, a set of sample images are transmitted through a channel with varying noise. EVM is measured at the output of noisy channel and edge detection algorithm is applied on the images. EVM threshold for an image with a certain IQM is taken as the maximum value after which edge detection algorithm fails to detect edges. Measured EVM vs. IQM relationship is plotted in Figure 83. It is evident from the plot that, for an image with higher IQM (quality) system performance can be degraded more (higher EVM) and vice versa.



**Figure 83: IQM vs. system EVM.**

### 6.4.2 Feedback - Channel and System Performance Metric

EVM as explained in earlier sections measures the combined effect of signal transmission quality, channel conditions and receiver processing quality. Therefore it is an ideal metric in the system feedback loop for receiver performance adjustments.

### 6.4.3 Locus based Operation - Design and Characterization Phase

The locus based low-power operation of a baseband receiver for varying channel conditions and process variation adaptation is explained in Section 6.3.1. In the locus based technique, multiple loci are stored in a system at the end of the design phase. During the characterization phase, path timing tests determine the process variations in a device, and the corresponding operating locus of the device is selected for the optimal low-power run-time operation. An important characteristic of the locus based technique is that the EVM value is almost the same on all the operating points of the locus. In other words, end system performance remains constant irrespective of the channel conditions and only the system power consumption varies at different points of operation on the locus. Moreover, EVM values of all the stored loci are same and are

selected in such a way that BER does not increase in the system under the changing channel conditions. To accommodate the application driven performance requirements, locus based system design and characterization phase (as explained in Section 6.3.1.6) is modified as given below.

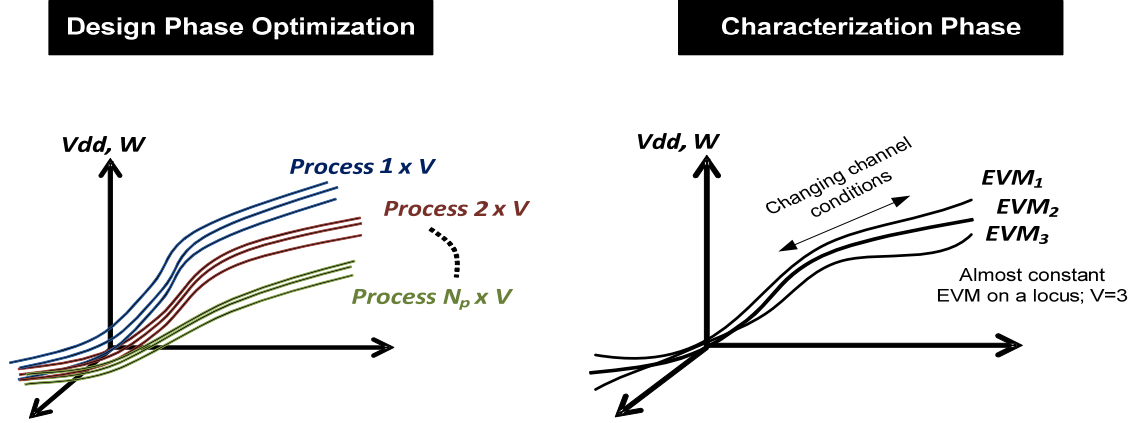
**Application Driven Design Phase:**

1. Repeat steps 1 to 7 (as given in Section 6.3.1.6) for a range of EVM values. EVM values are selected such that the higher EVM values induce higher BER in the system. At the same time, higher EVM values allow more power savings at the expense of system performance degradation.
2. At the end of the design phase, there will be  $N_p \times V$  number of loci to be stored in the system. Where  $N_p$  is the number of process variation combinations and  $V$  is the number of considered EVM values. In other words, for every process variation combination, there will be a  $V$  number of loci.

**Application Driven Characterization Phase:**

3. Run one time post-silicon POTTs on the device to determine process variation effects and pick the correct set of  $V$  loci based on the *process vector*.
4. System operates on one of the locus from the set of  $V$  loci depending upon the performance requirements (i.e. acceptable EVM value) of the end application.





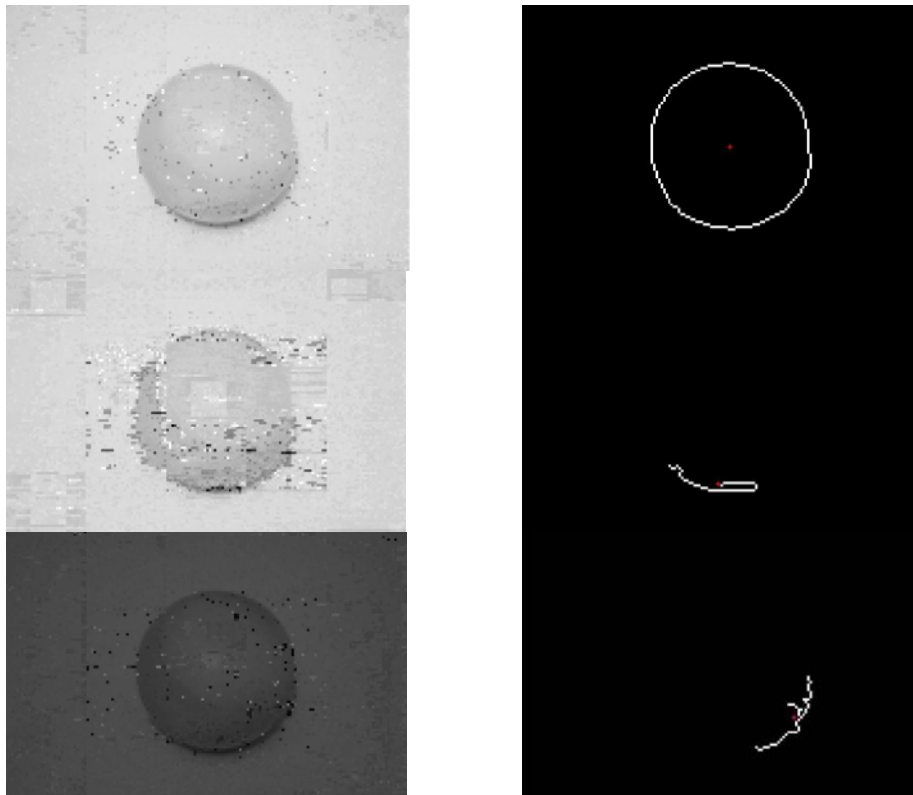
**Figure 84:**  $N_p \times V$  number of loci are stored at design time. System selects from a  $V$  number of loci at run-time depending upon the end-application performance requirements.

## 6.5 Evaluation

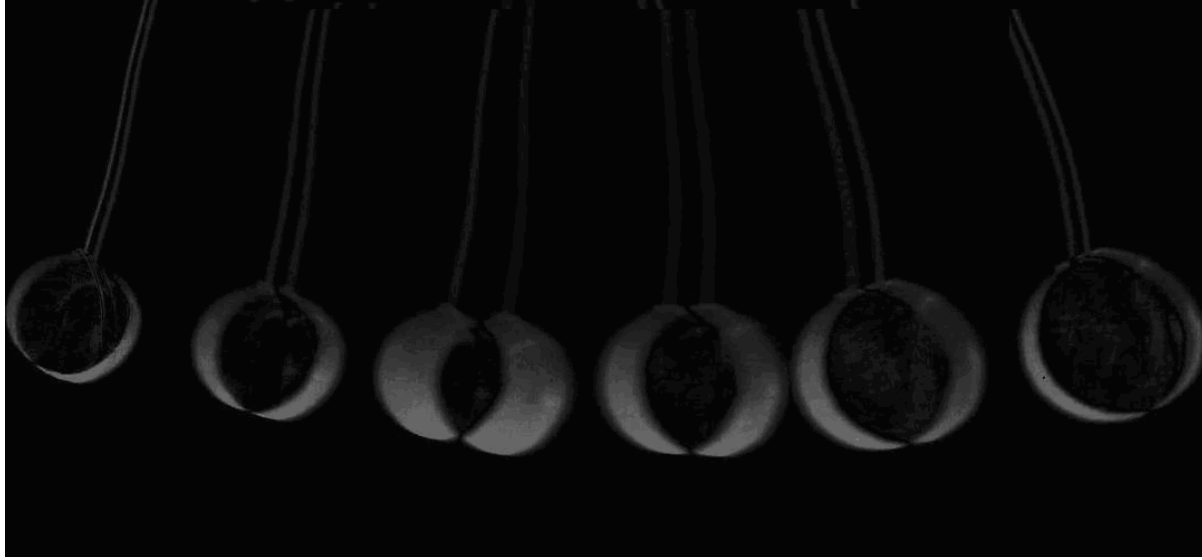
To validate the proposed power optimization technique, OFDM transceiver implementation as discussed in Chapter 3 is used. Implementation is done using Simulink Altera DSP builder and Megacore IP libraries. Images are QPSK modulated and data is transmitted in packets of 480 bytes. IQM information is appended in the first packet of the transmitted image. A 20-tap low-pass FIR filter and 128-point IFFT/FFT structures are implemented. Transceiver modules are implemented in FPGA with 14-bit precision. Data scaling is performed using shift registers at module inputs, effectively reducing the wordlength precision of the modules. EVM is calculated for every received packet and is used to determine the amount of data scaling (receiver performance degradation) for next packet. As long as calculated EVM is lesser than the threshold defined by IQM, system performance is degraded at run time to conserve power and vice versa. Canny edge detection [92] which is a Gaussian type edge detector is used in the experiments because of its better edge localization, improved signal to noise ratio and better performance in noisy conditions. In Figure 85, images of different quality at the receiver output for a given channel condition are shown. IQM values of images X and Y are 11.75 and 4.12 respectively. 7 bits were dropped in the receiver for images X and Y. Edge/centroid detection algorithm worked

successfully on good quality image X but failed to detect edges in image Y. It is evident from the result that receiver performance can be degraded more for a good image X as compared to a lower quality image Y, resulting in higher power savings for image X compared to image Y. Only 5 bits could be dropped for image Y for edge detection algorithm to work. Edge detection algorithm breaks down for image X at 8 bit drop.

The proposed methodology can easily be extended to other applications such as video transmission. With an end objective of object tracking the technique is applied to a video stream of a swinging ball. Figure 86 shows successive (overlapped) frames of a video where the swinging ball is tracked successfully after the video had been degraded in the baseband receiver. Power savings of more than 30% were observed by performance degrading the baseband processor for videos of high quality while meeting the performance requirements of the end application.



**Figure 85: (Top to bottom) Image X with 7-bit drop, Image X with 8-bit drop, Image Y with 7-bit drop.**



**Figure 86: Object tracking in a swinging ball video.**

## **6.6 Concluding Remarks**

In this chapter, various techniques are presented for channel driven adaptation of the OFDM baseband processor for low-power operation. The presented techniques are inherently process tolerant and results in better performance efficiency than if process variation is not taken into account. The techniques can be categorized into locus based operation and dual nested loop operation. In the locus based operation, design time system analysis results in operating parameters (voltage, wordlength precision, etc.) for changing channel conditions, and these parameters are stored in the system as operating locus. Incorporation of process variations results in multiple loci to be stored in the system. Before run-time operation of the device, one time device characterization selects the best operating locus for the device, which is then used for setting the system control parameters at run-time and results in optimal performance efficiency. Locus based operation is extended to application level by calculating different loci for varying performance levels. During run-time operation, the system selects one of the loci corresponding

to the performance requirements of the end application. It is shown with experimental results that locus based operation results in significant power savings under changing channel conditions.

The application driven technique results in higher power savings than simple locus based operation by changing the system performance depending upon the requirements of the end application. System operation parameters, i.e., loci are required to be stored in the system at design time and a one-time test is required before run-time for the device characterization (process variations estimation). In the dual nested loop architecture, no loci are required to be calculated and stored, and no test is required for process variation estimation. The two loops work together in a nested fashion to modulate the system tuning knobs for optimal performance efficiency. To avoid system instability due to nested loop oscillations, the threshold values in the loops are set with a higher margin, which can limit the power optimization of the nested architecture.

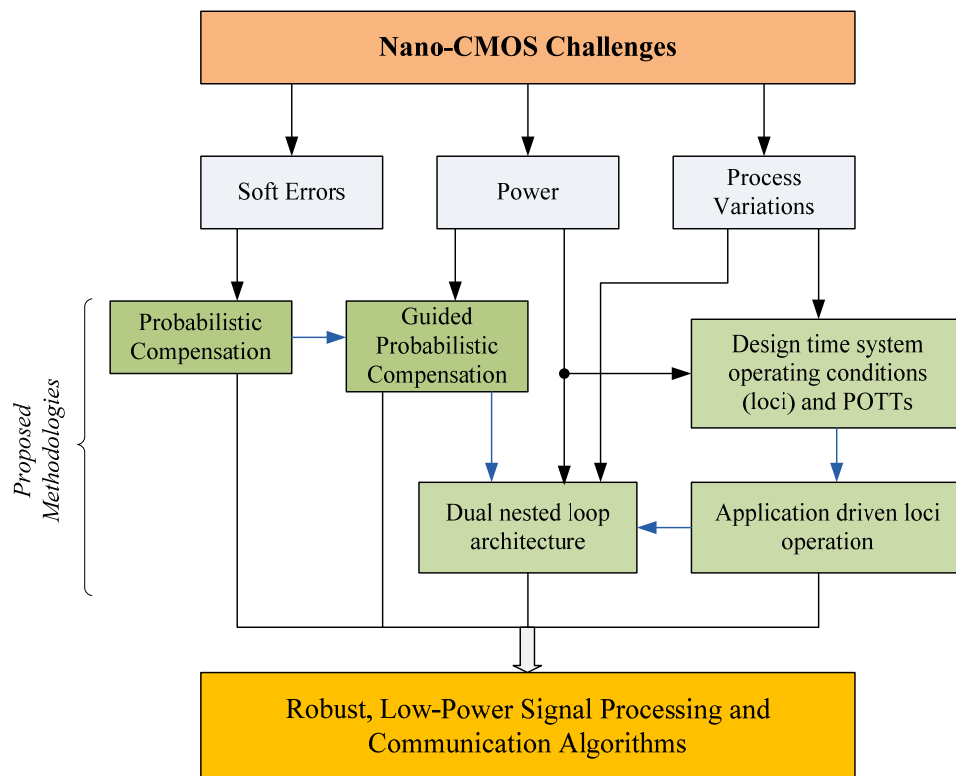
## CHAPTER 7

### CONCLUSIONS AND FUTURE WORK

Low-power, soft error mitigation and process tolerant techniques are presented in this thesis. The proposed techniques are applied on a wireless system but are equally applicable to other systems as well. Soft error mitigation technique allows considerable output signal quality improvement with minimal overhead cost. Error detection techniques are combined with low-cost error diagnosis to perform guided probabilistic compensation, which allows low-power operation of the DSP systems. Techniques are proposed for graceful system performance degradation and are integrated with low-power adaptation framework for wireless transceivers. The proposed framework employs a multidimensional control algorithm to tune the different modules of the baseband receiver for low-power, under changing channel conditions and process variation effects, while satisfying the performance requirements of the system. The wordlength precision and supply voltage adjustment are used as the tuning knobs in the control algorithm. A technique is also presented for tuning the receiver modules for low-power operation under changing channels and process variation effects without any design time calibration and without any adverse effect on the system performance. Another method enables application driven wireless receiver performance adaptation according to the performance requirements of the end application. The challenges addressed in this thesis and the proposed methodologies are summarized in Figure 87.

In the future work, more tuning knobs such as the analog-to-digital converter sampling rate can be incorporated in the low-power adaptation framework. The work in this thesis primarily focuses on the DSP components of the baseband receiver. A comprehensive framework that includes the forward error correction block and RF block of the receiver as well as the complete

transmitter will significantly enhance the potential of the proposed techniques and will result in significantly higher power savings. Receiver performance adaptation based on the channel and image quality metric provides a good example of system adaptation based on the end application requirements. However, development of more generic application performance metrics are needed that holds on many hardware platforms and have a stronger correlation with requirements of the system user. There exists tremendous opportunities to replace design time low-power adaptation framework with a self aware framework, which does not require design time calibration, an attempt at that is the presented dual nested loop architecture. Error detection and compensation methods using linearized checksum codes can be applied on the other DSP modules such as IFFT and FFT to provide soft error mitigation and low-power advantages with limited overhead.



**Figure 87: Proposed soft error mitigation, low-power and process tolerant techniques for robust, low-power signal processing and communication algorithms.**

## REFERENCES

- [1] G. E. Moore, "Cramming more components onto integrated circuits", Electronics, 1965.
- [2] Martin L. Hammond, "Moore's Law: The First 70 Years", Semiconductor International, 4/1/2004, <http://www.semiconductor.net/article/CA405656.html#fig>, 12/2009.
- [3] S. Borkar, "Design challenges of technology scaling", IEEE Micro, vol. 19, 1999, pp.23-29.
- [4] Stefan Rasu, Simon Tam, Harry Muljono, et al., "A 65-nm Dual-core Multithreaded Xeon® Processor with 16-MB L3 Cache", IEEE Journal of Solid-State Circuits, Vol. 42, No. 1, January 2007, pp. 17-25.
- [5] Peter Kogge, Karen Bargmen, Shekhar Borkar, et al., "ExaScale Computing Study: Technology Challenges in Achieving ExaScale Systems", Report sponsored by DARPA IPTO in the ExaScale computing study with Dr. William Harrod as Program Manager; AFRL contract number FA8650-07-C-7724, September 28, 2008.
- [6] ED L. Peterson, "Single - Event analysis and Prediction", IEEE Nuclear and Space Radiation Effects Conference, Short Course Text, 1997.
- [7] T. karnik, B. Bloechel, K. Soumyanath, V. De, and S. Borkar, "Scaling trends of cosmic ray induced soft errors in static latches beyond 0.18 $\mu$ ", 2001 Symposium on VLSI circuits. Digest of Technical Papers, 14-16 June 2001, Kyoto, Japan, pp.61-2.
- [8] S. Mitra, et. al., "Robust System Design with Built-In Soft error Resilience", Computer, vol. 38, no. 2, Feb. 2005, pp. 43-52.
- [9] D. F. Heidel, K. P. Rodbell, E. H. Canon, etc., "Alpha-particle Induced Upsets in Advanced CMOS Circuits and Technology", IBM Journal of Research and Development, Volume 52, Issue 3, May 2008.

- [10] J. Benedetto, P. Eaton, K. Avery, D. Mavis, etc., "Heavy-Ion Induced Digital Single-Event Transients in Deep Submicron Processes", IEEE Transactions on Nuclear Science, Vol. 51, No. 6, December 2004.
- [11] R. Baumann, "Soft errors in advanced computer systems," IEEE Design and Test of Computers, 2005.
- [12] P. Shivakumar, et.al., "Modeling the effect of technology trends on the soft error rate of combinational logic," Proc. Int. Conf. on Dependable System and Networks, 2002, pp. 389-398.
- [13] M. Nicolaidis, "Time redundancy based soft error tolerance to rescue nanometer technologies," Proceeding of VLSI Test Symposium, 1999, pp. 86-94.
- [14] K. H. Huang and J. A. Abraham, "Algorithm-based fault tolerance for matrix operations," IEEE Transactions on Computers," Vol. C-33, pp. 518-528, June 1984.
- [15] J. Jou and J. A. Abraham, "Fault-tolerant Matrix Arithmetic and Signal Processing on Highly Concurrent Computing Structures", Proceedings of the IEEE, vol. 74, No. 5, May 1986.
- [16] J. Y. Jou and J. A. Abraham, "Fault Tolerant FFT Networks," IEEE Transactions on Computers, Vol. 37, pp. 548-561, May 1988.
- [17] L. N. Reddy and P. Banerjee, "Algorithm-based fault detection for signal processing applications," IEEE Transactions on Computers, Vol. 39, No. 10, pp. 1304-1308, October 1990.
- [18] Y. S. Dhillon, A. U. Diril, A. Chatterjee, "Soft error tolerance analysis and optimization of nanometer circuits," Proceeding of Design Automation and Test in Europe, 2005, pp. 288-293.



- [19] U. Diril, Y. S. Dhillon, A. Chatterjee, A. D. Singh, "Design of adaptive nanometer digital systems for effective control of soft error tolerance," Proceeding of VLSI Test Symposium, 2005, pp. 98-303.
- [20] International technology roadmap for semiconductors (ITRS) update, 2008, <http://www.itrs.net/home.html>, 12/2009.
- [21] Y. Arima, et. al, "Cosmic-ray immune latch circuit for 90nm technology and beyond", Proc. SSCC 2004, pp. 492-493.
- [22] C. Hu, "Device and Technology impact on low power electronics", in Low Power Design Methodologies, J. M. Rabaey and M. Pedram, Eds.: Kluwer Academic Publishers, 1996, pp.21-35.
- [23] B. L. Austin, K. A. Bowman, X. Tang and J. D. Meindl, "A low power trans-regional MOSFET model for complete power-delay analysis of CMOS gigascale integration (GSI)", ASIC Conference 1998. Proceedings of the IEEE 2002, pp. 125-129.
- [24] R. K. Krishnarnurthy, A. Alvandpour, V. De, and S. Borkar, "High-performance and low-power challenges for sub-70 nm microprocessor circuits", Custom Integrated Circuits Conference, 2002. Proceedings of the IEEE 2002, pp. 125-128.
- [25] Ken Choi, "Ultra-low power design techniques and methodologies in nanometer CMOS technology", Feb. 2008 presentation at Gatech, Chicago, IL.
- [26] Uddalak Bhattacharya, Yih Wang, et. al., "45nm SRAM Technology Development and Technology Lead Vehicle", Intel Technology Journal, Vol. 12, Issue 02, June 2008, pp. 111-120.
- [27] K. Mistry, C. Allen, et. al., "A 45nm Logic Technology with High-k+Metal Gate Transistors, Strained Silicon, 9 Cu Interconnect Layers, 193nm Dry Patterning, and 100% Pb-free Packaging", Electron Devices Meeting, Dec. 2007, pp. 247-250.

- [28] Linden, D. and Reddy, T.; Secondary Batteries - Introduction. In handbook of Batteries, ed. D. Linden and T. Reddy, pp. 22.3-22.24. New York, NY: McGraw-Hill, 3rd edition, 2002.
- [29] Massoud Pedram, Jan Rabaey, "Power aware design methodologies", Kluwer Academic Publishers, 2002.
- [30] J. T. Kao, A. P. Chandrakasan, "Dual-threshold voltage techniques for low-power digital circuits," IEEE Journal of Solid-State circuits, Vol. 35, July 2000, pp.1009-1018.
- [31] K. Seta, H. Hara, T. Kuroda, et al., "50% active-power saving without speed degradation using standby power reduction (SPR) circuit", IEEE International. Solid-State circuits Conf., Feb. 1995, pp.318-319.
- [32] A. Abdollahi et. al., "Leakage current reduction current in CMOS VLSI circuits by input vector control", IEEE transactions on VLSI systems, Vol. 12, Issue 2, Feb. 2004, pp. 140-154.
- [33] A. Abdollahi et. al., "Runtime mechanisms for leakage current reduction in CMOS VLSI circuits", Proc. International symposium on low power electronics and design, august 2002.
- [34] L. Benini, P. Seigal and G. De Micheli, "Saving power by synthesizing gated clocks for sequential circuits", IEEE Design and Test of Computers, pp.32-40, Dec.1994.
- [35] G. E. Tellez, Amir Farrahi, Majid Sarrafzadeh, "Activity-Driven Clock design for Low Power Circuits", International Conference on Computer-Aided Design, pp. 62-5, November 1995.Vol.15, No.6, pp.630-43, June 1996.
- [36] H. Choi and W.P. Burleson, "Search-based wordlength optimization in VLSI/DSP synthesis", VLSI Signal Processing Workshop VII, pp.198-207, Oct.1994.

- [37] A. Oppenheim and R. Schaffer, Discrete-time Signal Processing, Prentice Hall, 1998.
- [38] K. Han and B.L. Evans, "Wordlength optimization with complexity and distortion measure and its applications to broadband wireless demodulator design", Proc. IEEE ICASSP2004, vol.5,pp.37-40, May 2004.
- [39] A.G. Dempster and M. D. Macleod, "Variable statistical wordlength in digital filters", IEE Proc.-Vis. Image Signal Process., Vol. 143, No. 1, February 1996.
- [40] S. Yoshizawa and Y. Miyanaga "Tuneable wordlength Architecture for a Low Power Wireless OFDM demodulator", IEICE Trans. Fundamentals, Vol.E89-A, No.10, October 2006.
- [41] J. M Chnag and M. Pedram, "Energy minimization using multiple supply voltages", IEEE transactions on Very Large Scale Integration (VLSI) Systems, vol. 5, pp.436-443, 1997.
- [42] "Introducing Intel Xscale Microarchitecture", <http://www.intel.com>, 12/2009.
- [43] "AMD Power now technology", <http://www.amd.com>, 12/2009.
- [44] Jan M. Rabaey, Massoud Pedram, "Low Power Design Methodologies", Kluwer Academic Publishers,1995.
- [45] D. Ernst, S. Das, D. Blaauw, "RAZOR: Circuit-level Correction of Timing Errors for Low-Power Operation", IEEE Micro, Vol. 24, Issue 6, Nov-Dec 2004, pp.10-20.
- [46] Naresh Shanbhag, "Reliable and Energy-Efficient Digital Signal Processing", DAC 2002, June 10-14, pp. 830-835.
- [47] A. Shim and N. R. Shanbhag, "Energy-Efficient Soft Error-Tolerant Digital Signal Processing", IEEE Tran. On VLSI, vol. 14, No. 4, pp. 336-348, April 2006.

- [48] Shim and N. R. Shanbhag, "Reduced Precision Redundancy for Low-Power Digital Filtering", Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, 2001, pp. 148-152.
- [49] Lei Wang and N. Shanbhag, "Low-Power Filtering via Adaptive Error-Cancellation", IEEE Transactions on Signal Processing, Vol/ 51, No. 2, Feb. 2003, pp. 575-583.
- [50] J. Choi, B. Shim, A. C. Singer and N. Ik Cho," Low-power Filtering via Minimum Power Soft Error Cancellation", IEEE Transactions on Signal Processing, Vol. 55, No. 10, Oct. 2007, pp. 5084-5096.
- [51] Swaroop Ghosh and K. Roy, "A New Paradigm for Low-power, Variation-Tolerant Circuit Synthesis Using Critical Path Isolation", ICCAD'06, Nov. 5-9, 2006
- [52] S. Nassif, "Within-Chip Variability Analysis", Proc. IEDM 1998, pp. 283-286.
- [53] S. Nassif, "Design for Variability in DSM Technologies", Proc. ISQED 2000, pp. 451-455.
- [54] B. E. Stine, et al, "Analysis and Decomposition of Spatial Variation in Integrated Circuit Processes and Devices", IEEE Trans. on Semiconductor Manufacturing, Vol. 10, No. 1, Feb. 1997.
- [55] Kelin Kuhn, Chris Kenyon, et. al., "Managing Process Variation in Intel's 45nm CMOS Technology", Intel Technology Journal, Volume 12, Issue 02, June 2008, pp. 94-109.
- [56] A. Chandrakasan, et al., Design of High-Performance Microprocessor Circuits, Wiley-IEEE Press 2000.
- [57] S. Borkar, et al., "Parameter Variations and Impact on Circuits and Micro-architecture", Proc. DAC 2003, pp. 338-342.
- [58] Pat Gelsinger, Senior Vice President and CTO, Intel Corporation, Key note speech at 41<sup>st</sup> DAC, June, 2004.

- [59] J. W. Tschanze, et. al, "Dynamic Sleep Transistor and Body Bias for Active Leakage Power Control of Microprocessors", IEEE Journal of Solid-State Circuits, Vol. 38, No. 11, Nov. 2003.
- [60] J. P. Uyemura, "Introduction to VLSI Circuits and Systems", John Wiley and Sons 2002.
- [61] O. Coudert, "Gate Sizing for Constrained Delay/Power/Area Optimization", IEEE Transactions on VLSI Systems, volume 5, Dec. 1997, pp. 465-472.
- [62] J.P. Fishburn, "LATTIS: An Iterative Speedup Heuristic for Mapped Logic", Proc. DAC 1992, pp. 488 – 491.
- [63] A. S. Chen and M. Sarafzadeh, "An Exact Algorithm for Low Power Library-Specific Gate Re-Sizing", Proc. DAC 1996, pp. 783 – 788.
- [64] P. Pant, et. al., "Dual-Threshold Voltage Assignment with Transistor Sizing for Low Power CMOS Circuits", IEEE Trans. on VLSI Systems, volume 9, April 2001, pp.390 - 394.
- [65] Y.S. Dhillon, et. al, "Algorithm for Achieving Minimum Energy Consumption in CMOS Circuits Using Multiple Supply and Threshold Voltages at the Module Level", Proc. ICCAD 2003, pp. 693-700.
- [66] S. H. Choi, et. al, "Novel Sizing Algorithm for Yield Improvement under Process Variation in Nanometer Technology", Proc. DATE 2004, pp. 454-459.
- [67] A. Srivastava, et. al, "Statistical Optimization of Leakage Power Considering Process Variations Using Dual-V<sub>th</sub> and Sizing", Proc. DAC 2004, pp. 773-778.
- [68] O. Neiroukh and X. Song, "Improving the Process-Variation Tolerance of Digital Circuits Using Gate Sizing and Statistical Techniques", Proc. DATE 2005, pp. 294 – 299.
- [69] M. R. Guthaus, et. al, "Gate Sizing Using Incremental Parameterized Statistical Timing Analysis", Proc. ICCAD 2005, pp. 1029 – 1036.

- [70] J. W. Tschanze, et. al, "Dynamic Sleep Transistor and Body Bias for Active Leakage Power Control of Microprocessors", IEEE Journal of Solid-State Circuits, Vol. 38, No. 11, Nov. 2003.
- [71] J. Tschanz, et. al, "Adaptive Body Bias for Reducing Impacts of Die-to-Die and Within-Die Parameter Variations on Microprocessor Frequency and Leakage", IEEE J. of Solid-State Circuits, Vol. 37, No. 11, Nov. 2002.
- [72] A. Keshavarzi, et. al, "Technology Scaling of Optimum Reverse Body Bias for Standby Leakage Power Reduction in CMOS IC's", Proc. ISLPED 1999, pp. 252-254.
- [73] T. Chen and S. Naffziger, "Comparison of Adaptive Body Bias (ABB) and Adaptive Supply Voltage (ASV) for Improving Delay and Leakage under the Presence of Process Variation", IEEE Trans. on VLSI, Volume 11, Issue 5, Oct. 2003 Page(s):888 – 899.
- [74] C. H. Kim, et. al, "A Process Variation Compensating Technique for Sub-90-nm Dynamic Circuits", Symp. on VLSI Circuits 2003, pp. 205-206.
- [75] C. H. Kim, et. al., "Self-Calibrating Circuit Design for Variation Tolerant VLSI Systems", Proc. IOLTS 2005, pp. 100-105.
- [76] Debaillie, B. Bougard, B., Lenoir, G., Vandersteen G., Catthoor, F., "Energy-scalable OFDM transmitter design and control", 43rd IEEE Design Automation Conference, July 24-28, pp. 536-541.
- [77] Tasic, A., Serdjin, W.A., Long, J.R., "Adaptive multi-standard circuits and systems for wireless communications", IEEE Circuits and Systems Magazine, Vol 6, Issue 1, pp. 29-37.
- [78] Abidi, A., Pottie, G.J., Kaiser, W.J., "Power-conscious design of wireless circuits and systems", Proceedings of the IEEE, Vol. 88, Issue 10, Oct 2000, pp. 1528-1545.

- [79] Arabi k., kaminska B., “Dynamic Digital Integrated Circuit Testing using oscillation-test method”, *Electronic Letters*, Volume 34, Issue 8, 16 Apr. 1998, Pages 762-764.
- [80] J.G. Proakis, *Digital Communications*, 2nd Edition, McGraw-Hill, 1989.
- [81] H. Wold and M. Despain, “Pipeline and Parallel-Pipeline FFT Processors for VLSI Implementations”, *IEEE Transactions on Computers*, Vol. C-33, No. 5, May 1984, pp.414-426.
- [82] P. Denyer and D. Renshaw, “VLSI Signal Processing: A Bit-Serial Approach,” Addison-Wesley, 1985.
- [83] K. Fazel, S. Kaiser, “Multi-Carrier and Spread Spectrum Systems”, John Wiley and Sons Ltd., 2003.
- [84] Bruno Bougard, “Cross layer energy management in broadband wireless transceivers”, PhD Thesis, March 2006.
- [85] Zhao, Y., Agee, B.G., Reed, J.H., “Simulation and measurement of microwave oven leakage for 802.11 WLAN interference management”, *Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*, vol 2, Aug 8-12, 2005, pp.1580-1583.
- [86] R. Burch, F. Najm, P. Yang, and T. Trick, “McPOWER: A Monte Carlo approach to power estimation,” *IEEE/ACM Int. Conf. on Computer-Aided Design*, Santa Clara, CA, Nov. 8-12, 1992, pp. 90-97.
- [87] Tasic, A., Serdjin, W.A., Long, J.R., “Adaptive multi-standard circuits and systems for wireless communications”, *IEEE Circuits and Systems Magazine*, Vol 6, Issue 1, pp. 29-37.
- [88] Abidi, A., Pottie, G.J., Kaiser, W.J., “Power-conscious design of wireless circuits and systems”, *Proceedings of the IEEE*, Vol. 88, Issue 10, Oct 2000, pp. 1528-1545.

- [89] J.G. Proakis, Digital Communications, 2nd Edition, McGraw-Hill, 1989.
- [90] R. Senguttuvan, S. Sen, A. Chatterjee, "VIZOR: Virtually zero margin adaptive RF for ultra low-power wireless communication", IEEE ICCD, Lake Tahoe, USA, 2007.
- [91] J.G. Ziegler, N.B. Nichols, Optimum settings for automatic controllers, Trans. ASME 64 (1942) 759–768.
- [92] J. Canny, "A computational approach to edge detection", IEEE Transactions on Pattern Analysis and Machine Intelligence Volume 8, Issue 6(Nov 1986) Pages: 679 - 698.
- [93] BPTM 70nm: Berkley predictive technology model.
- [94] Abhijit Chatterjee and Rabrinda k. Roy, "Concurrent Error Detection in Nonlinear Digital Circuits Using Time-Freeze linearization", IEEE Transactions on Computers, Volume 46, Issue 11, Nov. 1997, pp. 1208-1218.
- [95] M. Ashouei, et. al., "Design of Soft Error Resilient Linear Digital Filters Using Checksum-Based Probabilistic Error Correction', VLSI Test Symposium, Berkeley, CA, 2006, pp. 208-213.
- [96] M. Ashouei, et. al., "Improving SNR for DSM Linear Systems Using Probabilistic Error Correction and State Restoration: A Comparative Study", European Test Symposium, Southampton, UK, 2006, pp. 35-42.
- [97] V. S. Nair and J. A. Abraham, "Real-number codes for fault-tolerant matrix operations on processor arrays," IEEE Trans. on Computer, vol.39, pp. 426-435, April 1990.
- [98] A. Chatterjee, M. A. d'Abreu, "The design of fault tolerant linear digital state variable system: theory and technique," IEEE Trans. on Computers, vol. 42, pp. 794-808, July 1993.
- [99] Muhammad M. Nisar, M. Ashouei, "Probabilistic Concurrent Error Compensation in Nonlinear Digital Filters Using Linearized Checksums", IOLTS 2007, pp.173-182.



- [100] BPTM 65nm: Berkeley predictive technology model, “<http://www.eas.asu.edu/~ptm/>”, 12/2009.
- [101] M. Ashouei, M. Nisar, et. al., “Probabilistic Self-Adaptation of Nanoscale CMOS Circuits: Yield maximization under Increased Intra-Die Variations”, VLSI Design 2007, Pages 711-716.