

Cross-Layer Optimization for Power-Efficient and Robust Digital Circuits and Systems

Yanxiang Huang

Supervisors:
Prof. dr. ir. Wim Dehaene
Prof. dr. ir. Liesbet Van der Perre

Dissertation presented in partial
fulfillment of the requirements for the
degree of Doctor of Engineering
Science (PhD): Electrical Engineering

Cross-Layer Optimization for Power-Efficient and Robust Digital Circuits and Systems

Yanxiang HUANG

Examination committee:

Prof. dr. ir. Yves Willems, chair

Prof. dr. ir. Wim Dehaene, supervisor

Prof. dr. ir. Liesbet Van der Perre, supervisor

Prof. dr. ir. Francky Catthoor

Prof. dr. ir. Sofie Pollin

Prof. dr. ir. Rudy Lauwereins

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Electrical Engineering

Prof. dr. ir. Andreas P. Burg
(École Polytechnique Fédérale de Lausanne)

Prof. dr. ir. Liang Liu
(Lund University)



In collaboration with
imec vzw
Kapeldreef 75
B-3001 Leuven (Belgium)

September 12, 2017

© 2017 KU Leuven – Faculty of Engineering Science
Uitgegeven in eigen beheer, Yanxiang Huang, Kasteelpark Arenberg 10, B-3001 Leuven (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm, electronic or any other means without written permission from the publisher.

Acknowledgements

Undertaking this Ph.D. has been a truly life-changing experience for me. It would not have been possible without the support and guidance that I received from many people. It is a pleasure for me to take this opportunity to express my gratitude.

First of all, I would like to express my deep and sincere appreciation to my supervisors, Prof. Wim Dehaene and Prof. Liesbet Van der Perre. I have been incredibly lucky to have both as my advisors. Prof. Liesbet Van der Perre offered me the opportunity, backed in 2013 autumn, to pursue this research subject that I passionate. She always keeps faith in my work and provides full support in every aspect of my research, spanning from the knowledge on wireless communication systems, to scientific writing and even career advice. Prof. Wim Dehaene joined as my promoter in 2014 winter. He brings not only world-class understanding of circuits and systems, inspiring consultations on professional and personal developments. Furthermore, I would like to thank my committee member Prof. Francky Catthoor, for providing critical comments on my work, and for teaching me the enthusiastic attitude towards research. My cordial gratefulness also goes to all the jury members: Prof. Sofie Pollin, Prof. Rudy Lauwereins, Prof. Andreas P. Burg, and Prof. Liang Liu. It is their feedback that significantly improves this work.

My sincere thank goes to Andre Bourdoux, for his scientific supervision from the imec perceptive. I would like to thank Chunshu Li, towards whom I can share all the ideas freely. His advice helps shape a lot of the flashing thoughts towards research ideas. I would also express my appreciation to Meng Li. She has always been very supportive since the first day I joined imec.

Besides, I would like to express my gratitude my superb colleagues from the perceptive system group in imec, Maxim Rykunov, Khaled Khalaf, Claude Desset, Veerle Derudder, Qixian Shi, Hans Cappelle, Hans Suys, Simone Corbetta, Eddy De Greef, Miguel Glassee, and Karthick Parashar. Thank you

for your warm help when I came to you with mathematics formulation, Matlab code, system description, circuit simulation or even Latex code. Moreover, I would like to thank the group management and administration, Wim Van Thillo, Andy Dewilde, Charlotte Soens, Annemie Stas, and Bjorn Debaille, for their supportive assistance with the practical issues during the past years, especially scheduling tape-outs, conferences attendance, and measurement equipment.

Furthermore, I would like to thank various colleagues from imec, Peter Debacker, Min Li, Julien Verschueren, Sebastien Morrison, Paul Zuber, Petr Dobrovolny, Phillip Christie, Steven Redant, for the enjoyable cross-discipline collaborations. I am also very grateful for the colleagues from the KU Leuven, Chunming Liu, and Andras Boho for the inspiring discussions.

I am deeply grateful to my Ph.D. friends, Halil Kukner, Matthias Hartmann, Prashant Agrawal, Ubaid Ahmad, Steve Blandino, Alessandro Chiumento, Raf Appeltans, Oscar Elisio Mattia, Johan Nguyen, for the special 'Ph.D.' moments. My warm gratitude also goes to Zhijie Xu, Habib Khalid, Yuzhu Gai, Antonio Callejon, Sara Gunnarsson, and Miceale Bortas, for the splendid cooperation during their master thesis.

Moreover, I would like to share my gratitude towards the friends from imec and KU Leuven, Michael Chen, Xiaoqiang Zhang, Zhiwei Zong, Yao Liu, Cheng-Hsueh Tsai, Leqi Zhang, Linkun Wu, Mingxu Liu, Weiming Qiu, Tiwei Wei, Menglei Xu, Chang Chen, Ziyang Liu, Dongjie Shao, Gaozhan Cai, Xin Lin, Jiachen Wang, Xueji Zhang, many others and their family members, for invaluable help and consultant in all aspects.

Finally, I especially owe my loving thanks to my family. The trust and encouragement from my parents helped me to overcome every difficulty and will motivate me achieving any other success in the future. I devote my deepest gratitude to my wife, Dan Ning. We met, got acquaintance, and finally got married during my Ph.D. study in Leuven. This thesis would not have been possible without her warming hug, understanding, encouragement, and most importantly, love.

Yanxiang Huang

September 2017

Leuven, Belgium

Abstract

With the increasing digital services demand, performance and power-efficiency become vital requirements for digital circuits and systems. However, the enabling CMOS technology scaling has been facing significant challenges of device uncertainties, such as process, voltage, and temperature variations. To ensure system reliability, worst-case corner assumptions are usually made in each design level. However, the over-pessimistic worst-case margin leads to unnecessary power waste and performance loss as high as 2.2x. Since optimizations are traditionally confined to each specific level, those safe margins can hardly be properly exploited.

To tackle the challenge, it is therefore advised in this Ph.D. thesis to perform a cross-layer optimization for digital signal processing circuits and systems, to achieve a global balance of power consumption and output quality.

The first key contribution of this work is an algorithm to identify important flip-flops that contribute significantly to circuit outputs, by analytically modeling reliability threats at logic gate-level. Consequently, circuit designers are guided on which flip-flop to harden against random errors (on ASIC, e.g. FFT, LDPC decoder). With this help, this thesis eliminated the unnecessary hardening overheads (achieving the same SNR with only half of hardening overheads) for a traditional FFT design. This work also contributes by optimizing the micro-architecture for the embedded OpenRISC processor, exploiting application-level word-length opportunities. It improved the power efficiency for universal signal processing applications. For instance, It saved power by 9.5% for the whole execution unit on the Coremark benchmark. Moreover, this work mitigated circuit-level errors in micro-architectural and algorithm levels for recursive applications. The techniques were applied on a CORDIC hardware accelerator in digital front-end, which saves power consumption by up to 46%. The last key contribution of this thesis is the algorithm and application-level optimization to mitigate lower-level errors on Massive MIMO wireless communication systems. It demonstrated that for typical Massive MIMO setup, a few (3%) antenna

outage will only degrade the target system SNR by 0.5dB. This promotes operating the circuit in a more aggressive region, which might lead to circuit errors, to improve power efficiency.

To conclude, the traditional over-pessimistic worst-case approach leads to huge power waste. In contrast, the adaptive voltage scaling approach saves power (25% for the CORDIC application) by providing a just-needed supply voltage. The power saving is maximized (46% for CORDIC) when a more aggressive voltage over-scaling scheme is applied. These sparsely occurred circuit errors produced by aggressive voltage over-scaling are mitigated by higher level error resilient designs. For functions like FFT and CORDIC, smart error mitigation schemes were proposed to enhance reliability (soft-errors and timing-errors, respectively). Applications like Massive MIMO systems are robust against lower level errors, thanks to the intrinsically redundant antennas. This property makes it applicable to embrace digital hardware that trades quality for power savings.

Nederlandstalige Synopsis

Met de toenemende vraag naar digitale diensten worden prestaties en vermogen-efficiëntie energie-efficiëntie essentiële vereisten voor digitale circuits en systemen. De schaling van CMOS technologie die dit mogelijk maakt, wordt echter geconfronteerd met belangrijke uitdagingen door onzekerheden in geminiaturiseerde componenten, zoals proces-, spanning- en temperatuurvariaties. Om de betrouwbaarheid van systemen te waarborgen, worden meestal in alle ontwerp-niveaus conservatieve marges voorzien. Deze aannames zijn echter te pessimistisch, en de genomen marges zorgen voor energieverstopping en prestatieverlies dat kan oplopen tot een factor 2.2 van wat haalbaar is door de technologie. Aangezien de optimalisaties traditioneel beperkt zijn tot elk specifiek niveau, kunnen deze veilige marges niet goed worden benut.

Om deze uitdaging aan te pakken, wordt het daarom geadviseerd in dit Ph.D. proefschrift om een cross-layer optimalisatie uit te voeren voor digitale signaalverwerkingscircuits en systemen, om een globaal evenwicht te bereiken tussen stroomverbruik en kwaliteit aan de uitgangen van de schakelingen.

De eerste sleutelbijdrage van dit werk is een algoritme om belangrijke flip-flops te identificeren die aanzienlijk bijdragen aan de uitgangen van het circuit, door analytische modellering van betrouwbaarheidsdrempels op logisch poortniveau. Bijgevolg worden circuitontwerpers begeleid om enkel een hogere marge te nemen op flip-flops die gevoeliger zijn voor willekeurige fouten (bijvoorbeeld FFT, LDPC decoder circuits in een applicatie specifieke schakeling). Met deze hulp elimineerde dit proefschrift de onnodige marges op de helft van de flipflopseen traditioneel FFT-ontwerp, en kon een identieke kwaliteit bekomen worden als het conservatief ontwerp. Dit werk draagt ook bij aan het optimaliseren van de micro-architectuur van een microprocessor voor het verbeteren van de energie-efficiëntie voor universele signaalverwerking toepassingen . Dit verbeterde de energie-efficiëntie van de verwerkingseenheid van een OpenRisc processor met 9.5% voor de volledige Coremark benchmark . Bovendien verminderde dit werk fouten op circuitniveau in micro-architecturale en algoritme-niveaus voor recursieve

toepassingen. De technieken werden toegepast op een CORDIC hardware accelerator in digitale front-end, waardoor een besparing van energieverbruik met 46% wordt bekomen. De laatste belangrijke bijdrage van dit proefschrift is het optimaliseren van de algoritme en applicatie om de fouten van de Massive MIMO draadloze communicatiesystemen te beperken. Het blijkt dat voor een typische Massive MIMO-installatie antenne-uitval van 3% de signaal ruis verhoudingslechts met 0.5 dB zal degraderen. Dit laat toe de elektronische circuits te laten werken in een foutgevoeliger gebied, om de energie-efficiëntie van het totaal systeem gevoelig te verbeteren.

Tenslotte stelden we vast dat de traditionele keuze voor pessimistische marges inderdaad tot gevoelige vermogentoeename leiden. In tegenstelling hiermee bespaart de adaptieve spanningsschaalbenadering energie (25% voor de CORDIC-toepassing) door de benodigde voedingsspanning te beperken. De energiebesparing wordt gemaximaliseerd (46% voor CORDIC) wanneer een meer agressieve spanningsoverschaling wordt toegepast. Deze zelden voorkomende schakelfouten, die worden veroorzaakt door agressieve verlagen van de spanning, worden vermeden door een robuuster ontwerp op hogere niveaus van de meest gevoelige schakelingen. Voor functies zoals FFT en CORDIC werden slimme foutverminderingsschema's voorgesteld om de betrouwbaarheid te verbeteren (respectievelijk softfouten en timingfouten). Toepassingen zoals Massive MIMO-systemen zijn robuust tegen hoger fouten, dankzij de intrinsieke redundante antennes. Deze eigenschap laat toe om kwaliteit af te wegen ten opzichte van energiebesparing voor digitale hardware.

Abbreviations

This list of acronyms only specifies the most important ones. The abbreviations lists below represent both the singular and the plural forms.

ANT Algorithmic Noise Tolerance.

ASIC Application Specific Integrated Circuit.

ASIP Application-Specific Instruction set Processor.

AVFS Adaptive Voltage Frequency Scaling.

AWGN Additive White Gaussian Noise.

BER Bit Error Rate.

BS Base Station.

CMOS Complementary Metal Oxide Semiconductor.

CORDIC COordinate Rotation Digital Computer.

CPI Cycles Per Instruction.

CPU Central Processing Unit.

CS Computation-Skip.

DAC Digital-to-Analog Converter.

DFE Digital Front-End.

DL Down Link.

DSP Digital Signal Processing/Processor.

DVFS Dynamic Voltage Frequency Scaling.

ECC Error Correction Code.

EDA Electronic Design Automation.

ENoB Effective Number of Bits.

EVM Error Vector Magnitude.

FF Flip-Flop.

FFT Fast Fourier Transform.

FIR Finite Impulse Response.

FPGA Field-Programmable Gate Array.

IC Integrated Circuit/Chip.

IoT Internet of Things.

IR-drop dynamic voltage drop.

LDPC Low-Density Parity-Check code.

LSB Least Significant Bit.

LTE Long Term Evolution.

LUT Look-Up Table.

MAC Multiply Accumulate.

MCU Micro-Controller Unit.

MIMO Multiple Input Multiple Output.

MMSE Minimum Mean Square Error.

MSB Most Significant Bit.

MSFF Master-Slave Flip-Flop.

OCV On-Chip Variation.

OFDM Orthogonal Frequency Division Multiplexing.

- PA** Power Amplifier.
- PoFD** Point of First noticeable Degradation.
- PoFF** Point of First Failure.
- PoFW** Point of First Warning.
- PVT** Process Voltage Temperature.
- QAM** Quadrature Amplitude Modulation.
- QPSK** Quadrature Phase Shift Keying.
- RAM** Random Access Memory.
- RISC** Reduced Instruction Set Computer.
- RTL** Register Transfer Level.
- SDDR** Signal to Digital Distortion Ratio.
- SDR** Software Defined Radio.
- SEU** Single Event Upset.
- SIMD** Single Instruction, Multiple Data.
- SNR** Signal to Noise Ratio.
- SRAM** Static Random-Access Memory.
- STA** Static Timing Analysis.
- TDD** Time division duplex.
- UL** Up Link.
- VLSI** Very Large Scale Integration.
- VOS** Voltage-OverScaling.
- ZF** Zero-Forcing.

Contents

Acknowledgements	i
Abstract	iii
Nederlandstalige Synopsis	v
Contents	xiii
List of Figures	xix
List of Tables	xxiii
1 Introduction	1
1.1 Context: motivation for power consumption and reliability trade-offs	1
1.1.1 Demand: performance and power efficiency improvements	2
1.1.2 Reality: device and application uncertainties endanger successful designs	4
1.2 Calling for cross-layer optimizations	6
1.2.1 Traditional pyramid-shaped design	6
1.2.2 Cross-layer optimization	8
1.3 Thesis scope	10

1.4	Main contributions	11
1.5	Thesis structure	12
2	Background: pessimistic safety margins should be exploited	15
2.1	Uncertainties in circuit paramaters	15
2.1.1	Process variations and aging effects	16
2.1.2	Environmental and runtime uncertainties	19
2.2	Adjust supply voltage to trade quality for power savings	22
2.2.1	Limitations to the worst-case corner approach	23
2.2.2	Adaptive supply voltage schemes	24
2.3	Cross-layer error-resilient voltage over-scaling schemes	27
2.3.1	In-situ error detection and corrections	28
2.3.2	Arithmetic noise tolerance	29
2.4	Conclusion	30
3	Gate-level error impacts modeling analysis for random error mitigation	31
3.1	Demands for a reliability model	32
3.2	Modeling and enhancement techniques for reliability	33
3.2.1	Error modeling	33
3.2.2	Reliability enhancement techniques	34
3.3	Contributions of this chapter	35
3.4	SERIAL – a SignificancE RankIng ALgorithm for error effects modeling	36
3.4.1	Determine the distribution factor DF	39
3.4.2	Determine the significance S	44
3.5	Experimental verification	48
3.5.1	Scalability	48

3.5.2	Validation on an LDPC decoder	51
3.6	Application to harden an FFT design	52
3.7	Conclusion	53
4	Microarchitecture-level power optimization exploiting application opportunities	55
4.1	Processor power waste in using unsuitable execution units . . .	56
4.2	Contribution of this chapter	56
4.3	Targeted embedded processing platform	57
4.3.1	OpenRISC microarchitecture	57
4.3.2	Power of the multiplication unit	58
4.4	Fine-grained hardware-switch scheme for multiplier (<i>MUL</i>) . .	59
4.5	Algorithm profile	61
4.5.1	Utilization frequency of <i>MUL</i>	62
4.5.2	Word-length distribution for <i>MUL</i>	62
4.6	Power saving validation of the proposed fine-grained hardware switch scheme	63
4.6.1	Area comparison	64
4.6.2	Power savings	64
4.7	Conclusion	65
5	Algorithm-level error-resilient design mitigating circuit-level errors	67
5.1	Motivatuion to algorithm-level error resilience	68
5.1.1	Review of error-resilient techniques	68
5.2	Contributions of this chapter	69
5.3	Error resilient circuits implementations	71
5.3.1	Circuits with Canary FF	71
5.3.2	Circuits with <i>computation-skip</i> scheme	72

5.3.3	<i>Computation-skip</i> scheme settings	74
5.4	Case study on a CORDIC hardware accelerator	77
5.4.1	CORDIC algorithm	77
5.4.2	State-of-the-art <i>Core1</i> with Canary FF	79
5.4.3	Proposed <i>Core2</i> with <i>computation-skip</i> scheme	80
5.4.4	Pre-silicon analysis	83
5.4.5	Post-silicon Measurement	84
5.5	Conclusion	88
6	Application-level error-resilience on massive MIMO base stations	91
6.1	Opportunities and challenges of massive MIMO	92
6.1.1	Massive MIMO, the highly-demanded future technology	92
6.1.2	Power consumption, a design challenge in massive MIMO digital processing	93
6.2	Contribution of this chapter	94
6.3	Masive MIMO system introduction	94
6.4	Digital hardware error impacts on signal quality	97
6.4.1	VOS (Voltage-OverScaling) impacts	98
6.4.2	Antenna outage impacts	99
6.5	Random antenna error impacts assessment	99
6.5.1	Error effects on uncoded QPSK 100x10 massive MIMO	100
6.5.2	Antenna outage effects for other massive MIMO setups	103
6.6	Controlled antenna outage	106
6.7	Conclusion	110
7	Conclusions and future work	111
7.1	Conclusions	111
7.2	Key messages	114

7.3	Future work	116
A	A digital front-end processor for 60 GHz polar transmitter	119
A.1	Introduction	119
A.2	System architecture	120
A.2.1	Polar conversion	121
A.2.2	Phase shaping filter	122
A.2.3	Pre-distortion	123
A.3	Implementation details	124
A.4	Measurement results	127
A.5	Conclusions	130
	Curriculum Vitea	155
	List of publications	157

List of Figures

1.1	The world is immersed in digital services.	2
1.2	The free lunch of CMOS scaling is over.	4
1.3	Digital designs must cope with permanent (time-zero and time-dependent) and temporary (environmental and runtime) uncertainties (variations), to reach a balance between power consumption and output quality.	5
1.4	The traditional pyramid-shaped design flow leads to over-designs at all levels.	6
1.5	Over-designed guardbands diminish the benefits of CMOS scaling.	7
1.6	Promoted tower-shaped cross-layer design approach reduces unnecessary power and area waste.	9
1.7	Structure of the thesis and overview of the chapters.	13
2.1	Spatial process variations consist of global and local components.	17
2.2	The speed (inverse delay) of 28nm fan-out of 4 (FO4) ring oscillators (RO) follows a Gaussian distribution because of process variability.	18
2.3	The supply voltage of an IC is fluctuating.	19
2.4	The speed of a 28nm circuit changes along with temperature and supply voltage.	20
2.5	Worst-case corner, adaptive scaling, and error resilient voltage over-scaling approaches all cope with speed variability.	22

2.6	Spatial process variability are described by slow and fast corner-case files.	23
2.7	The speed of a 28nm circuit exhibits a 2.2x difference between the typical-case and worst-case corners.	24
2.8	The dynamical scaling method handles speed variability. Moreover, the error-resilient V_{dd} scheme also handles circuit timing-errors.	25
2.9	The operation region of a digital front-end IC can be much wider than conservatively assumed.	26
2.10	With VOS, errors firstly occur at MSB for carry-ripple adders.	28
2.11	VOS with error resilient techniques save power on FIR filter, at the cost of signal quality degradations [Liu, 2010].	30
3.1	Modeling process and structure for error impacts.	33
3.2	A significance graph inverses the signal direction of the corresponding netlist.	37
3.3	The SERIAL model consist of two procedures.	39
3.4	An example of computing the distribution factor DF (Procedure 1).	43
3.5	An example of computing the significance (Procedure 2).	47
3.6	In Procedure 2, the mismatch error ϵ_2 converges rapidly towards 0.	50
3.7	In an LDPC decoder circuit, errors on FF with high significance values leads to worse BER.	51
3.8	Compared with hardening randomly, selective hardening with SERIAL increases the SNR, at the same hardening ratio.	53
4.1	A customized OpenRISC microarchitecture: the mor1kx Cappuccino flavor.	57
4.2	The original multiplier in the Cappuccino implementation is 32 bits wide.	58
4.3	MUL with the proposed hardware-switching (HS) scheme selects between a 32-bit and an 8-bit multiplier, based on the data range.	60
4.4	HS saves power when the multiplication inputs are small.	61

4.5	Utilization frequency of the multiplier unit.	62
4.6	The chance that both multiplicands are short increases when the criteria for ‘short’ gets loose.	63
4.7	The proposed <i>HS</i> scheme results to more area.	64
4.8	The power consumption of the <i>HS</i> reduces substantially.	65
5.1	The conventional pipeline scheme requires worst-case design margins.	71
5.2	Circuits with in-situ <i>canary FF</i> that shave worst-case design margins.	72
5.3	Circuits with <i>computation-skip</i> scheme, which utilizes <i>DTSBs</i> for error-detection, eliminate worst-case design margins.	73
5.4	Timing constraints for circuits with the <i>computation-skip</i> scheme.	75
5.6	The CORDIC output ENOB evolves with numbers of iterations n	79
5.9	Example timing diagrams of the proposed <i>computation-skip</i> CORDIC.	82
5.10	450 MHz is selected for the operating condition.	83
5.11	Cell area breakdown of the conventional and the proposed hardware CORDIC accelerators designed at 450 MHz.	84
5.12	Die photo and chip information.	85
5.13	Measured timing violation ratio.	86
5.14	Measured CORDIC ENOB.	86
5.15	Comparison of the <i>core1 canary</i> (implementing the baseline state-of-the-art technique) and <i>core2 CS</i> (implementing the proposed scheme) CORDIC accelerators during voltage scaling.	88
6.1	Massive MIMO exploits large antenna arrays to spatially multiplex many terminals.	92
6.2	A massive MIMO base station (BS) equips with M antennas and serves K users.	96
6.3	Slight VOS digital distortion errors, i.e. SDDR = 10 dB, only degrades the system BER marginally.	101

6.4	If the massive MIMO system is suffering from extensive VOS digital errors on some antennas, i.e. $SDDR = 0$ dB, the output is still manageable.	102
6.5	The system can tolerate some antenna outage (DFE output stuck at a fixed value), i.e. $SDDR = -\infty$	103
6.6	Simple modulation schemes are more resilient to antenna outage.	104
6.7	Lower loads leave more spaces for error absorption.	105
6.8	Coding and uncoded degradations are similar.	106
6.9	Massive MIMO system performance when victim antennas are discarded and the channel estimation and DL is carried out by the remaining error-free antennas.	108
6.10	An error resilient adaptive scaling technique to manage hardware errors.	109
6.11	The frequency of changing of the process, aging, temperature, and voltage noise that affect the timing errors of circuits.	110
A.1	Block diagram of the digital intensive 60 GHz polar transmitter.	121
A.2	Poly-phase implementation of the phase shaping filter.	122
A.3	EVM vs. PH/AM resolution.	123
A.4	Pre-distortion circuits for PA non-linearities.	123
A.5	Pipeline scheme of the proposed DFE processor.	125
A.6	Die photo and chip information of the proposed digital polar DFE processor.	127
A.7	Cell area breakdown of the proposed DFE processor.	127
A.8	Typical chip speed vs. V_{dd} @25°C.	128
A.9	Measurement DFE outputs PSD.	129
A.10	Measurement DFE output 16-QAM constellation, with a corresponding EVM of -30.5 dB.	129

List of Tables

3.1	Truth table of a 3-input AND gate.	41
3.2	SERIAL model solver runtime analysis for benchmark circuits.	49
4.1	Multiplier area and power consumption during each instruction.	59
5.1	In-situ timing-error detection and correction/mitigation schemes.	70
5.2	Timing constraints for the <i>computation-skip</i> error mitigation scheme.	76
6.1	Complexity of digital components for a 100x10 massive MIMO system.	95
A.1	Performance comparisons of digital polar front-end systems. . .	130

Chapter 1

Introduction

The groundbreaking digital revolution has refined the work and life styles of every individual. However, the gap between great application requirements and the enabling CMOS technology limitations is prominent. Therefore, design techniques to balance the quality and the power consumption of integrated circuits are demanded in the nano-CMOS era. This work promotes cross-layer optimizations for power and quality trade-off. It accepts errors that traditional designers advocated avoiding at all cost. The CMOS devices are working at their extremes with fewer safety margins. This enables extra power saving without noticeable quality degradations. This chapter introduces this work.

The chapter is structured as follows: Section 1.1 describes the motivation of this thesis. It reviews the contradiction between the need and the reality of current digital circuits and systems. A cross-layer optimization approach is therefore promoted (Section 1.2). Section 1.3 summarizes the scope of this thesis. Section 1.4 lists the main contributions of this work. Finally, Section 1.5 presents the structure of this thesis.

1.1 Context: motivation for power consumption and reliability trade-offs

The ubiquitous digital infrastructure and services have totally changed our way of life, leisure and our means of communication and information. The new generation is enjoying the conveniences offered by the computers, phones, gadgets, and varieties of smart IoT devices that exceed the visions of last

generation's craziest sci-fi movies. The immersion of digital world is playing a key role in human lives (Fig. 1.1). According to [EMarketer, 2017], adults are spending more than 12 hours accessing digital data in 2017.

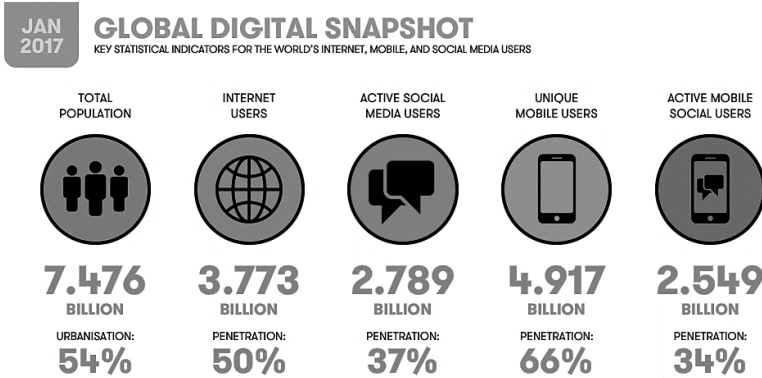


Figure 1.1: The world is immersed in digital services ([Kemp, 2017]).

The digital reality cannot be realized without the development of the enabling technologies. For the past few decades, we have seen marvelous breakthroughs in the domains such as personal computers, the internet, and mobile communications. Those breakthroughs, not only redefine the frontier of technologies but also expedite the mass adoption of them. For instance, almost three-quarters of the world's population now use a mobile phone, with the total number of unique global mobile users rapidly approaching 5 billion [Kemp, 2017]. The advance of technology is now spreading to the applications as artificial intelligence, smart vehicle, and IoT devices.

1.1.1 Demand: performance and power efficiency improvements

Consumers are demanding higher data volume, in a faster rate. For instance, the world monthly mobile data traffic goes steeply from 3.7 EB (Exabytes) in 2015, to 9.9 EB in 2017, and will increase to 30.6 EB by 2020 [Index, 2016]. The exponential growth of wireless data services driven by the mobile internet and smart devices has triggered the investigation of the 5G cellular network. The mobile phone of the future has to provide seamless connectivity anywhere and anytime. Around 2020, the new 5G mobile networks are expected to be deployed [Andrews, 2014]. These networks will support multimedia applications with a wide variety of requirements, including higher peak and user data rates (more

than 100 megabits per second for metropolitan areas [Osseiran, 2014]), reduced latency (less than 1 ms [Best, 2014]), enhanced indoor coverage, improved energy efficiency and so on.

Powering up those digital services requires a big amount of energy. Worldwide, data centers use about 400 terawatt-hours of electricity each year [Andrae, 2015]. That's a little more electricity than all of the United Kingdom uses. Already, they have mushroomed from virtually nothing 10 years ago to accounting for about 2 per cent of total greenhouse gas emissions [Bawden, 2016]. That gives it the same carbon footprint as the airline industry. If left unchecked, they could use almost 8,000 terawatt-hours by 2030. That's about the amount of electricity all of Europe and Africa and much of Asia use today. Another optimistic estimation predicts that the global IT services will consume 15% electricity production world-wide by 2025 [METI, 2008]. Considering the continuing demands for increased digital services, the energy efficiency for digital computation must be improved accordingly.

In the past few decades, the demands of increasing performance and power efficiency were realized with the CMOS technology progress. The complexity of integrated circuits has approximately doubled every 18 months; the cost per function has decreased several thousand-fold. The exponential growth has fit to the well-known Moore's Law (prediction) [Moore, 1998]. In the computer sector, the CPU had kept pace with the Dennard's scaling [Dennard, 1974], which suggests that the performance per watt also grows exponentially (Fig. 1.2). However, CPU performance, which is largely coupled with the clock speed, has stalled at around 2005, since the 65nm CMOS process. This suggests the end of the free lunch brought by the CMOS scaling [Sutter, 2005].

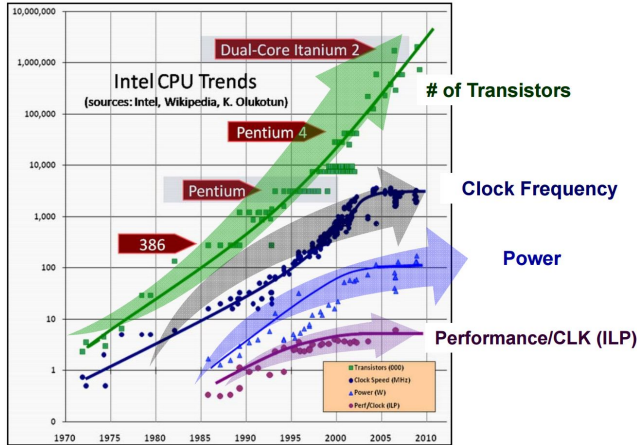


Figure 1.2: The free lunch of CMOS scaling is over [Sutter, 2005].

The power consumption (too high), heat (too much of it and too hard to dissipate), and current leakage problems are among the biggest challenges for the continuing of free-lunch scaling. Multicore parallelisms are employed to keep improving the performance, which eventually leads to new power challenges (dark silicon) that may end the multicore era [Esmailzadeh, 2011]. Nowadays, power consumption has become an, arguably the most, important metric for digital computing devices.

Apart from saving energy, another motivation to minimize power consumption is to fit digital chips into ubiquitous IoT devices, much of which are powered by batteries or energy harvesters [Bravos, 2005; Gyselinckx, 2005; Rawat, 2014]. Looking to the future, Cisco IBSG predicts there will be 25 billion devices connected to the Internet by 2015 and 50 billion by 2020 [Evans, 2011].

In summary, the increased IT service calls for, in addition to advanced CMOS technologies, successful enhancement in design of digital circuits and systems, to fulfill the performance and power efficiency requirements.

1.1.2 Reality: device and application uncertainties endanger successful designs

The variability has become a major roadblock to CMOS scaling (Fig. 1.3). Below the sub-65nm regime, transistors no longer act deterministically as a consequence of fluctuations in device parameters. This phenomenon is caused

by the process challenges (lithography, etching, chemical mechanical polishing, etc.) [Ghosh, 2010]. These process challenges not only alters chip parameters (speed, area, power, etc.), they also result in functionality failures, e.g. stuck-in fault. The time-zero manufacturing process is not the only source of device uncertainties, chips also face aging problems during lifetimes. Worse still, the aging phenomena are uncertain in themselves. They heavily depend on the environment or workloads [Mintarno, 2013].

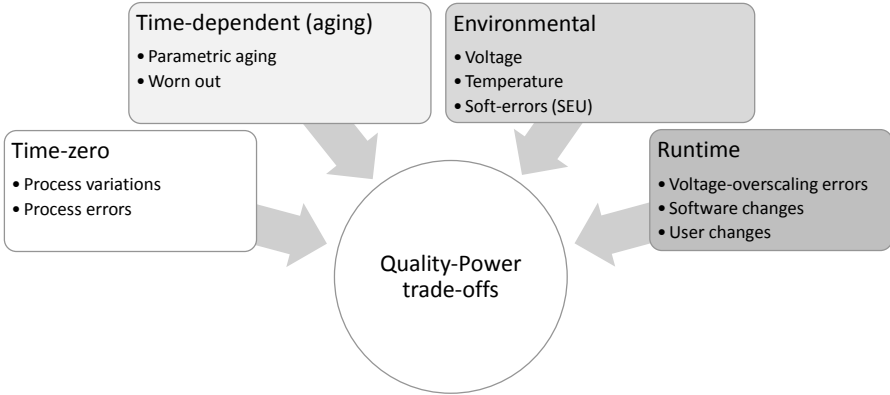


Figure 1.3: Digital designs must cope with permanent (time-zero and time-dependent) and temporary (environmental and runtime) uncertainties (variations), to reach a balance between power consumption and output quality.

Apart from these (semi-) permanent effects, environment [Unsal, 2006] (voltage, temperature, cosmic particle strikes, etc.) and runtime (timing errors, software, and user) variations also add uncertainties to chip designs. For instance, high energy particle strikes can lead to random bit flippings on storage elements. VOS (Voltage Over-Scaling) circuits [Hegde, 2004; Jeon, 2012] tries to save power by operates at a riskily-low supply voltage ¹, which produces uncertain errors. In summary, these uncertainties share a similarity that they only affect the IC temporaries.

The time-zero and time-dependent challenges are often regarded as a yield problem, which is mainly tackled by the IC foundries [Tsai, 2004]. In contrast, the fast changing runtime uncertainties are so dynamical that they cannot be simplified solved by traditional post-silicon testing [Nithin, 2010]. Therefore,

¹It is different from the sub-threshold computing technique [Dreslinski, 2010] where the V_{dd} and the clock frequency are very low. In contrast, the VOS reduces V_{dd} , but not the clock frequency. The operating voltage of VOS is much higher than the sub-threshold region.

runtime uncertainties management is a domain that is still wide open and much profitable for digital circuits and systems designers. The fact that these uncertainties only leads to temporary effects makes trading quality for power savings possible.

The source and effects of them are further explained in Section 2.1. To conclude, careful consideration of all those uncertainties is essential for successful digital circuit and system design.

1.2 Calling for cross-layer optimizations

1.2.1 Traditional pyramid-shaped design

Traditionally, the digital design flow follows a top-down procedure (Fig. 1.4). That is, the top-level design specs, e.g. quality, speed, power consumption, are assigned by system architects. The algorithm and circuit-level designers are forced to come up with solutions to fulfill these requirements. This rigid task dividing simplifies the design process, as fewer confusions during design are expected.

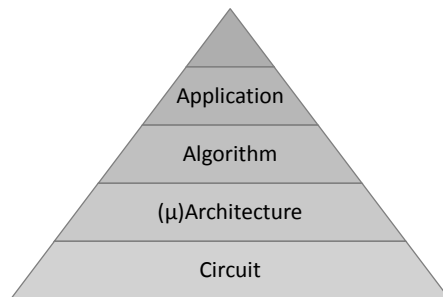


Figure 1.4: The traditional pyramid-shaped design flow leads to over-design at all levels (derived from [Hennessy, 2011]).

Following this design flow, the variations are traditionally tackled in particular levels. For instance, the process, voltage, and temperature (PVT) variations are packed up together and guaranteed with worst-corner safety margins at the circuit level (Section 2.2.1). Temporal degradation issues are hidden in the time-zero variations margins, albeit early works on designing with accurate time-dependent models [Liu, 2017] start to gain popularity.

A variety of safety margins are inserted in all levels of the design for manufacturing cost, reliable computing, acceptable device lifetime, device performance, etc [Austin, 2008]. If keeps inserting margins and being pessimistic, the gain obtained by the CMOS scaling will reduce and might eventually diminish (Fig. 1.5).

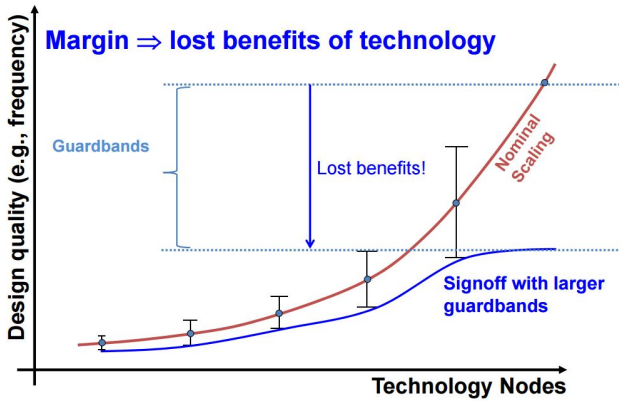


Figure 1.5: Over-designed guardbands diminish the benefits of CMOS scaling [Kahng, 2011].

The drawback is that, following the traditional pyramid-shaped design flow, smart engineering on quality and power consumption is usually limited to specific design levels, leading to only local optimal solutions. The application-level and the algorithm-level designs call for error-free results from lower levels, which makes lower level designs (e.g., at the architecture and circuit-level) unnecessarily complex sometimes. For example, chips should work at any temperature within the specification range, and thus transistors delays are pre-characterized on all temperatures. Circuits designers always try to avoid uncertainties and constrain chips to work equally on that (wide) delay range, which is not easy. Not many “if-then-else” cross-layer co-optimization are possible conventionally. The waste of resources of this worst-case guard-band is reviewed in Section 2.2.1.

Considering that the lower levels are utilizing too much resources to provide the higher level application service, the design flow is named as a pyramid-shaped method. In summary, the lack of information across design levels leads to wasted resources (in terms of power consumption and area cost). It is therefore

calling for a new design and optimization paradigm.

1.2.2 Cross-layer optimization

To exploit the resource waste in the pyramid-shaped design, this thesis promotes a cross-layer optimization design approach ². With this optimization paradigm, information are sharing across design layers for power savings or quality improvements. As will be demonstrated in the rest of this thesis, by extending optimization across different design levels, the over-design can be much reduced. This provides an advantage in conserving area and power consumption at the silicon foundation and increasing performance and quality at the application-level. This paradigm is named as a tower-shaped design flow (Fig. 1.6), reflecting the aim that each design layer is utilizing just-needed resources, which avoids over-designs.

²Corss-layers in this work means to optimize across different design levels, which is different from the notion of “layer” from the Open Systems Interconnection (OSI) model.

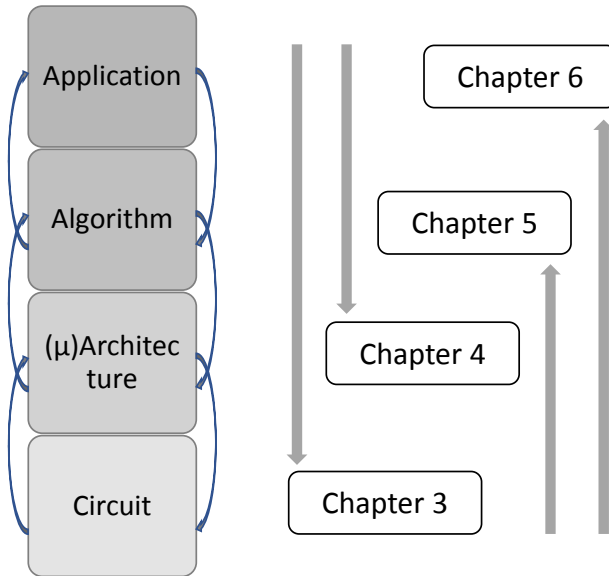


Figure 1.6: Promoted tower-shaped cross-layer design approach reduces unnecessary power and area waste. This is achieved by exchanging information across design levels. Comparing with the pyramid-shaped cartoon (Fig. 1.4), the new paradigm requires less space at the low level, yet provides more rooms at the top of the tower (hence a tower rather than a pyramid). Chapter 3 and 4 transfers information from higher levels to circuit and micro-architecture levels. Chapter 5 and 6 mitigates lower level errors at algorithm and application levels.

In contrast of the tower-shaped design, errors are often permitted in the context of cross-layer optimizations. [Djahromi, 2007] showed that by allowing data bearing memories to have a controlled number of errors, significant gains in power consumption are possible. In fact, for a 3GPP modem, power savings of up to 17.5% are achievable assuming a 32nm technology. The Razor [Ernst, 2003] and ANT [Hegde, 1999a] techniques are excellent cross-layer optimization examples. They are discussed in details in Section 2.3.

[Carter, 2010; Mitra, 2010; GimmlerDumont, 2013] overviewed of the design techniques for cross-layer resilience. They highlights that distributing resilience and reliability across the system stack can improve performance and reduce power and area costs by taking advantage of the strengths of each layer and exploiting the characteristics of individual applications.

Admittedly, the design effort of this new paradigm is larger because a cross-layer information should be shared and evaluated. Though, its benefits are worth the effort. This thesis demonstrates that the cross-layer approach brings considerable benefits for digital circuits and systems, with limited design effort increase. For example, lower-level designs assume (predicate) some properties of the higher-level, which reduce the power cost once predicated successfully; lower-level designs generate some errors under the worst situations, which will be handled by the higher-level designs.

1.3 Thesis scope

This thesis aims to reduce safety margins and saving power. Runtime adjusting approaches are employed to exploit those margins. More specifically, this work mainly handles the environmental and runtime uncertainties. The reason is that, these uncertainties are so dynamical that they cannot be simplified solved by traditional post-silicon testing [Tsai, 2004; Nithin, 2010]. It not only responses to the environmental changes as the DVFS techniques (Section 2.2.2), but also allows occasional errors to propagate through different levels (e.g. Section 5). Techniques in this work, however, also exploit margins for slowly changing process variations (time-zero and time-dependent), assuming that they are already packed in the safety margin. The permanent breakdowns are not covered in the thesis. This is because time-zero breakdowns are classified as yield problems and are mostly taken care of by semiconductor foundries; workload-dependent aging effects are considered by EDA tools recently [Karapetyan, 2015].

The targeted applications of this thesis are soft-quality requirement digital signal processing devices. These cross-layer optimizations are particularly beneficial in the context of soft-quality requirement systems. In these systems, errors are acceptable as long as the system output still meets the requirements. For instance, the transient error in a wireless communication application can easily be tolerated by the symbol detector, considering that the error vector magnitude is allowed [Karakonstantis, 2012b]. This distinguishes from the general-purpose computer case [Carter, 2010] where usually no errors are permitted. Another example is video processing applications, where skipping few erroneous pixels can be tolerated [Driscoll, 2003]. This leaves plenty of spaces for quality-power trade-offs.

Therefore, in this work, the uncertainties incurred errors are allowed to propagate to the algorithmic and application-level. The end result is a good balance between the quality and power consumption. In specific, Chapter 3 firstly models the algorithm-level impact of random errors, in the circuit-level, without

digging into the algorithm. It then provides valuable guidelines to selectively harden designs against random errors at the circuit-level. Chapter 4 predicates programs' behavior, and modifies the microarchitectural structure, which leads to power savings for typical usages. Chapter 5 mitigates circuit-level errors at the microarchitecture and eventually the algorithm-level, leading to a graceful quality degradation. Chapter 6 demonstrates that lower-level generated errors can be handled at the application-level. This provides opportunities to embrace hardware uncertainties for power saving. It takes the Massive MIMO wireless communication application as a case-study and demonstrates its resilience to lower-level errors. The chapter, therefore, encourages to use low-power yet erroneous components in the Massive MIMO.

1.4 Main contributions

This work has contributed to better power solutions of digital designs in scaled CMOS, through a cross-layer optimization. These approaches were demonstrated in the content of wireless communication applications. The main research contributions of this thesis are summarized below. A more elaborate version of the main messages is provided in Chapter 7.

- Introduction of an analytical circuit-level random error effects model (Chapter 3). This thesis proposed a graph travel approach that solves the model. It is shown how a graph based scheme can identify the sensitivity of individual Flip-Flops. This helps to selectively protect only those. It also demonstrated the scalability and effectiveness of the model on ISCAS and ITC benchmark circuits. Finally, this work validated the benefits of the model on an FFT processor design that reduces soft-error hardening overhead. This work was unveiled in [Huang, 2016b].
- Proposed a novel fine-grain hardware-switch scheme to save power in embedded processors (Chapter 4). The thesis applied the proposed scheme to the multiplier unit of an OpenRISC processor. This technique competes with the idea of VLIW based SIMD instructions that requires large compiler modification. It demonstrated power savings on 11 typical signal processing applications, e.g. FFT, IIR, AES, JPEG. This work was published in [Huang, 2016c].
- Application of Razor circuit-level error detection techniques with error mitigation achieved through an algorithmic approach. Proposed a novel computation-skip scheme to mitigate errors for recursive applications (Chapter 5). It saves power saving by reducing the supply voltage,

exploiting not only the error-free but also error resilient safety margins. This is achieved by skipping part of the computation and sacrificing some accuracy. The thesis implemented the scheme on a CORDIC hardware accelerator in 28nm CMOS technology with standard digital design flow. The work was published in [Huang, 2014], and elaborated in [Huang, 2016d].

- Investigation of application-level error absorption and handling for Massive MIMO wireless communication applications (Chapter 6). It demonstrated the error resiliency of Massive MIMO systems under hardware errors and even antenna outage. It also proposed a damage control strategy for Massive MIMO applications. The work is published in [Huang, 2017].

List of publications

The list of publications can be found in the attached Curriculum Vitae section.

1.5 Thesis structure

In this thesis, cross-layer optimization are performed for digital circuits and systems for power consumption and reliability trade-off (Fig. 1.7).

Chapter 2 reviews techniques for power and quality tradeoffs. It starts with device-level phenomena of variations. The chapter then reviews the worst-case design and the adaptive scaling method that tunes supply voltage to save power. Finally, the benefit of the cross-layer VOS (Voltage Over-Scaling) approach is justified.

As shown in Fig. 1.6, Chapter 3 develops a gate-level random error model, SERIAL. It models the importance of flip-flops regarding their impact on algorithm outputs. The efficiency and effectiveness of the model are shown in typical circuits, including ISCAS and ITC benchmarks, and an LDPC decoder. Finally, the model is applied to design a reliable FFT processor.

In Chapter 4, a microarchitecture-level fine-grain hardware-switch scheme for embedded processors power savings is proposed. In specific, the chapter modifies the multiplier unit of the OpenRISC platform. It demonstrates power savings on typical signal processing applications.

Chapter 5 proposes a method for cross-layer error interplay between the circuit-level and the algorithm-level. It presents a computation-skip scheme to mitigate

errors in recursive applications. The error mitigation scheme, together with the state-of-the-art timing error detection benchmark, are applied to a hardware CORDIC accelerator. The CORDIC accelerator is processed and verified in a standard 28nm CMOS process with only standard-cells.

Chapter 6 presents the application-level error absorption and handling. It focuses on a Massive MIMO communication case-study. This chapter assesses hardware random errors (VOS) and antenna outage impacts. Finally, damage control strategies are proposed.

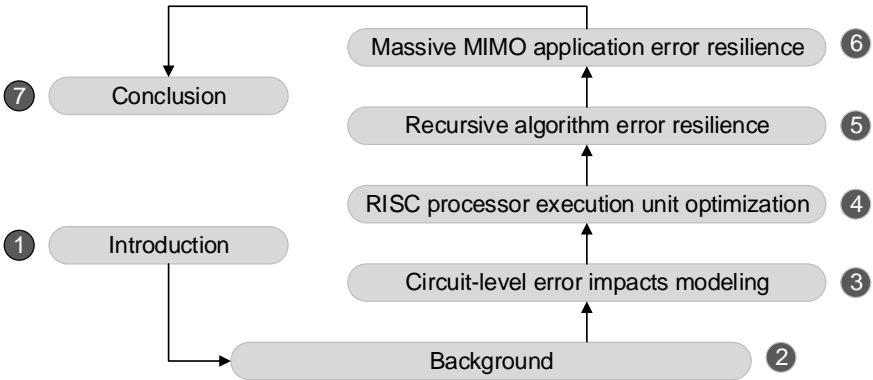


Figure 1.7: Structure of the thesis and overview of the chapters.

Chapter 2

Background: pessimistic safety margins should be exploited

This chapter reviews techniques for power and quality trade-offs. It firstly introduces PVT (process, voltage, and temperature) variations and reliability threats. It then reviews the worst-case design and the adaptive scaling method that tunes supply voltage to save power. Finally, the benefit of the cross-level VOS (Voltage Over-Scaling) approach is discussed.

The rest of this chapter is structured as follows: Section 2.1 explains the variations from process, environment and runtime changes. Section 2.2 analyses two conventional methods, i.e. the worst-case approach, and dynamical voltage scaling, which handle variability. The limitation of the worst-case approach is pointed out. The benefits and disadvantages of adaptive scaling are also reviewed. Section 2.3 discusses adventurous VOS methods that reach a balance between power consumption and output quality. Finally Section 2.4 concludes this chapter.

2.1 Uncertainties in circuit parameters

IC design has always been subject to variations which make it impossible at design time to determine exactly how a circuit will perform. Worse still, these

uncertainties have become increasingly significant as a result of the scaling of technology. To cope with those variations, the concept of design margin has been introduced in the design process.

This section reviews the cause, and more importantly the effects, of the long-term and short-term variations, from a digital circuits and systems designer's perspective. This thesis does not try to model for these variations directly. Instead, it saves power by reducing over-pessimistic safety margins.

2.1.1 Process variations and aging effects

Process variations and aging effects are long-term or permanent uncertainties for IC. They result to parametric variability (in area, speed, power consumption) and functional breakdowns. The parametric variability effects are reviewed in this subsection, because they usually lead to pessimistic parametric safety margins. Functional breakdowns, e.g., stuck-in errors, electrical stress, burn-in, [Veendrick, 2008], are not covered in this work. The reason is that those extra threats have very specific characteristics that require individual studies. Another reason is that, most of them have already been addressed properly in the subjects of design-for-testability [Fujiwara, 1985] and design-for-manufacturability [Strojwas, 1989; Chiang, 2007; Orshansky, 2007].

Time-zero process variations

Spatial process variations are deviations of IC parameters compared to their targeted values at the design time. They are created by the limited controllability of a manufacturing process [Nourani, 2006]. The origins of these variations are categorized into inter-die (global) and intra-die (local) components [Ghosh, 2010]. As plotted in Fig. 2.1, the global components consist of between-lots, between-wafers, and within-wafer variations. The local components are the within-die variations.

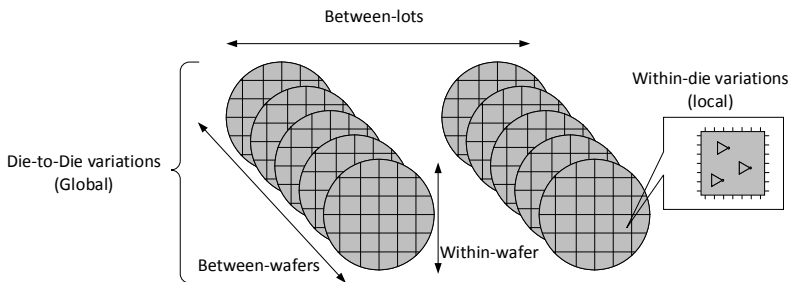


Figure 2.1: Spatial process variations consist of global and local components.

Global variability refers to the parameter changes for identical devices/interconnects separated by a longer distance, or fabricated at a different time, that result from factors such as processing temperature, equipment/tool properties, etc. between different runs, lots, wafers and dies [Saha, 2010]. Therefore, they impact equally on all transistors and interconnects on a die [Bowman, 2009a].

On the other hand, local variability causes parameter mismatch between identically designed devices/interconnects across a short distance within a die [Saha, 2010; Ohnari, 2013]. Typical causes are fluctuations in length, width, oxide thickness, flat band control, and the number of dopants [Herr, 1986].

Conventionally, the main focus was on global variations. This is convenient for designers, as all transistors on the same IC can be modeled with an equal offset. However, local variations have recently entered into the interest zone, and have been pronounced more as a consequence of the aggressive technology scaling, increased chip area, clock frequency, and leakage power distributions [Boning, 1996; Tschanz, 2002; Bowman, 2002]. Moreover, it was observed that local variations can have a much higher impact compared to global variations, e.g. lithography and etch technology can achieve 5% mismatch of wafer-scale metal width uniformity, whereas within-die variations were reported on the order of 15% [Dai, 2001].

The most prominent sources of the variabilities in nano-CMOS transistors are analyzed in [Wang, 2011; Asenov, 2007], which are Front-end-of-the-line (FEOL) random process variability sources, namely Random Discrete Doping (RDD), Line Edge Roughness (LER); PolySilicon (Poly-Si) Granularity (PSG), Metal Gate Granularity (MGG), and Oxide Thickness Variation (OTV).

The time-zero spatial process variations impact delays of a deeply scaled technology. Fig. 2.2 shows the circuit speed with randomized transistor width and depth, using the Monte-Carlo method. It demonstrates that transistor width and depth variations lead to 25% speed difference (3σ) in the standard

28nm CMOS process.

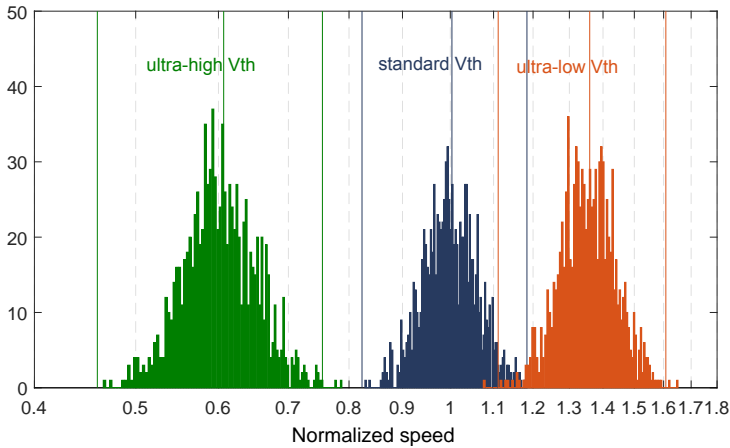


Figure 2.2: The speed (inverse delay) of 28nm fan-out of 4 (FO4) ring oscillators (RO) follows a Gaussian distribution because of process variability. Global and local variations are applied to the width and depth of transistors. The vertical lines indicate the mean speed and the 3σ speed.

Time-dependent aging effects

A device will degrade over time, according to not only temperature impacts, but also workloads (e.g. voltage, frequency, duty-cycle) [Kükner, 2013; Stamoulis, 2016]. These problems are prominent than ever before in the sub-20nm technology nodes [Wilson, 2013]. A throughout model that contains both reliability characteristics and system-level behavior modeling is thus required. An example is illustrated in [Chen, 2014].

As of today, the de-facto solution to the temporal degradation, during design, is to add another layer of margin. This margin is therefore stacked onto the existing process voltage and temperature (PVT) margins (discussed in Section 2.1). While temporal degradations are not directly addressed in this thesis, the model introduced in Chapter 3 can extend to temporal degradation impacts. Furthermore, the trade-offs between reliability (quality) and power savings (by margin shaving) in Chapter 5 and Chapter 6 can also include

temporal degradation margins.

2.1.2 Environmental and runtime uncertainties

In addition to (semi-) permanent variations, plenty of environmental and runtime threats also put the digital system quality in danger. These uncertainties are very difficult to model at design time. Therefore, they are assumed with the worst-case corner at design time. Luckily, they usually only affect IC behavior temporarily. Once the uncertainty resources are removed, the circuitry will return to normal condition. Therefore, those uncertainties only modify the circuit computation results, without permanent damages. Occasionally operating outside the specified region is acceptable, as long as the according errors are handled. Although some environmental or runtime conditions (e.g., permanent dose radiation error, mechanical vibrations and shocks, and electrostatic discharge) will fail chips irreversibly, those impacts are easier to deal with. Those permanent environmental impacts are beyond the scope of this work.

Supply voltage variation

Voltage variation is caused by non-idle power generators/regulators and IR-drop in the power delivery network. Fig. 2.3 shows that the voltage fluctuation from a 28nm power regulator can be as high as 100 mV (peak-to-peak).

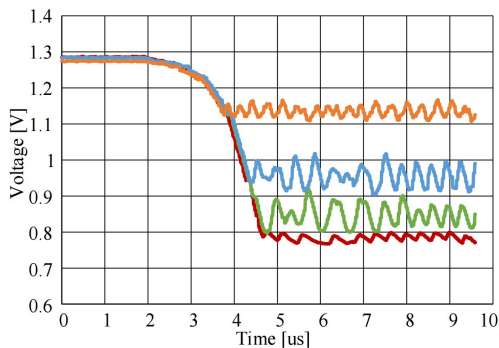


Figure 2.3: The supply voltage of an IC is fluctuating. The output of a state-of-the-art 28nm power regulator is shown [Rachala, 2016]. Curves represent different output voltage targets.

In addition to the power source, the power delivery network also leads to voltage drop when currents are flowing through. This is called IR drop. This effect can be modeled with EDA tools so that the delay of each gate is modeled with an individual voltage.

Temperature variation

Integrated chips are also affected by the temperature. The usual temperature environment of an IC is -40 to 120 °C. Furthermore, the circuit consumes energy and hence dissipates heat. This heats up the IC locally and temperature hotspots are produced. Counter-measures are needed for temperature variation, especially for complex multi-core processors [Chaparro, 2007].

The temperature and voltage combined effects on 28nm circuit speed is plotted in Fig. 2.4. In the regime of sub-1.1 V, higher temperature boosts up circuit speed. Reducing voltage decreases circuit speed linearly, down to threshold voltage (V_{th}).

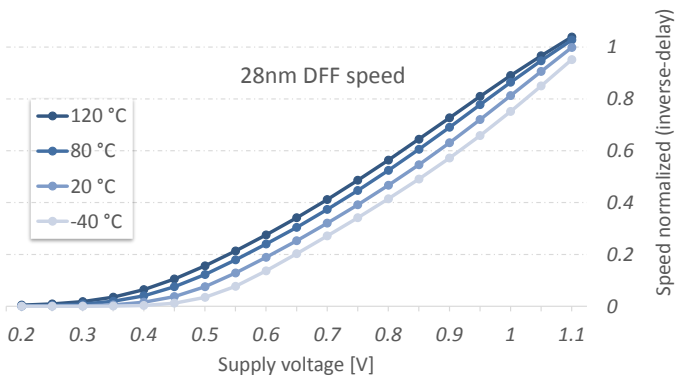


Figure 2.4: The speed of a 28nm circuit changes along with temperature and supply voltage.

SEU (Single Error Upset)

SEU is logic bits flipping triggered by external high-energy cosmic rays (e.g. protons and neutrons), and by α -particles from package [Pavlov, 2008]. It is also called soft-errors. Fortunately, the earth magnetic fields shield a majority of the cosmic rays. Consequently, in the past, soft-errors are only relevant to space applications [Baumann, 2002; Lesea, 2005]. Nonetheless, this view has been

challenged, mainly due to the following two reasons. Firstly, with more and more data processed and stored in the cloud, the memory in these data centers easily exceeds hundreds of petabyte. Even though the individual error-rate is still negligible, the combined reliability threat should never be overlooked [Feng, 2010; Kleeberger, 2013]. Secondly, the amount of mission-critical applications has or will increase substantially, spreading from the conventional niche industrial control market to daily usage scenarios. For the autonomous driving application, every decision in every car is critical [Marchiò, 2014]. In these systems, soft-errors become an issue that must be solved.

Soft-errors can be mitigated by using redundant circuits or systems, in the hope that they will not all fail at the same time. For instance, Chapter 3 presents a model to help design systems with less redundancy. Note that high-energy particle rays also lead to permanent degradations. This degradation is not covered in this thesis because those degradations are mostly relevant to space applications, which is out of the scope of this work.

VOS (Voltage Over-Scaling) hardware errors

The other transient reliability threats are errors caused by hardware operating at risky situations. This category is different from the previous two in that, the reliability risks can be tackled at design time, but they are intentionally kept risky, knowing that they can be handled by higher level designs.

By applying VOS [Hegde, 2004], the energy consumption reduces quadratically according to the voltage, V_{dd} , while the delay only increases linearly. Although energy savings are achieved, the drawback lies in the mis-captured data for the memory elements that are caused by the reduced delay. For instance, designs with Razor FF error detectors [Ernst, 2003] produce sparse timing-errors that need to consider. The ANT techniques [Hegde, 1999b] intentionally introduce computation errors, in the hope that they will be removed at higher levels.

The Razor FF and the ANT techniques are further investigated in Chapter 5. In that chapter, error resilient VOS designs are employed to save power for a digital accelerator with algorithm-level optimization. Moreover, Chapter 6 presents a power reduction potential in a Massive MIMO application, by embracing

light-weight but erroneous hardware.

2.2 Adjust supply voltage to trade quality for power savings

The dynamic power consumption of a digital IC scales with V_{dd}^2 , where V_{dd} is the supply voltage. Therefore, digital circuit designers usually reduce V_{dd} (VOS) for power savings. The savings are error-free, as long as the signal setup timing constraint is satisfied [Kulkarni, 2015]. However, the critical (minimum) V_{dd} that guarantees setup-timing closure cannot be determined at the design-time due to permanent and temporary variations. Consequently, hardware errors might be introduced: for logic components, the signal from the longest propagation paths are mis-captured [Han, 2013]; for memory components, this leads to incorrect write/read data/address or data loss [Karl, 2005].

As shown in Fig. 2.5, methods for selecting the V_{dd} are summarized into three categories, i.e., the worst-case corner, adaptive scaling, and error-resilient VOS (Voltage Over-Scaling).

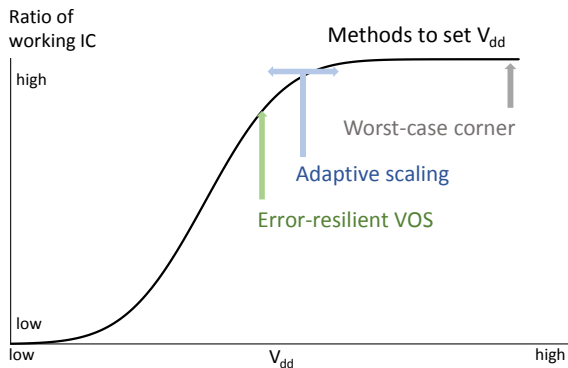


Figure 2.5: Worst-case corner, adaptive scaling, and error resilient VOS (Voltage Over-Scaling) approaches all cope with speed variability. The worst-case corner leaves energy savings on the table. The adaptive scaling (DVFS and AVFS) provides just needed V_{dd} for error-free operation. Continuing to reduce V_{dd} (VOS) saves more power, but errors will occur. This calls for error-resilient design techniques.

2.2.1 Limitations to the worst-case corner approach

Conventionally, the worst-case corner approach is applied to manage the supply voltage. All chips are set to a fixed and more-than-enough V_{dd} , to meet the rarely occurring worst cases. Corner-files are usually used for circuit guard-banding. An example on describing the corner cases of process variations is drawn in Fig. 2.6. These files describe the worst-case, the typical-case, and the best-case delay values of standard-cells.

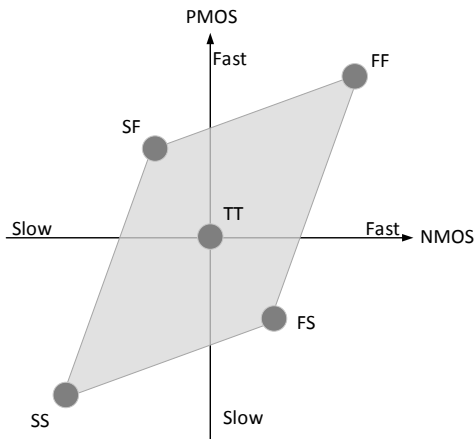


Figure 2.6: Spatial process variability are described by slow and fast corner-case files. This figure can be seen as a top-view of the pyramid in Fig. 1.4, where at the application-level TT is desired while all corners are ensured at the circuit-level.

With the worst-case design approach, logic synthesis is performed for the slow-case process corner. However, these corner files lack detailed information on local within-die variations. Instead, global on-chip variation margins [Chang, 2012; Stine, 2007] are added for all transistors. The global timing margins assume pessimistically that all devices within a die are performing according to their worst-case process conditions. Another pessimistic assumption is that all chips always operate at the worst-case in terms of voltage and temperature. The gap between the worst-case and the typical case is large. For instance, Fig. 2.7 shows that for a 28nm digital circuit, the performance difference (in terms of speed) is as large as 2.2x.

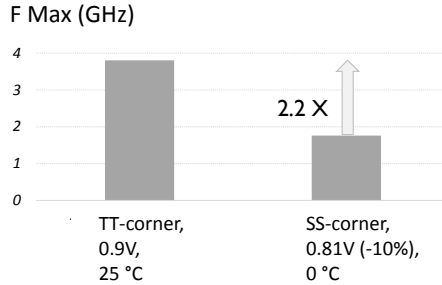


Figure 2.7: The speed of a 28nm circuit exhibits a 2.2x difference between the typical-case and worst-case corners. Results are obtained by Spice simulation. The worst-case assumes the most pessimistic PVT variations.

These worst-case conditions are often extremely rare combinations of complex interactions across an IC, which are almost impossible to predict at design time. Hence, as is often stated, the conventional worst-case design style leaves too much performance and power efficiency on the table.

2.2.2 Adaptive supply voltage schemes

Adaptive supply voltage schemes minimize the worst-case supply voltage margin used to account for PVT variations, using a suitable feedback signal to close the control loop. It finds the most optimal V_{dd} for each chip, at chip setup stage (post-silicon) [Kulkarni, 2006], or periodically (runtime) adjust depending on the workload [Pillai, 2001; Martin, 2002; Horvath, 2007], and environments [Martin, 2002; Das, 2006; Herbert, 2009; MiroPanades, 2014]. This method reduces the V_{dd} and hence reduces the power consumption.

A stereotypical scaling scheme is illustrated in Fig. 2.8. At run-time, the speed detector checks whether the DUT circuit has failed. This information is fed to a supply voltage control unit to adjust the V_{dd} . If no or very few errors are detected, the V_{dd} will be scaled down to save power. The supply voltage controller adjusts the V_{dd} slowly (in a coarse-grained temporal manner, e.g. every thousands/millions of cycles), for two main reasons: i) slowly adjusting saves power in the supply voltage controller itself; ii) the transition delay for modifying V_{dd} is inherently much larger than the clock period of the core circuit.

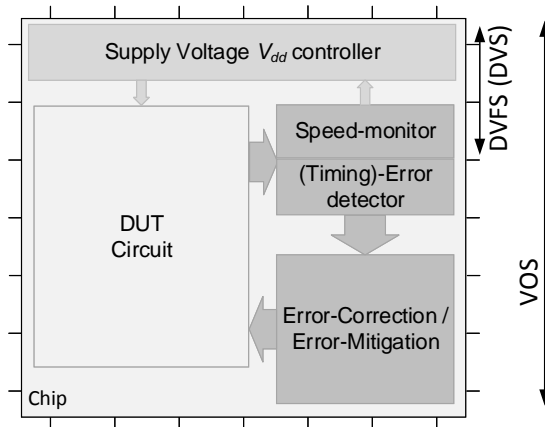


Figure 2.8: The dynamical scaling method handles speed variability. Moreover, the error-resilient V_{dd} scheme also handles circuit timing-errors. The dynamical scaling approach utilizes speed monitors and V_{dd} controllers. The VOS approach equips extra error detection and correction/mitigation units.

The runtime adaptive schemes are divided into two sub-categories: dynamical voltage frequency scaling (DVFS) [Nowka, 2002; Skadron, 2004; Calhoun, 2006] and adaptive voltage frequency scaling (AVFS) [Burd, 2000; Das, 2006; Elgebaly, 2007; MiroPanades, 2014]. DVFS uses what’s called open-loop scaling. A system with DVFS schemes listens to the change of system requirements. The hardware vendor determines the optimal voltage for the chip based on the target application and frequency. DVFS is not calibrated to any specific chips. Instead, vendors create a statistical model that predicts what voltage level a chip that’s already verified as good will need to operate at a given frequency. For example, our LDPC decoder designs [Li, 2015c; Li, 2015b] are characterized for different throughput-voltage combinations. Therefore, the system can opt to reduce V_{dd} , at run-time, if the required throughput is low. Note that these frequency-voltage sets are verified at the worst-case corner. So timing margins remain.

Although sometimes used interchangeably, AVFS, in contrast, uses a closed-loop system in which on-die hardware mechanisms manage the voltage — by taking real-time measurements of the junction temperature and current frequency, and adjusting the voltage to match them. This method eliminates the power waste discussed above by reducing the traditional guard bands that are required to ensure proper operation of every piece of silicon. AVFS can detect the circuit speed directly or indirectly. Indirect methods are observing the temperature and supply voltage that affects the circuit speed [Tschanz, 2007]. A replica

circuit [Kao, 2002] provides direct hints to the circuit speed. By monitoring the replica circuit, the actual speed of the circuit, which might be difficult to measure, can be guessed. However, these prediction schemes suffer from the delay mismatch between the replica and the actual critical path caused by within-die variations. In another word, the inaccuracy of the timing prediction limits the full exploitation of the variability design margin.

Therefore, in-situ timing-error detectors are proposed to measure the circuit speed. Canary FF [Calhoun, 2004] compares the results on a flip-flop (FF) with a redundant FF that captures a delayed input. It warns the circuit when potential timing-errors are about to occur. The operating condition when the first warning is produced is called the point of first warning (PoFW). The circuit design of Canary FF is explained in Section 5.3.2. Chapter 5 uses canary FF as the benchmark to demonstrate the power savings of the margin shaving.

An example of potential adaptive scaling benefits is shown in the Appendix A (published in [Huang, 2016a]). It presents a digital front-end for a polar RF transmitter, that demonstrates 70% speed gain or 33% power saving potential if the adaptive scaling method is applied.

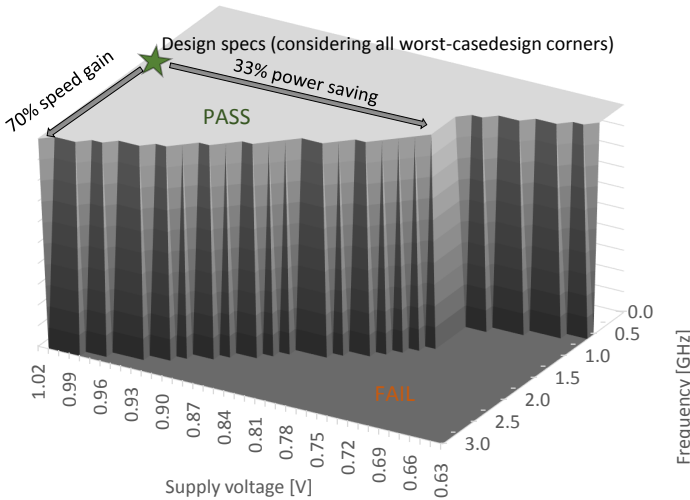


Figure 2.9: The operation region of a digital front-end IC can be much wider than conservatively assumed. The IC is measured at 25°C. Original data is published in [Huang, 2016a].

These adaptive scaling schemes provide error-free power savings. However, because of transient degradation, e.g. supply voltage noises (IR drop), V_{dd} will

drop occasionally [Dietel, 2014]. To avoid errors, the circuit either boosts V_{dd} up again quickly or guards with an additional margin from the PoFW. Both methods waste power. Therefore, a more aggressive version of adaptive scaling, VOS, is proposed. It shaves more design margins at the cost of infrequent timing violations.

2.3 Cross-layer error-resilient voltage over-scaling schemes

The third V_{dd} adjustment method is VOS with error resilient designs. When V_{dd} is further scaled down (lower than the safe operating condition), infrequent setup timing-errors occur. However, through adequate error resilient designs, these errors are be detected and corrected. In other words, the cross-layer optimizations are performed to handle errors. This goes beyond the dynamical scaling scheme where only environmental and runtime information is shared across design levels.

An error-resilient VOS scheme operates in a closed-loop as shown in Fig. 2.8. In addition to V_{dd} adjustment loop, it also utilizes an error detection and correction loop. The most popular error resilient approaches, i.e., in-situ error detection and corrections, and ANT techniques, are discussed in the following subsections.

Signals on timing critical paths take longer to propagate. Therefore, they call for more consideration during VOS, since they are more likely to fail during VOS. An example is shown in Fig. 2.10, where the MSB usually has a higher error possibility than the LSB for a digital adder. This is especially true for carry-ripple adders, in which the carry propagates from the LSB to the MSB, making the delay of MSB longer [Liu, 2010].

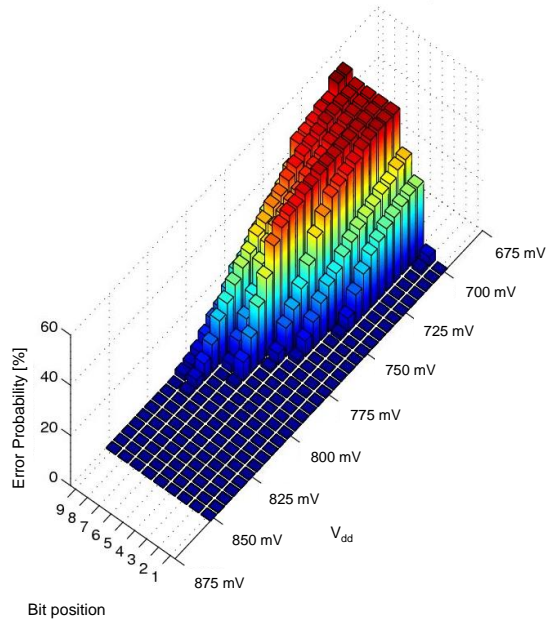


Figure 2.10: With VOS, errors firstly occur at MSB for carry-ripple adders.

2.3.1 In-situ error detection and corrections

For error detection, timing checks on flip-flops (FF) have been proposed and widely utilized. This scheme modifies the FF in the circuit. In-situ FF based schemes [Ernst, 2003; Bowman, 2009b; Nicolaidis, 2013; Fojtik, 2013] are suggested to overcome this mismatch for microprocessors. Razor I [Ernst, 2003] detects a timing-error by employing an extra shadow latch. In contrast to Canary FF, the Razor can find the exact timing slack for design margin shaving. The details of Razor flip-flops and its variants are discussed in Chapter 5.

For error correction, counter-flow [Ernst, 2003] and instruction-replay [Bowman, 2009b], that issues extra clock cycles, are proposed. They result in multiple-cycle throughput penalties once a timing-error is detected. Bubble Razor [Fojtik, 2013] reduces the throughput penalty to 1 cycle. However, the design is based on two-phase latches, which is difficult to be incorporated into mainstream EDA tools. Global clock gating scheme [Ernst, 2003] also achieves one cycle penalty. Nevertheless, it is difficult to implement for large area high-speed circuits. Recently, a local 1-cycle error correction scheme is proposed [Shin, 2013].

However, the timing constraint for its error flag signal becomes a challenge for multiple fan-in situations.

The Razor techniques [Ernst, 2003; Bowman, 2011; Bull, 2011; Fojtik, 2012] predicts the PVT variation at circuit-level and exploits the design margin by providing minimum but sufficient V_{dd} to the chip. [Bull, 2011] reports 30% and 52% power consumption saving on a typical die and a fast die, respectively; [Fojtik, 2012] achieves 54% saving on a typical die and 60% saving on a fast one. A potential hazard of applying this technique is massive throughput reduction when V_{dd} is dropped lower than the sufficient voltage. In this situation, the signal processor would terminate the processing to meet the throughput requirement, which leads to output quality degradation.

The infrequent errors, occurred in the time frame of nano seconds, are resolved by micro-architectural level error correction schemes. The supply voltage controller adjusts may the V_{dd} every tens of seconds,

2.3.2 Arithmetic noise tolerance

The arithmetic noise tolerance (ANT) techniques [Shim, 2004; Hegde, 2004; Narayanan, 2010; Karakonstantis, 2009] save power in digital signal processors by gracefully sacrificing the signal-to-noise-ratio (SNR), admitting that a certain amount of errors might occur. They detect errors by algorithmic comparison. For example, [Hegde, 2004] detects errors by observing the error-prone results, to check if it is within a reasonable range. The mitigation is accomplished by temporal or spatial redundancies. As a consequence, the setup timing-errors during V_{dd} scaling are translated into graceful signal quality degradation.

ANT is applied to systems that deal with soft output requirements. That is, the output quality is not measured as a yes or no criteria. Instead, it is in the gray zone for quality. For instance, signals in a wireless communication system are measured by their SNR, EVM, or BER; signals in an audio player are measured by the PESQ value. In these soft output systems, a certain degradation can be acceptable.

Numerous designs [Hegde, 2004; Narayanan, 2010; Karakonstantis, 2009; Huang, 2014; Shim, 2004; Huang, 2016d] are proposed to reduce the hardware errors at a given power budget. For instance, Fig. 2.11 provides an example of energy saving in 65nm COMS FIR filter brought by VOS, at the cost of SNR degradation. Besides, [Shim, 2004] utilizes reduced precision redundancy to reduce the power consumption by 40% on a digital FIR filter at the cost of slightly degrading the 23 dB SNR signal into 22 dB, and by 35% for a 64-point FFT when lowering the SNR from 55.5 dB to 55 dB.

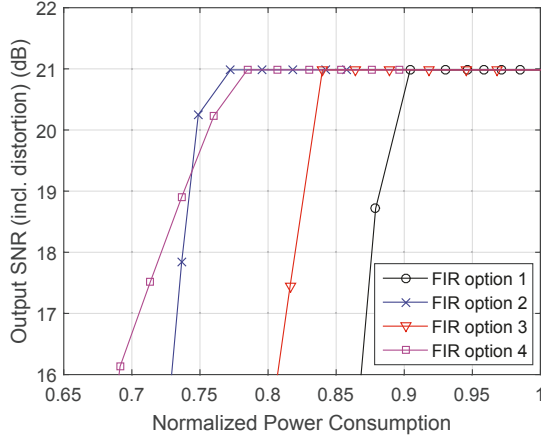


Figure 2.11: VOS with error resilient techniques save power on FIR filter, at the cost of signal quality degradations [Liu, 2010].

2.4 Conclusion

PVT and other environmental and runtime variations challenge the quality of digital circuits and system. The traditional worst-case, which introduces big safety margin, waste power consumption. In 28nm CMOS technology, the safety margin is as high as 2.2x the speed potential. Consequently, careful trade-offs on power and quality that considers across all design levels are encouraged.

Dynamical and adaptive scaling with on-chip timing-error monitors saves power by providing a just-needed supply voltage. The power saving will be maximized when the voltage is scaled down more aggressively where errors occur. In these situations, error-resilient designs are crucial to mitigate errors and ensure quality.

Chapter 3

Gate-level error impacts modeling analysis for random error mitigation

This chapter presents a gate-level random error model. The model, named as SERIAL – Significance RankIng ALgorithm, is an analytical approach that eliminates the pain of traditional time-costly Monte-Carlo simulation. It ranks the FF in a digital circuit according to their contribution to the outcome. The efficiency and effectiveness are shown on benchmark circuits. For instance, the computation of the algorithm can be finished within 30 seconds for a 64-point FFT accelerator (52k gates). With the ranking, circuit designers have the opportunity to selectively ensure the most important FF (e.g. FF hardening, VOS margin), without excessive hardening overheads. On an FFT circuitry, the algorithm helps to reduce the hardening overhead of 100% FF into 45%. The work in this chapter is unveiled in [Huang, 2016b].

The rest of this chapter is organized as follows: Section 3.1 presents the motivation. Section 3.2 reviews pre-existing approaches and highlights the contribution of the proposed SERIAL model. Section 3.4 explains the model and the algorithm to calculate it. Section 3.5 verifies the scalability and effectiveness of the model. Section 3.6 applies the model to facilitate the design

of an FFT processor. Finally, Section 3.7 concludes this chapter.

3.1 Demands for a reliability model

System reliability has been a major concern since the beginning of electronic design age [Siewiorek, 2014]. Traditional major threats are yield-related manufacturing faults (stuck-at) [Mei, 1974], space-radiation incurred soft-errors [Hazucha, 2000], and wearing out/aging degradation [Yamabe, 1985; Vattikonda, 2006]. Recently, VOS techniques introduce another reliability threat. These techniques save energy by design-margin shaving, e.g. Razor [Ernst, 2003] and ANT [Hegde, 1999a]. However, errors are deliberately introduced during the margin shaving process. These traditional and newly introduced reliability threats need to be handled. Otherwise, system quality is at risk.

Circuit and system designers used to takes a ‘best effort’ design flow. They apply reliability enhancement techniques with the best effort to ensure reliability. The most effective methods include ECC on memory [Slayman, 2005], FF and SRAM hardening [Jahinuzzaman, 2009], design rule check (DRC) modification and other safety margin insertion methods. The enhancement overhead ranges from as low as 12% for a SECEDED (Single Error Correction and Double Error Detection) Hamming code [Richter, 2008], to as high as 200% for TMR (Triple Module Redundancy). Designers usually verify the application-level reliability at the very end of the design process. Following this design flow, designers usually have no knowledge of which unit contributes the most to the system reliability, which is to say, we cannot distinguish which part is necessary and which part is over-design.

The ‘best effort’ design flow, although easy to implement, either leaves safety margin wasted, or requires modifications at the very end (sometimes even leads to another round of IC process tape-out). A more efficient design flow is to model the reliability factor in every phase of the design process. As a consequence, methods to tackle these issues can be performed effectively and timely.

To tackle the reliability threat, a model to quantify impacts of reliability threats is needed. With the help of the model, RTL designers can provide just-needed reliability counter-measures, avoiding over-designing margin in chip power and

area. The following section discusses prior modeling studies.

3.2 Modeling and enhancement techniques for reliability

Reliable system design depends on accurate and easy-to-use reliability models. Besides, it also requires effective and low-cost techniques to enhance reliability, once the system fails to meet the reliability target. This section discusses works on both aspects, as a background for this chapter.

3.2.1 Error modeling

The works on reliability modeling consist of two categories: i) error generation and ii) error propagation (Fig. 3.1). The partitioning is the same for both traditional errors (e.g. soft-error and stuck-at error) and VOS-induced errors.

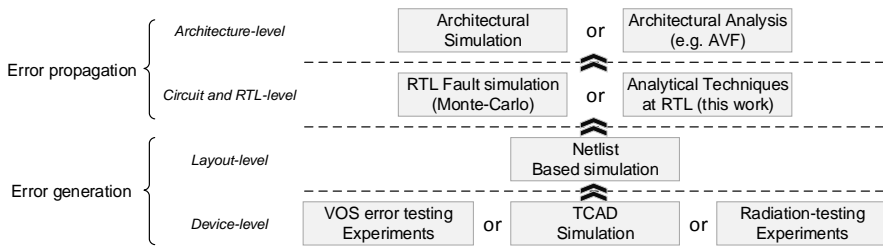


Figure 3.1: Modeling process and structure for error impacts.

Error generation

The principal difference among various error sources lays in the error generation. The error-generation model of soft-errors demands radiation-testing experiment and device-level simulation. It also requires standard-cell level analysis of error production. The model is finalized by the layout-level (e.g. Spice) single-cycle analysis to address effects of logical masking, electrical masking, and vulnerable window [Ebrahimi, 2015].

In contrast, to model VOS-induced error generation, device-level reliability issues and timing (signal delay) degradation are analyzed at reduced V_{dd} and

IR-drop noise. The layout is simulated (e.g. using Spice), to provide error generation information.

Error propagation

The error propagation model describes the error impact on algorithm output, or on application-level performance. Therefore, the propagation model is uniform, regardless of the source of errors (soft-error and VOS error).

Several estimation techniques for reliability have been proposed to investigate the behavior of circuits under faults. For instance, ad-hoc investigations into digital circuits are carried out for specific applications [Gaisler, 2002; Karakonstantis, 2012b; May, 2008; Murali, 2005]. However, they require an in-depth understanding of both reliability and the circuit design, which are not always available.

Therefore, a more generic design approach is coveted. Within this context, the Monte-Carlo simulation (e.g. [Holcomb, 2009; Clark, 1995; Mirkhani, 2015]), taking the pre-computed error generation information, has become the most popular method for gate-level error propagation modeling.

Whole system simulation on AVF (Architectural Vulnerability Factor) modeling [Mukherjee, 2003] are performed, taking the results from gate-level models. Recently, [Biswas, 2008; Wang, 2013] discuss the vulnerability behavior of faulty structures at the architectural level. [Polian, 2011] suggests a transient error model. It analyzes the possibility that soft-errors reflect on the output. Although it predicts the occurrence rate of an output error, it cannot model the consequent severity of errors, which however is essential.

3.2.2 Reliability enhancement techniques

This subsection discusses reliability enhancement techniques that will be utilized when the modeling is complete.

Many reliability enhancement techniques have been proposed [Ottavi, 2015; Haghi, 2009]. At gate level, [Knudsen, 2006; Haghi, 2009] introduced hardened flip-flops (FF hardening), which replace the traditional flip-flops, to reduce single-event upset without large area and power overhead.

Various mitigation schemes to cope with the impact of soft-error have been investigated. For memories and communication systems, errors can be corrected by redundant data, e.g. error correction code [Ejlali, 2007; Mitra, 2005]. For digital circuits, selective triple modular redundancy [Samudrala, 2004;

[Bolchini, 2007](#)] is effective in reducing the chance of transferring the circuit-level faults to micro-architectural level errors.

Besides the efforts in reducing the application-level occurrence of errors, designing systems that operate reliably even with the presence of errors is also of great interest. In these soft-quality requirements systems, errors are acceptable as long as the system output still meets the requirements. For instance, the transient error in a wireless communication system can easily be tolerated by the symbol detector, as long as the error vector magnitude is allowed [[Karakonstantis, 2012b](#)].

Despite the green-power benefit of VOS, the circuit reliability is at risk of timing-errors and noise contamination. Therefore, it is either suggested to apply an additional margin to guarantee reliability [[Kunitake, 2011](#)] or to implement error handling schemes [[Ernst, 2003](#)]. Both methods introduce overhead.

Considering that all the error mitigation methods discussed above introduce area and power expenses for the system, circuit designers must carefully investigate the probability and characteristics of reliability risks, as well as their consequences, to provide just-needed reliability enhancement efforts.

Although methods for error mitigation are proposed, they do not provide systematically guideline on which part should harden. This chapter identifies the bottlenecks in random error hardening, and serves as a guideline for selectively hardening.

3.3 Contributions of this chapter

The error propagation model method is essential in judging the impact of reliability threats. It can analysis errors with various generation mechanism. However, the general Monte-Carlo method does not scale to large circuits because of runtime constraints, even with the help of grouping of similar gates. In summary, it is of immense interest to provide analytical approaches to investigate logic gate behavior under faults.

This chapter presents an analytical approach (SERIAL) for gate-level modeling. The approach efficiently identifies the most significant flip-flops that contribute mostly to the output quality. As a result, all flip-flops (FF) and input ports are sorted out based on their significance to the final output. It analyzes algorithmic effects of errors (error magnitude), in contrast to [[Polian, 2011](#)] where only the chance of detectable errors were covered.

This chapter focuses only on finding out the most significant FF because they

are the starting and end point of logic signals. Soft errors can be mostly eliminated locally by applying FF hardening [Ramanarayanan, 2003]. Once the most significant FF are identified, circuit designers can exploit extra redundant FF or the corresponding logic (e.g. double/triple modular redundancy), to alleviate the reliability issue. From the VOS errors point of view, finding out the most significant FF helps to understand which FF need to be protected (from VOS errors). The only difference, compared to soft errors, is the error occurrence data, which can be obtained by Spice simulation.

This algorithm is challenging: errors might occur in random situations, and their impact is highly dependent on the circuit's states. Therefore, this work assumes that the input data and circuit's states are completely random. Another assumption is that signals coming from different FF to the same FF are independent. These assumptions make the model unrealistic at first sight. However, it still provides very useful information: the identified most significant FF are usually the clock-gating controllers, the state recording registers and loop controller, which are always essential for the system's reliability, regardless of input data and states. By only hardening the most significant FF (a small percentage of FF), the system's reliability will improve remarkably.

For an absolute correct result, all FF should function correctly, which is difficult to guarantee due to area/power constraints. This work answers the question which FF to protect, when only a small portion of protected is allowed due to overhead restricts. Note that the result of SERIAL serves as a relevant guidance for circuit designers to choose which FF to harden, yet, it does not provide a guaranteed reliability model.

This approach focuses on the effects of faults in digital circuits, rather than specific consequences of soft-errors. Therefore, it can analyze the effect of soft errors, VOS errors, and even the conventional stuck-at faults, provided that the corresponding error generation model is given.

3.4 SERIAL – a Significance Ranking Algorithm for error effects modeling

The proposed algorithm, SERIAL, takes the netlist of the digital circuits as the input, extracts the connection information between FF and in/out ports, and ranks all FF regarding their contribution to the final output. This section explains the algorithm in detail.

In this chapter, individual values are denoted by normal math symbols (e.g. df). The corresponding collections, e.g. sets, vectors, and matrices are denoted

as upper case math symbols (e.g. DF).

The **significance** of an FF or an in/out port is defined as the impact of its state change on the output. An FF with high significance implies radical system failure, once it is infected by an error. Therefore, to guarantee the system performance, circuit designers must identify the FF with high significance and then harden them. The significance on the output is initialized by the user of SERIAL users. The algorithm distributes the significance to all FF and input ports.

To compute the significance, the notion of significance graph is introduced. Fig. 3.2 shows an example of converting a netlist into significance graph.

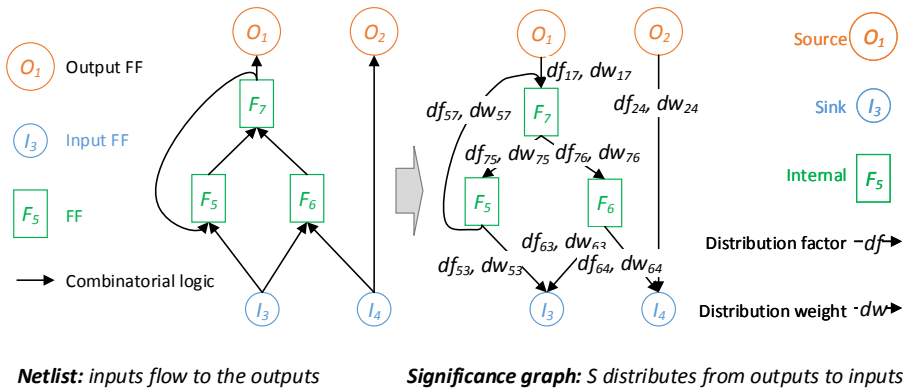


Figure 3.2: A significance graph inverts the signal direction of the corresponding netlist. Each node in the significance graph distributes its significance S to its neighbors through the arrows.

A **significance graph** is a weighted directed graph, which is formally modeled as a tuple $\langle O, I, F, S, DF, DW \rangle$. The elements are explained as follows:

- *Nodes*, which include:
 - *Source* (O): the output ports for the digital circuit are regarded as sources, since they distribute the significance to their neighbors;
 - *Sink* (I): the input ports;
 - *Internal* (F): the FF, as the intermediate nodes for the significance graph.
- *Significance* (S): The s_i of each node represents its contribution to the final output.

- *Arrows* represent the s distribution between nodes. Each arrow has two features:
 - *Distribution Factor (DF)*: it represents the portion of s distributed from one node to its adjacent tails. Therefore, the sum of the df_{ij} that start from the same node is normalized to a unity 1, representing the significance of the node is distributed to its adjacent nodes. For instance, in Fig. 3.2, $df_{75} + df_{76} = 1$.
 - *Distribution Weight (DW)*: it denotes the distributed s in the arrow. Therefore, $DW = S \cdot DF$.

The S of each node is determined by its adjacent heads, determined by the DF , which is shown as Equation 3.1:

$$\begin{bmatrix} S_F \\ S_I \end{bmatrix} - DF * \begin{bmatrix} S_O \\ S_F \end{bmatrix} = \epsilon \approx 0, \quad (3.1)$$

where S_F denotes the S of the FF (n in total); S_I represents the S of the m inputs, S_O is the significance of the k outputs; ϵ is the computational error (which is 0 for the exact solution); and DF is the transfer matrix (size $[n+m, k+m]$) for S :

$$S_F = \begin{bmatrix} S_{f_1} \\ S_{f_2} \\ \vdots \\ S_{f_n} \end{bmatrix}; S_I = \begin{bmatrix} S_{i_1} \\ S_{i_2} \\ \vdots \\ S_{i_m} \end{bmatrix}; S_O = \begin{bmatrix} S_{o_1} \\ S_{o_2} \\ \vdots \\ S_{o_k} \end{bmatrix}; \quad (3.2)$$

$$DF = \begin{bmatrix} DF_{O \rightarrow F} & DF_{F \rightarrow F} \\ DF_{O \rightarrow I} & DF_{F \rightarrow I} \end{bmatrix}. \quad (3.3)$$

The sum in each column of DF is 1. The initial values of S_O in Equation 3.1 are initialized by users. By default, they are all set to 1, implying the equal importance of the output ports.

The designer is free to set these initial values, according to the application requirement. For instance, if the output represents a 2's complementary binary number, it is advised to set the significance to 2^i for some applications, where i is the bit position.

The matrix DF is determined by the combinational logic between nodes in the significance graph, which is extracted from the netlist.

Equation 3.1 shows $n+m$ independent linear equations with $n+m$ unknowns (S_F and S_I). The number of unknowns is huge for practical digital circuits. For

example, there are more than 1M unknowns for a 1 mm^2 circuit in 28nm. This makes the exact solution ($\epsilon = 0$) unrealistic. However, DF is usually sparse, since not all FF are directly connected. This chapter proposes to use a heuristic algorithm to get an accurate-enough S (ϵ close enough to 0) for all FF. The structure of the method is summarized in Fig. 3.3.

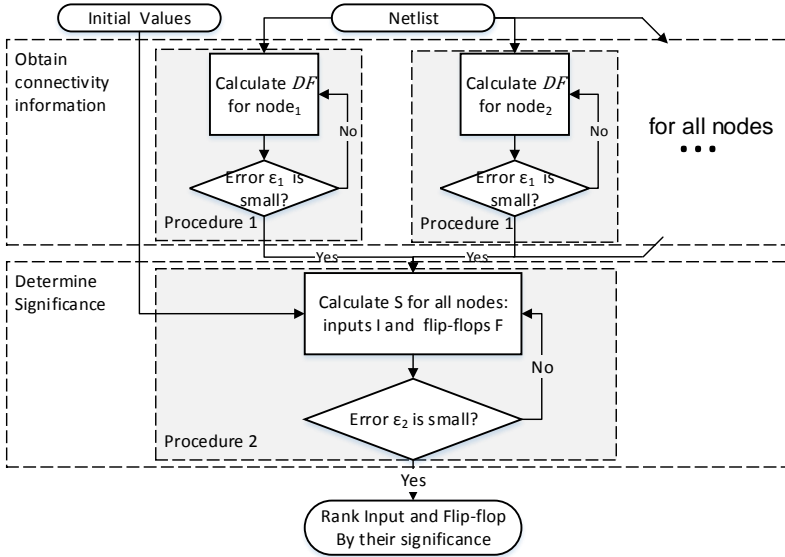


Figure 3.3: The SERIAL model consist of two procedures. It computes the connectivity before determining the significance.

The solver of the SERIAL model consists of the following components:

1. Initialize: set initial values S for output ports (S_O);
2. Compute DF (Procedure 1): determine the df from each node to all adjacent tails of it;
3. Compute S (Procedure 2): breadth-first search to update DW for each arrow, and hence S for each node (S_I and S_F).

3.4.1 Determine the distribution factor DF

For one node, its adjacent tails inherit different portions of S , since the combinational logic is different. Procedure 1 computes DF for one node (a row

of DF). Therefore, it should be executed for every node to obtain the complete DF .

The notation of a **logic graph** is introduced. Similar to the significance graph, it also reverses the signal transferring direction of the netlist. The **logic graph** is modeled as a tuple $\langle O, I, G, LS, LDF, LDW \rangle$:

- *Nodes*, which include:
 - *Source* (O): the aim of this logic is to compute the DF for this node;
 - *Sink* (I): all input ports and sequential FF;
 - *Internal* (G): the combinational logic gates are the intermediate nodes for the graph.
- *Logic Significance* (LS) for each node in the logic graph.
- *Arrows* denote the LS distribution between nodes. Arrows have two features:
 - *Logic Distribution Factor* (LDF): the portion of LS distributed from a node to its tails.
 - *Logic Distribution Weight* (LDW): the LS on the arrow.

The logic graph is a directed weighted graph. It contains branches and conjunctions, as the netlist might have branches and multi-port logic gates. The internal nodes are combinational logic gates, implying that the logic significance graph is non-cyclic (otherwise, the corresponding netlist contains cycles in combinational logics).

A head node distributes its LS to its tail nodes along the arrows. The distribution factor ldf is determined by the gate type: ldf is defined as the normalized possibility that an changed input data leads to changed output data. The input port corresponds to the tail node in the logic graph. The output port corresponds to the head node in the logic graph. Note that since the logic significance can only be distributed, the summation of ldf from a node is a normalized 1.

For instance, the procedure of computing the ldf for a 3-input AND gate (see Table 3.1) is discussed with its truth table. For the input port A, if the signal is independently changed from 0 to 1, the state of the AND gate might change from state [1->5, 2->6, 3->7, 4->8], respectively, depending on the initial signal of port B and C. Of all four possible changes, only the state change 4->8 leads to a changed output Z. This also applies to changing port A from 1 to 0. Therefore, for port A, the ldf before normalization is 2/8, meaning that 2/8 of random

changes in port A result to port Z value change. Considering that for input B and C the *ldf* are both 2/8 (by symmetry), the normalized *ldf* is therefore 1/3, since the sum of *ldf* for any node is 1. In fact, for any symmetrical logic gates, the normalized *ldf* is 1/*n*, where *n* is the number of input ports. This implies that all inputs contribute equally to the output.

Table 3.1: Truth table of a 3-input AND gate.

State	1	2	3	4	5	6	7	8	
in	A	0	0	0	0	1	1	1	1
	B	0	0	1	1	0	0	1	1
	C	0	1	0	1	0	1	0	1
out	Z	0	0	0	0	0	0	0	1

One possible approach to computing *DF* is enumerating the paths from the source node to each sink node. However, the number of unique paths increases exponentially with the number of branches and conjunctions, making the enumeration approach impractical. Therefore, this chapter proposes a heuristic breadth-first graph traversal algorithm to compute the *DF* (see **Procedure 1**). It records the *LS* for each node and visits each node for only once during an iteration.

Algorithm 1 Procedure 1: Determine the DF for all arrows that start from a node ($node_i$).

```

1: Input: logic graph of  $node_i$                                 ▷  $node_i.ls$  are initialized to 1
2: Output:  $DF$  for all arrows of  $node_i$ 
3: repeat                                                       ▷ loop until mismatch is small
4:   push  $node_i$  to the empty stack;
5:   while stack not empty do
6:     pop a  $node_j$  from stack;
7:     visit( $node_j$ );
8:     for each  $node_k$  in <adjacent nodes of  $node_j$ > do
9:       if  $node_k$  is a gate AND  $node_k$  not visited then    ▷ only when
        $node_k$  is not a end point of the logic graph
10:        mark  $node_k$  as visited;
11:        push  $node_k$  to the stack;
12:      end if
13:    end for
14:  end while
15: until error  $\epsilon_1$  is small enough
16: for each  $node_j$  in <FF and inputs> do                        ▷ end points of logic graph
17:   return  $node_i.node_j.df \leftarrow node_j.ls$                 ▷ return the ls as df
18: end for

19: function visit( $node_j$ )
20:   for each  $arrow_k$  that starts from  $node_j$  do                ▷ name the tail as  $arrow_k$ 
21:      $\Delta \leftarrow node_j.ls * arrow_k.ldf - arrow_k.ldw$ ;      ▷ new vs. existing ldw
22:     error  $arrow_k.\epsilon_1 \leftarrow \Delta / arrow_k.ldw * 100\%$ ;  ▷ error for proc.1
23:      $arrow_k.ldw \leftarrow arrow_k.ldw + \Delta$ ;                ▷ update ldw of  $arrow_k$ 
24:      $node_k.ls \leftarrow node_k.ls + \Delta$ ;                    ▷ update ls of  $node_k$ 
25:   end for
26: end function

```

When visiting a node, the algorithm updates the LS of its adjacent tails, as well as the weight for the adjacent arrows (LDW). Besides, the LDW change rate (ϵ_1) is recorded as the metric of computational error (see Procedure 1). ϵ_1 is checked at the end of each iteration to determine whether additional iterations are required.

Fig. 3.4 shows an example of computing DF for a specific FF ($node_b$). The netlist is converted to the logic graph in Fig. 3.4(b) (Note the direction of the arrows). The number of the node indicates the S , and the number on the arrow denotes the LDW . Fig. 3.4(c)-(f) shows the significance update in the first iteration.

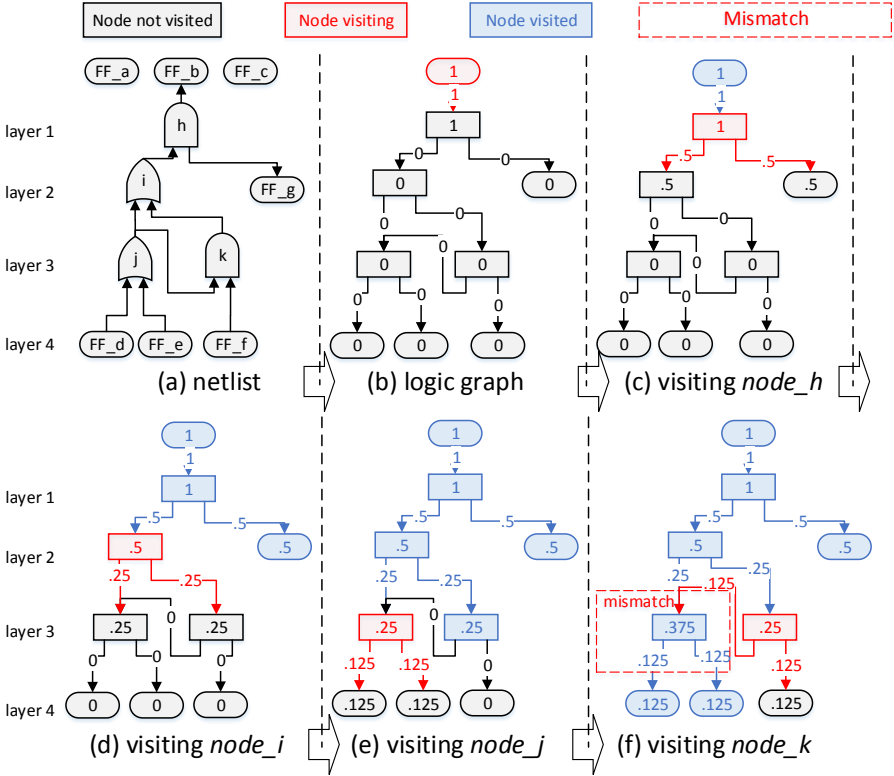


Figure 3.4: An example of computing the distribution factor DF (Procedure 1). (a): Netlist of the logic gates and FF within a pipeline stage; (b-f): Computing DF using Procedure 1 in the first iteration. The procedure distributes the $S = 1$ at FF_b to FF_d - FF_g .

A mismatch is observed in Fig. 3.4(f), in which the logic significance of $node_j$, $ls_j=0.375$, is not equal to the sum of the ldw of its adjacent arrows ($0.125+0.125$). This is because $node_j$ was visited in Fig. 3.4(e), where its ldw is updated. Therefore, $node_j$ will not be visited afterwards, even though its ls is updated in Fig. 3.4(f). Note that in the second iteration, the mismatch error on $node_j$ will be eliminated, since when visiting $node_j$, its ls is already correct.

The mismatch is generated because of the conjunction of arrows from different layers (nodes in the same layer means they have the same logic depth). For example, $node_i$ (in $layer_2$) and $node_k$ (in $layer_3$) are both connected to $node_j$ (in $layer_4$), which results in the mismatch in Fig. 3.4(f). The magnitude of the computation error ϵ_1 is used to estimate the mismatch of the computation

procedure.

To reach zero mismatches (zero error ϵ_1) for the logic graph, the maximum number of iterations is equal to the maximum logic depth. This is because, after each iteration, the correct logic significance from nodes of the same layer will at least propagate to the nodes of the next layer. In reality, ϵ_1 converges more quickly than the maximum bound, because, by the time a node is visited, it has probably got the correct update information (ldw) from all of its adjacent heads.

3.4.2 Determine the significance S

This section discusses the algorithm to solve Equation 3.1, which determines the significance for all nodes. **Procedure 2** shows the algorithm.

Algorithm 2 Procedure 2: Determine the S for all nodes.

```

1: Input: the significance graph.                ▷  $node_i.ls$  are initialized to 1
2: Output:  $s$  for all nodes.
3: repeat                                         ▷ loop until mismatch is small
4:   push output ports (sources) to the empty stack;
5:   while stack not empty do
6:     pop a  $node_j$  from stack;
7:     visit( $node_j$ );
8:     for each  $node_k$  in <adjacent nodes of  $node_j$ > do
9:       if  $node_k$  is not a input port (sink) AND  $node_k$  not visited then
      ▷ only when  $node_k$  is not a end point of the significance graph
10:        mark  $node_k$  as visited;
11:        push  $node_k$  to the stack;
12:      end if
13:    end for
14:  end while
15: until error  $\epsilon_2$  is small enough
16: for each  $node_j$  in <FF and inputs> do         ▷ nodes in significance graph
17:   return  $node_j.s$                                ▷ return the  $s$  for all nodes
18: end for

19: function visit( $node_j$ )
20:   for each  $arrow_k$  that starts from  $node_j$  do ▷ name the tail as  $arrow_k$ 
21:      $\Delta \leftarrow node_j.s * arrow_k.df - arrow_k.dw$ ;    ▷ new vs. existing  $dw$ 
22:     error  $arrow_k.\epsilon_2 \leftarrow \Delta / arrow_k.dw * 100\%$ ;    ▷ error for proc.2
23:      $arrow_k.dw \leftarrow arrow_k.dw + \Delta$ ;           ▷ update  $dw$  of  $arrow_k$ 
24:      $node_k.s \leftarrow node_k.s + \Delta$ ;             ▷ update  $s$  of  $node_k$ 
25:   end for
26: end function

```

Similar to Procedure 1, it is also a heuristic breadth-first algorithm. However, this problem is more complicated than Procedure 1, because the significance graph contains not only branches and conjunctions but also loops. By applying Procedure 2, each node is only visited once in an iteration, eliminating the situation of the algorithm being trapped in the loop.

Fig. 3.5 shows the example of computing the significance during the first and the second iteration. The initial significance s at O_1 is set as an unity 1, and the s for O_2 is 2. In Fig. 3.5(d), a mismatch can be observed on F_7 due to loops. This mismatch is measured by computational error ϵ_2 , and it will reduce in future iterations: during the first iteration, F_7 obtains a Δ of 0.25 from

$arrow_{F_5 \rightarrow F_7}$ (Fig. 3.5(c)):

$$S_{F_7} = 1 + \Delta = 1 + \frac{1}{4}. \quad (3.4)$$

The same $\Delta arrow_{F_5 \rightarrow F_7}$ becomes $0.25/2^2$ in the second iteration (Fig. 3.5(g)):

$$S_{F_7} = 1 + \Delta_1 + \Delta_2 = 1 + \frac{1}{4} + \frac{1}{16}. \quad (3.5)$$

This is because the loop $F_7 \rightarrow F_5 \rightarrow F_7$ contains 2 branches, i.e. F_7 and F_5 both have two adjacent tails. When the number of iterations goes to infinite, the Δ forms geometric progression with a common ratio of $1/4$, causing the final significance of F_7 to converge:

$$\begin{aligned} S_{F_7} &= 1 + \Delta_1 + \Delta_2 + \Delta_3 + \dots \\ &= 1 + \frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \dots = 1.3\dot{3}. \end{aligned} \quad (3.6)$$

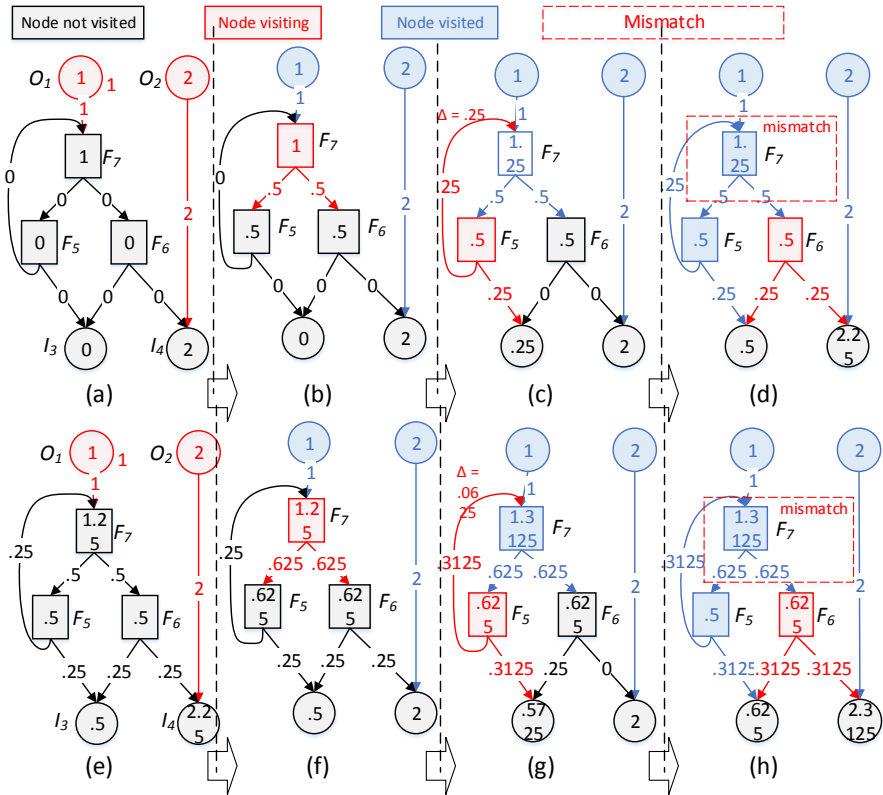


Figure 3.5: An example of computing the significance (Procedure 2) for the netlist in Fig. 3.2. (a-d): the first iteration; (e-h): the second iteration.

Since every loop must contain branches (otherwise the corresponding loop in the netlist does not contain influx/conjunctions, as the graph reverses the direction of the netlist), the common ratio of Δ must be smaller than 1, preventing the significance to go to infinity. Therefore, the computational error ϵ_2 is reduced with every iteration. Unlike Procedure 1 where a zero- ϵ_1 result can be expected after a fixed amount of iterations, graphs with loops can only produce zero error after infinite iterations. However, in practical digital circuits, loops are usually very large (lots of branches in the loop, and hence the common ratio is very small), the breadth-first approach ensures that the algorithm converges faster. This is demonstrated with benchmark circuits in Section 3.5.1. Moreover, the most significant FF normally have very high fan-in rates in the significance graph. As a consequence, they emerge on the top of the significance ranking list after several iterations.

Loops in the graph might result in a larger significance value for internal nodes than that of sources, e.g. SF_7 is greater than SO_1 . One example in digital circuits is the loop controller, which is more significant than the output ports, and thus should be protected more carefully.

Another important feature worth mentioning is the **confidence factor**. It is defined as the portion of significance each node absorbs from the arrow coming towards it. Typically it is fixed to 1, implying full absorbance from the head. If it is set to 0, the tail node of the arrow receives no significance (or logic significance) distribution from the corresponding head node. As a result, the significance only allocates to other interested FF. This is used for situations when circuit designers assure certain paths/components are secure (or important), where the significance is saved to the insecure components. In this work, the *clk* port and the *reset* port are assumed as hardened and thus are free from errors. Therefore, the confidences for the arrows connecting to them are fixed to 0.

3.5 Experimental verification

A proof-of-concept program written in Python was built to verify the proposed approach experimentally. It parses the input netlist into logic graphs and the significance graphs.

For benchmarking purpose, SERIAL is analyzed on a number of representative benchmark circuits: 6 ISCAS'89 circuits (e.g. s27) [Brglez, 1989], 3 ITC'99 benchmarks (microprocessors e.g. b14) [Corno, 2000], 8-bit 32-point and 16-bit 64-point FFT processors (FFT32 and FFT64¹) (developed from [Li, 2015d]), and a commercial-ready low density parity check (LDPC) decoder (developed from [Li, 2015b]). The ITC'99 circuits and LDPC are deliberately selected for their high complexity as each FF is connected to various FF.

3.5.1 Scalability

When performing Procedure 1 for one node, the visited gates are formed in tree-structure (see Fig. 3.4). Therefore,

$$\#gate_visited \sim degree_{node}^2, \quad (3.7)$$

¹FFT32: 8-bit resolution, 1 butterfly unit; FFT64: 16-bit resolution, 2 parallel butterfly units

where $degree_{node}$ is the average degree of connectivity for the graph (it represents the number of nodes that one node, i.e. I/O and FF, connects to). Since the runtime for each gate is the product of mean iteration number ($iter_1$) and the $\#gate_visited$, the overall complexity for all nodes of Procedure 1 (T_1) is summarized as:

$$T_1 \sim \#nodes * iter_1 * degree_{node}^2. \quad (3.8)$$

Note that $\#nodes$ scales linearly with the circuit size, while $iter_1$ and $degree_{node}$ depend on the circuit characteristics.

The runtime for Procedure 2 is summarized as

$$T_2 \sim iter2 * \#node * degree_{node}, \quad (3.9)$$

where $degree_{node}$ implies the runtime for visiting one node, $\#node * degree_{node}$ suggests the runtime for one iteration, and $iter2$ is the number of total iterations for Procedure 2.

To measure the scalability of the algorithm, the SERIAL algorithm was applied to all benchmark circuits, on a desktop PC with Intel i7 CPU and 8 GB memory. The runtime analysis is shown in Table 3.2. The benchmark complexity metrics that affect the algorithm execution time is listed.

Table 3.2: SERIAL model solver runtime analysis for benchmark circuits.

Circuits	Circuit complexity			Procedure 1 $epsilon_1 = 0$		Procedure 2
	$\#gate$	$\#node$ (IO & FF)	$degree_{node}$	mean $iter_1$	runtime	runtime per iter.
s27	28	8	2.6	2.25	0.2 ms	0.1 ms
s510	315	32	3.3	2.8	2 ms	0.8 ms
s641	392	78	6.6	3.5	9 ms	3.4 ms
s5378	3.3k	263	9.3	2.6	39 ms	18 ms
s13207	9.7k	852	6.3	2.6	97 ms	44 ms
s38584	28k	1.8k	12.5	2.8	292 ms	160 ms
b14	9.6k	233	90.7	14.4	5.7 s	158 ms
b20	21k	546	115	14.0	10 s	325 ms
b21	22k	546	129	14.6	12 s	369 ms
FFT32	11k	2.2k	8.8	2.8	0.63 s	188 ms
FFT64	52k	8.6k	12.6	2.9	6.1 s	965 ms
LDPC	778k	59k	136.9	4.2	1424 s	40 s

For Procedure 1, the runtimes for SERIAL on all ISCAS'89 circuits are all well below 1s. The runtime for the 52k-gate FFT64, comparing with FFT32, increases linearly with the node number, and quadratically with the $degree_{node}$. For LDPC and ITC'99 circuits, the runtime is relatively large, due to their high $degree_{node}$, which is the unique property of their memory addressing units. Note that the runtime for Procedure 1 can be substantially reduced by parallelism as the computing processes for all node are independent. For all circuits, the number of iterations to reach zero ϵ_1 for Procedure 1 are all small, confirming our assertion in Chapter 3.4.1.

For Procedure 2, the runtime per iteration is reasonably small for all circuits. Fig. 3.6 shows the computational error ϵ_2 convergence w.r.t. iterations. Naturally, ϵ_2 reduces with more iterations performed. After 40 iterations, even the most demanding circuit, LDPC, reaches a maximum of 3% computational error ϵ_2 for all arrows in the graph, which is very precise for the significance ranking. This also demonstrates the fast convergence of ϵ_2 , despite all the branches, conjunctions and cycles in the graph for practical circuits.

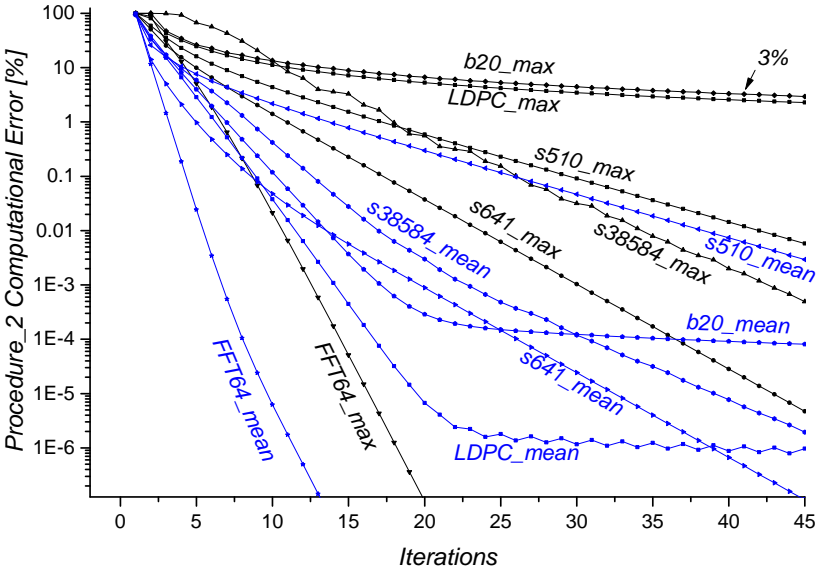


Figure 3.6: In Procedure 2, the mismatch error ϵ_2 converges rapidly towards 0.

In summary, considering that $degree_{node}$ does not change with the circuit size, the SERIAL algorithm scales linearly with the circuit size ($\#nodes$). The only time-consuming case is the exceptional circuit, LDPC, of which the $degree_{node}$ is

high. Despite that, the algorithm for all benchmark circuits finishes reasonably fast.

3.5.2 Validation on an LDPC decoder

After applying the SERIAL to the LDPC, each FF is labeled with its ranking order regarding significance. Fig. 3.7 (left y-axis) shows the significance distribution. A small portion (around 100) of FF has an extra-high significance value. These FF are mostly found in the control logic and clock-gating units. These high significance FF control lots of outputs. Moreover, the faults generated by these components cannot be flushed away easily.

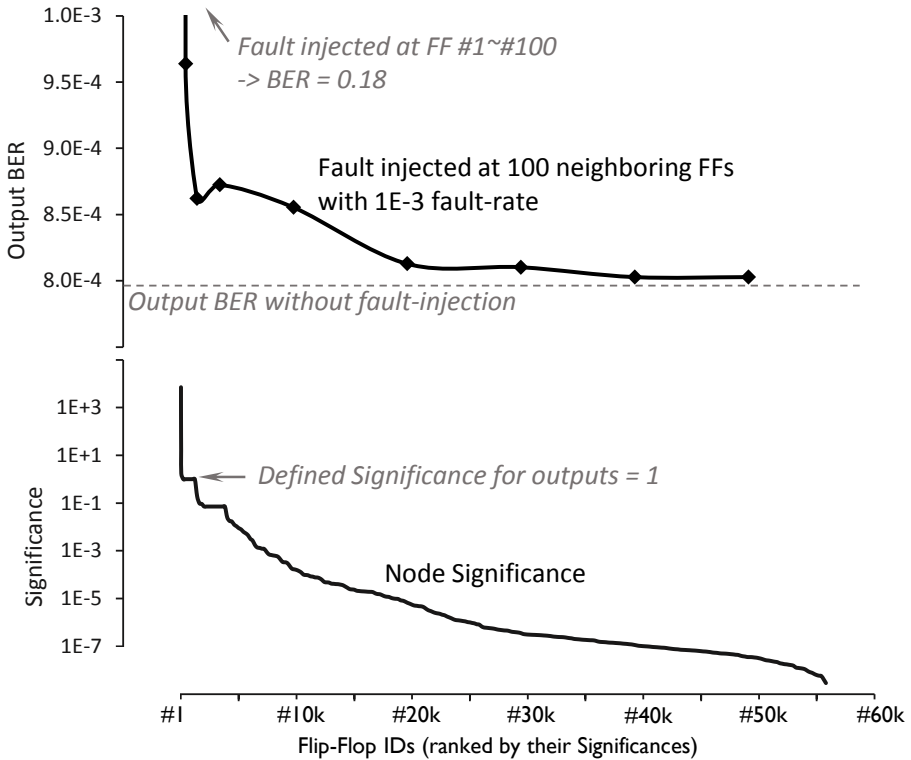


Figure 3.7: On an LDPC decoder circuit, errors on FF with high significance values leads to worse BER. The output quality is measured as BER, when random soft-errors are injected into FF. At each run, 100 neighboring FF with similar significance are under soft-errors, while the rest are free from soft-errors.

For comparison, a Monte-Carlo method is applied to simulate the LDPC under soft-error (randomly fault injection). The simulator randomly flips the Q port for every selected FF at a chance of 0.1% (1 in every thousand cycles) for each FF. For each run (denoted as a circle in Fig. 3.7), 100 FF with similar significance (e.g. FF_{1000} to FF_{1099} , or FF_{2000} to FF_{2099}) are selected for fault injection. Each chosen flip-flops is flipped during a randomly-chosen clock cycle with a uniform fault injection rate of 0.1% (1 in every thousand cycles). The output decoded bit error rate (BER) is denoted as the system's output quality.

If faults are injected to FF with significance ranking smaller than 300 (high significance), the output degrades significantly (BER = 0.18). This is because once errors are introduced in these most significant nodes, a huge amount of FF are affected, and the output results are gradually degraded. The significance of most FF are smaller than the defined output significance (defined as 1), suggesting that they are less sensitive to errors. This reflects the error absorbent capability of the LDPC circuit. Fig. 3.7 shows a good coherence between the rank of the significance and the severity of soft-errors on the corresponding FF.

Note that the BER curve is not monotonously decreasing around the Flip-Flop ID # 3k. This shows that the ranking cannot guarantee a node with higher significance is more important than others. The inaccuracy can be because of a lack of statistical and run-time logic state information, and mismatches in the significance distribution, etc. Despite that, the ranking serves as a guideline to distinguish which nodes are more important that should be protected, from the statistical point of view, as shown in Fig. 3.7.

3.6 Application to harden an FFT design

With the help of SERIAL, selective hardening can be performed on the FF with the highest significances. The FFT64 is simulated under random errors on FF, to capture its output SNR (Fig. 3.8).² It easily represents the scenario when chips are enduring soft-error when the error generation is uniform across all FF. It can also mimic the error of VOS, with some extra modeling of the error generation.

An error rate of 10^{-3} and 10^{-4} is assumed for all basic FF, except for the hardened FF, where no error is assumed. A logical method to increase the reliability of a system under this threat is selective FF hardening, until the point that the reliability (in this case output quality) demand is met.

²In Chapter 6, the output SNR is named as the Signal-to-Digital-Distortion Ratio (SDDR), to avoid the confusion with the term SNR in wireless communication society to quantify the quality of channel.

Comparing with the hardening without the help of SERIAL, where randomly selecting FF for protection is the viable option, hardening with the guidance of SERIAL yields much better SNR even at a lower hardening coverage (smaller overhead). For instance, if the target SNR is 20dB under the fault rate of 10^{-4} , only 45% FF are required for hardening, compared with the almost 100% hardening need for the random selection approach. Considering that hardening an FF doubles the FF area, the selective hardening saves more than half of the hardening overhead.

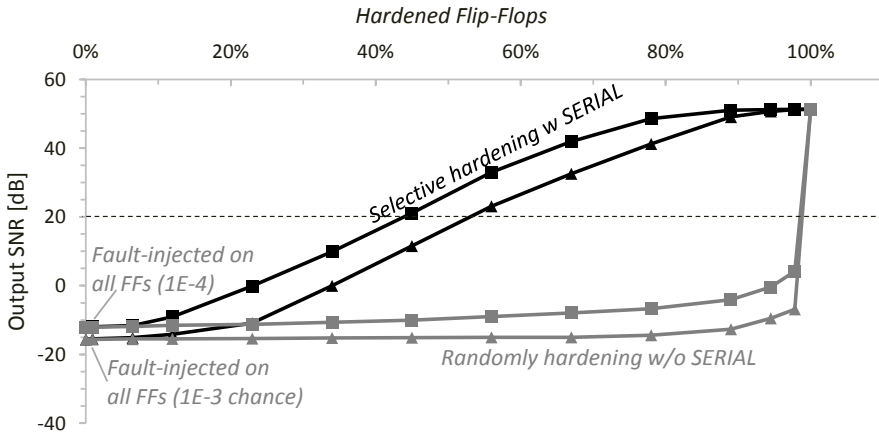


Figure 3.8: Compared with hardening randomly, selective hardening with SERIAL increases the SNR, at the same hardening ratio.

That proves that for the same reliability/quality target, the FFT that hardened with the help of SERIAL can be superior in power consumption and area cost.

3.7 Conclusion

In this chapter, an analytical approach to estimate the effect of error occurrence at gate-level is presented. The main novel idea in the proposed method is to pursue a heuristic approach to find out the controlling logics that is too important to fail. This serves as a guidance for circuit designers to selectively harden the design by methods such as FF hardening and design margin insertion, and consequently reduce the power overhead of error-hardening.

The solving algorithm has shown a low execution runtime and good scalability of our method on various benchmark designs, i.e. ISCAS'89 and ITC'99 circuits,

an FFT processor and an LDPC decoder. The coherence of the significance and the error effect, as well as the selective hardening benefits, was verified. Using this model, an FFT processor is designed. It shows huge gains regarding output quality with the same error generation rate (power overhead).

This work can further extend to analysis the impact when input data and circuit's states are not completely random. Besides, the workload dependent error characteristic can also be introduced into this model to address aging-related errors [[Hamdioui, 2013](#); [Reddy, 2005](#)].

Chapter 4

Microarchitecture-level power optimization exploiting application opportunities

This chapter performs microarchitecture-level power saving when considering the opportunities from applications. It proposes, in the context of general-purpose and domain-specific processors (embedded CPU, MCU or DSP), a fine-grained hardware switching scheme to select the proper device for low power computing. This scheme exploits word-length optimization opportunities for a multiplication unit. This scheme reduces the power of the multiplication unit from an OpenRISC processor by 23.7%, which is equivalent to a 9.5% power saving for the whole execution unit. The work in this chapter is published in [Huang, 2016c].

The rest of this chapter is structured as follows: Section 4.1 pinpoints the opportunity in the power-hungry multiplier for an embedded processor. Section 4.3 reviewed the power consumption waste issue in an OpenRISC processor. Section 4.4 proposes the fine-grained hardware switch scheme and applied it to the multiplier unit. Section 4.5 profiles a variety of benchmark algorithms to find out the possible power benefit of this scheme. Section 4.6 verifies the benefits in energy saving by netlist simulation. Section 4.7 concludes

the chapter.

4.1 Processor power waste in using unsuitable execution units

Energy consumption is one of the most critical metrics for embedded signal processing systems. Traditionally, designers optimize the fixed-point word length that provides just-necessary precision for minimizing the power consumption. On the other hand, driven by the increasing demand for computing re-programmability, general-purpose computing devices, e.g. DSP, ASIP, and application processors, are becoming more favorable.

In these systems, designers are constrained to perform the arbitrary word-length optimization, since processors typically sacrifice hardware costs to cater for the most complicated computing cases. In the meantime, a lot of lightweight computations that can be performed in low-energy operation devices, are executed on these over-complicated and power-hungry hardware, which leads to energy waste consequently.

In general-purpose processors, subword SIMD exploits the over-reserved word-length by applying parallelism in data-path processing. Employing SIMD reduces the number of operations, and hence decreases energy consumption. However, It requires dedicated hardware as well as software tuning to enable these SIMD intrinsic functions. Subword soft-SIMD [Kraemer, 2007], on the other hand, relies purely on software to exploit the sub-word parallelism. Nevertheless, in this scheme, guard bits are needed to be inserted, which is non-trivial for software developers [Novo, 2010; Catthoor, 2010].

4.2 Contribution of this chapter

This chapter introduces an alternative low precision computation unit besides the traditional full precision unit. A hardware word-length detector is used to switch the hardware units, in a fine-grained manner, to reduce the computational cost when the full precision computation is not necessary.

Without degrading the output quality, this work detects small word-length computations and executes them in an extra reduced precision unit. This mechanism radically reduces the activation chance of the full precision unit. As a result, the dynamic power consumption decreases notably.

The detection and execution units are both implemented with hardware at the microarchitecture level. Consequently, this technique requires neither modification on compiler nor software. The mechanism is applied to the multiplication unit. An alternative low-precision multiplier is therefore proposed. It leads to significant power saving, as the power consumption of a multiplier is $O(n^2)$ regarding the word-length n .

This proposal is applied to OpenRISC, an open-source embedded microprocessor. This work implements the multiplier with proposed fine-grained hardware switch scheme. It verifies the energy improvement with algorithm profiling and gate-level simulation.

4.3 Targeted embedded processing platform

OpenRISC [Lampret, 2014], RISC-V [Waterman, 2011], Sparc [SPARC, 1994] are among the most famous open-source computing platforms. They are suitable to study power savings at the microarchitecture level. Without losing generality, this chapter adopts a simple 32-bit OpenRISC processor, called mor1kx (Cappuccino implementation) [Kristiansson, 2013].

4.3.1 OpenRISC microarchitecture

The schematic of the processor is shown in Fig. 4.1. The clock frequency is 1 GHz. The processor is realized in a standard 28nm CMOS technology. The execution stage consists of an ALU (Arithmetic Logic Unit), a logic computation unit, a Load/Store unit, a serial divider and a 4-cycle multiplier.

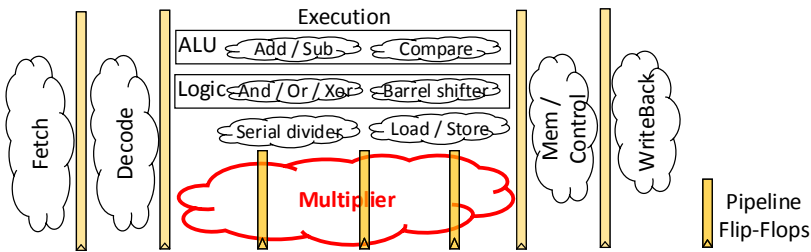


Figure 4.1: A customized OpenRISC microarchitecture: the mor1kx Cappuccino flavor. It consists 5 pipeline stages. The 4-stage multiplier resides in the execution unit.

4.3.2 Power of the multiplication unit

The circuit diagram of the multiplication unit (*MUL*) of the default implementation is shown in Fig. 4.2. The *MUL* contains four pipeline stages with clock-gating. Clock-gating helps to save energy, as it avoids signals from toggling when the *MUL* is not in operation, for instance, when the processor is performing irrelevant instructions.

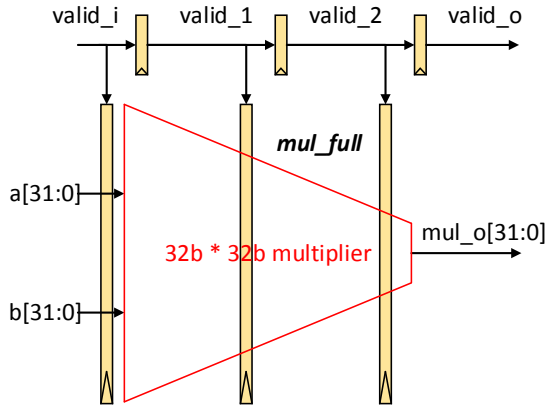


Figure 4.2: The original multiplier in the Cappuccino implementation is 32 bits wide.

The area utilization and power consumption of the multiplier unit for each instruction are profiled in Table 4.1. For the 32-bit multiplier, even if the word-length of multiplicands is much shorter than 32-bit, the power consumption is comparable with full-precision multiplication. The reason lies in the fact that multiplicands are represented in 2’s complement form. In this form, the most significant bits (MSB) are filled with ‘1’s or ‘0’s when the number is short, which results in lots of toggling during positive-to-negative or negative-to-positive transitions. Nevertheless, if proper multiplier units are used, e.g. 8-bit multipliers for 8-bit multiplicand, the power consumption will be reduced accordingly.

Table 4.1: Multiplier area and power consumption during each instruction.

Multiplier size	Cell area [μmm^2]	Power during instructions [μW]				
		NOP	4-bit MUL	8-bit MUL	16-bit MUL	32-bit MUL
4-bit	109	12.152	48.022	N/A	N/A	N/A
8-bit	289	16.497	100.417	111.18	N/A	N/A
16-bit	1030	49.511	234.2	278.934	349.657	N/A
32-bit	1744	50.737	391.350	451.808	531.32	567.924

This phenomenon provides an excellent opportunity for power optimization in processors, as the multiplicands type are not always declared as the full-size 32-bit long integer. Moreover, even if they are declared as 32-bit long integer, the actual value can be small, e.g. between -128 and 127 (which can be represented by an 8-bit number).

4.4 Fine-grained hardware-switch scheme for multiplier (*MUL*)

Considering that there is huge power waste because of unnecessary gate toggling, this chapter introduces an alternative lower-cost multiplier to perform the computation for the cases when the word-length of multiplicands is short enough (see Fig. 4.3).

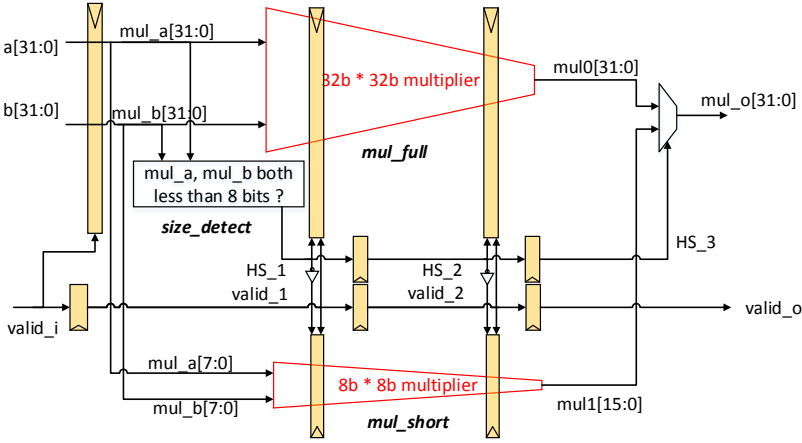


Figure 4.3: *MUL* with the proposed hardware-switching (HS) scheme selects between a 32-bit and an 8-bit multiplier, based on the data range.

A simple size detecting unit (*size_detect*) is deployed to detect if both multiplicands are small, by checking if the MSBs (in this example from 8-bit on) is the same (all ‘1’s or ‘0’s). If both multiplicands are short, *mul_short* will execute the operation while the *mul_full* is clock-gated, and vice versa. This hardware-switching (HS) scheme ensures that the signal only toggles in the proper multiplier unit, and the toggling in the other multiplier unit is minimized. The multipliers are divided into four stages by three sets of pipeline registers. Signals in the first stage always toggle even if the unit is not enabled since the logic inputs of the first stage are not clock-gated by the *size_detect*.

The multiplier is retimed using a commercial RTL synthesizing compiler, which minimizes the power cost of the first pipeline stage. It moves more computations into the second stage, reducing the gate toggling incurred power consumption during irrelevant operations in the first stage.

The cell area of the HS multiplier is $2053 \mu\text{mm}^2$, which is 18% higher than the Original multiplier. The area overhead is due to the introduction of the short multiplier and the corresponding MUX circuit.

The power consumption of the *Original* and the *HS* multiplier is compared in Fig. 4.4. It is broken down into 3 parts: *mul_full*, *mul_short*, and *mul_rest* (rest parts in the multiplier). During NOPs, both multipliers consume less than $40 \mu\text{W}$, which mainly attributes to clock gating cells. For the HS multiplier, if the multiplicand is shorter than 8 bits, the *mul_full* unit is clock-gated, and the processing is assigned to the low-power *mul_short*. Therefore, the overall power

consumption is significantly lower than the original multiplier. This advantage diminishes when all the multiplicands are larger than 8 bits. In that situation, the *HS* multiplier suffers from the power penalty of the *size_detect* and the MUX unit.

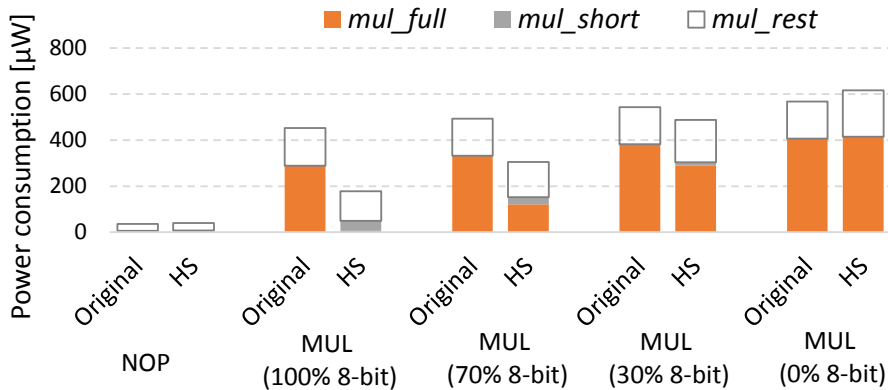


Figure 4.4: *HS* saves power when the multiplication inputs are small. During nops, the difference is marginal. The multiplicands are randomly generated to be either 8-bit or 32-bit, with the accordingly possibility.

4.5 Algorithm profile

The power savings only happen when using the low-precision multiplier. In order to measure the power consumption benefits of the *HS* multiplier, it is important to track the utilization frequency of the multiplication operation ($\#multiplication/\#instructions$), and the statistical chances that both multiplicands are short. These statistics depend heavily on application algorithms and the input data.

For the benchmark, *Cormark* 1.0 [EEMBC, 2006] and ten other common algorithms for software radio and multimedia processing are profiled. *Cormark* focuses on benchmarking CPU cores of embedded systems. The selected algorithms cover a broad range of typical applications in embedded processing, e.g. FFT, filtering, JPG decoding, cryptography, and error correction. The

input data are set to represent the typical usage scenario.

4.5.1 Utilization frequency of *MUL*

Fig. 4.5 shows the utilization frequency of the multiplier. In average, 1.2% of the instructions is a multiplication. The processor actually takes more than 1.2% of execution time in multiplication, as each multiplication takes four cycles.

Since each multiplication takes four cycles, the processor will take around 4.8% of the cycles for multiplications.

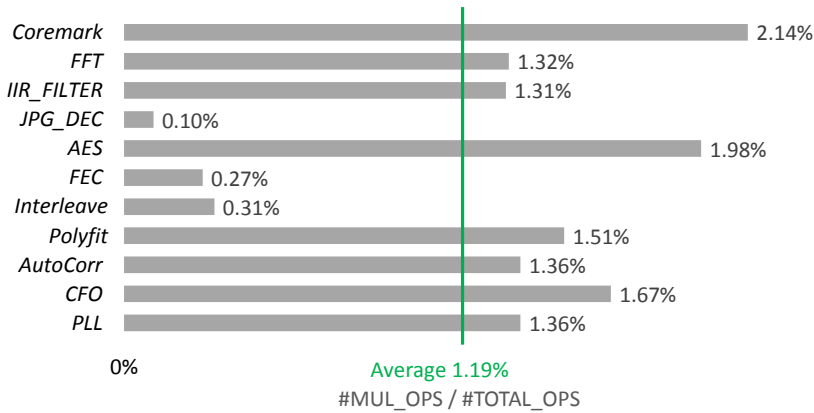


Figure 4.5: The utilization frequency of the multiplier unit is around 1.2%. So optimization on the multiplication unit is profitable.

4.5.2 Word-length distribution for *MUL*

The word-length distribution of the multiplicands is illustrated in Fig. 4.6. The data is obtained by the cycle-accurate OpenRISC simulator. The multiplicands are recorded for each multiplication. There is a trade-off to choose how large the *mul_short* should be. If the criterion for short input is more strict, i.e. # of bits is larger, the activation chance of the *mul_short* unit will increase, which leads to a lower power consequently. On the other hand, a larger *mul_short* unit itself consumes more power. Therefore, designers are suggested to profile the multiplication size coverage for typical applications and the corresponding power consumptions.

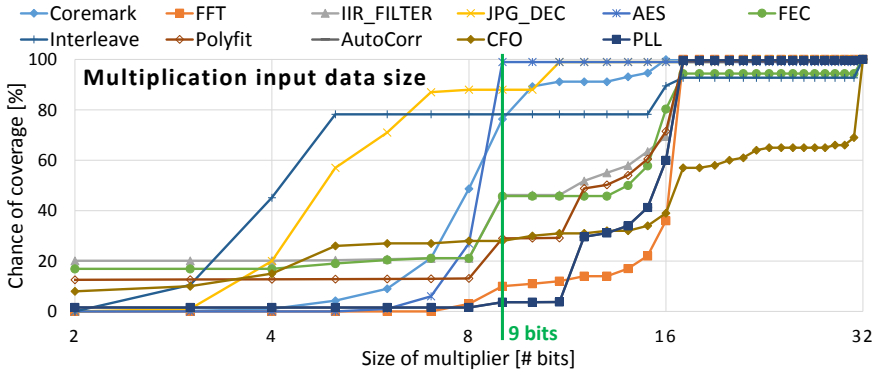


Figure 4.6: The chance that both multiplicands are short increases when the criteria for ‘short’ gets loose.

Based on algorithm-level benchmark results (Fig. 4.6), a 9-bit multiplier is used for the *mul_short*. With this setting, the *mul_short* unit performs more than 80% of the multiplication for *Coremark*, *JPG_DEC*, *AES*, and *Interleave*; around 40% multiplication for *IIR_FILTER*, *FEC*, *Polyfit*, and *CFO*; around 5% for *FFT* and *PLL*.

This result shows that the HS scheme best fits algorithms that heavily use *short_integer* data-types for multiplications. In this scenario, the *size_detect* takes the role of the compiler to choose the suitable multiplication hardware. Moreover, for algorithms that use only full-width *integer* data-type, e.g. *IIR_FILTER* and *Polyfit*, the *textitmul_short* still performs around 40% of the multiplications. This is due to the fact that the varying input data has a very high tendency of falling into the short-size range, even though they are defined to be very wide to avoid the overflow in the worst case.

4.6 Power saving validation of the proposed fine-grained hardware switch scheme

The *mor1kx* is synthesized at 1GHz in a standard 28nm CMOS process. The derived netlist, together with its corresponding delay file (.sdf) and parasitic parameters, are simulated with the instructions from the most realistic and

representative stimuli – *Cormark*.

4.6.1 Area comparison

The area and power metrics with *Original* or *HS* (with 9-bit *mul_short*) schemes are shown in Fig. 4.7 and Fig. 4.8. For the processor with HS scheme, the extra *mul_short* and *size_detect* results in 23.0% area overhead for the multiplier unit, which is equivalent to 11.5% area overhead for the whole execution unit.

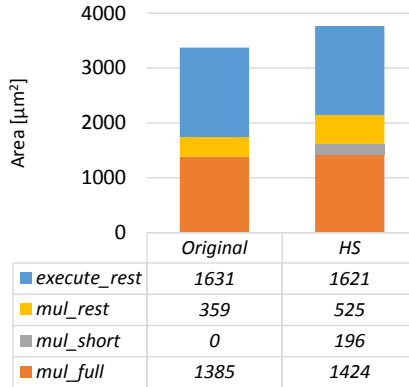


Figure 4.7: The proposed *HS* scheme results to more area.

4.6.2 Power savings

The power consumption of the *mul_full* is reduced from 31.167 μW to 12.344 μW , since its execution ratio is greatly reduced. It accounts for a total of 23.7% power saving for the multiplier unit and 9.5% power saving for the execution unit. In summary, the fine-grained hardware switch scheme introduces redundant area, which saves execution power for low-precision computations.

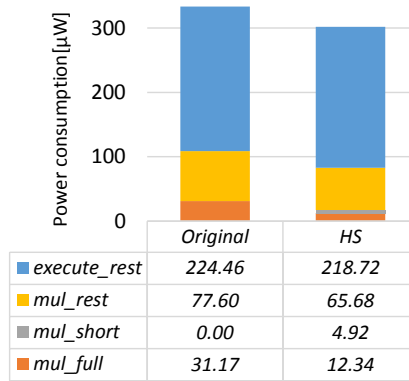


Figure 4.8: The power consumption of the *HS* reduces substantially. The results are based on gate-level simulation with the Coremark benchmark.

4.7 Conclusion

The *HS* scheme proposed in this chapter exploits word-length opportunities to reduce dynamic power consumption. It is achieved by utilizing an alternative short multiplier when the circuit detects the inputs are short enough.

The proposed scheme does not affect the software nor the compiler since the detection and switching are implemented at hardware level. It best fits processors which frequently perform short multiplications. In such processors, the multiplier unit power is significantly reduced.

In this chapter, the hardware switch scheme does not alter the final output of the program, since the opportunistic hardware switch is only enabled if the input size is small enough to fit into the low-precision unit. In the future, for programs with soft quality requirements, the activation of the low-precision unit can be extended to scenarios that the data size is marginally larger than the low-precision unit. Therefore, the occurrence of low-precision activation is increased, and hence more power saving could be achieved, though at the cost of degraded output quality.

In this regards, more cross-level optimization between the microarchitecture and the algorithm-level is encouraged. One such example of circuit and algorithm interplay is explained in Chapter 5.

Chapter 5

Algorithm-level error-resilient design mitigating circuit-level errors

This chapter proposes a method for cross-level error interplay between the circuit-level and the algorithm-level. Traditionally, circuit-level results need to be error-free. This confines the variability design margin shaving methods to error-free approaches, e.g. Canary FF. In contrast, this chapter promotes to embrace some circuit errors, for a more aggressive margin shaving. Errors produced at the circuit-level are mitigated at the algorithm-level. In particular, a *computation-skip* scheme is discussed. The error mitigation scheme, together with the Canary FF that serves for benchmark purposes, is applied to a hardware CORDIC accelerator. The typical applications for CORDIC are QR decomposition and Cartesian to polar coordinate vector translation. The CORDIC accelerator is processed and verified in a standard 28nm CMOS process with only standard-cells. Using only standard-cells, this work eliminates the traditional semi- (or even fully-) customized design effort for in-situ error detection circuits. The work in this chapter is published in [Huang, 2014; Huang, 2016d].

The rest of this chapter is structured as follows: Section 5.1 describes the background and the contributions. Section 5.3 explains the design implementation trade-offs of the error-resilient circuits. It pinpoints the advantages of the proposed *computation-skip* scheme. Section 5.4 takes a

CORDIC hardware accelerator for case-study. Section 5.5 concludes the chapter.

5.1 Motivatuion to algorithm-level error resilience

Integrated circuits are designed with inherent guardbands, i.e. using worst PVT (Process, supply Voltage noise V_{dd} and Temperature) corners, to ensure correct functionality for all chips with the presence of dynamic temperature and V_{dd} noise fluctuation. However, typical usage patterns usually operate at the nominal PVT condition. The over-pessimistic margin leads to power waste. Nowadays, digital circuits aiming for low-energy consumption usually adjust the V_{dd} (V_{dd} scaling) to exploit the design margin. The energy consumption will reduce quadratically to the V_{dd} . Nevertheless, the circuit setup-timing constraints must be met, otherwise data errors will occur.

5.1.1 Review of error-resilient techniques

In-situ timing-error detection circuits have been proposed to detect setup timing-errors when reducing V_{dd} [Ernst, 2003; Calhoun, 2004; Bowman, 2009b; Nicolaidis, 2013; Fojtik, 2013]. A canary FF [Calhoun, 2004] generates a warning when the timing is critical. This enables dynamically adjusting the V_{dd} and/or f_{clk} for microprocessors. A circuit with canary FFs gradually reduces its V_{dd} , at the training phase, until the first occurrence of warning. This ensures correct functionality during operating, as no errors, but only warnings, occur. This scheme is regarded conservative, as a delay margin (between warnings and actual timing-errors) still exists. Razor like techniques, e.g. Razor [Ernst, 2003], DSTB & TDTB [Bowman, 2009b], and Bubble Razor [Fojtik, 2013], exploit the margin even further, by detecting actual timing-errors. This condition is called VOS (Voltage-OverScaling), as the voltage is scaled beyond the safety region. The corresponding error correction schemes, i.e. global clock gating [Ernst, 2003], counter-flow [Ernst, 2003], instruction-replay [Bowman, 2009b], and Bubble Razor [Fojtik, 2013], correct errors by issuing extra instructions. Therefore, the circuit can operate around the operating condition that produces sparse errors. These schemes, although applicable to general-purpose computers, result in throughput penalty when timing-errors are detected. Another common drawback of the previously proposed works in this class is the utilization of customized circuit design methodologies (mostly for error detection circuits), which is not a classical digital design flow.

TIMBER [Choudhury, 2010; Constantin, 2015] delays the clock for 1-phase to compensate the time borrowing for a timing-error at the circuit-level. However,

this technique requires substantial effort in adjusting the clock phase for error recovery. This is very challenging for the timing closure with commercial EDA tools [Tam, 2000; Bowman, 2016].

Another class of methods to handle VOS errors is arithmetic noise tolerance (ANT) technique [Shim, 2004; Hegde, 2004; Karakonstantis, 2009; Karakonstantis, 2012a; Whatmough, 2014]. ANT techniques detect errors by algorithmic comparison, and correct them without extra cycle penalty, by linear prediction [Hegde, 2004], reduced precision redundancy [Shim, 2004] or adaptive error cancellation [Wang, 2003]. However, a major drawback of regular ANT techniques is the non-generic algorithmic error detection. They require careful ad-hoc design. Another common problem of most ANT schemes is the requirements of dedicatedly selected data-path, e.g. specific adder micro-architectures [Whatmough, 2014]. This prevents data-path synthesis optimization that modern EDA offers.

5.2 Contributions of this chapter

A common drawback of the previously proposed work is the utilization of customized circuit design methodologies (mostly for error detection circuits), which is not a classical digital design flow. Besides, although extensive measurements are performed for Razor-like error recovery circuits, seldom applications address the relation between V_{dd} drop and output quality for normal DSP blocks [Karakonstantis, 2012a; Whatmough, 2014]. Furthermore, the design margin reduction effectiveness is not verified with deeply scaled sub-28nm technologies.

In contrast, this work proposes a power reduction method (named as *computation-skip* scheme) for a DSP accelerator. It demonstrates the power reduction and implementation feasibility of the mentioned techniques in deeply scaled technologies. The *computation-skip* scheme handles errors at the algorithm-level that were created at the circuit-level. It mitigates the timing-error during V_{dd} scaling for recursive application, at the cost of output SNR degradation without throughput drop. The proposed *computation-skip* scheme can be applied to signal processing algorithms with a recursive structure. In these algorithms, signals will be processed by the same combinational logic for multiple times. Examples are CORDIC, Viterbi, LDPC decoding, loop counter and genetic algorithms. The *computation-skip* scheme is compared with existing in-situ error correction schemes in Table 5.1.

The major advantage of this scheme, compared with other FF based techniques and the temporal clock adjusting schemes, is the 0-cycle overhead during error

Table 5.1: In-situ timing-error detection and correction/mitigation schemes.

Features	FP based techniques	Temporal clock adjust	Algorithmic approximation	<i>computation-skip</i> Error mitigation
Examples	Razor [Borwman, 2000b]; DSTB and DDTB [Borwman, 2009b]; Bubble Razor [Fojtik, 2013]; One-cycle [Shin, 2013]	TIMBER [Choudhury, 2010]	ANT [Hogde, 1999a; Abdallah, 2009]	proposed <i>computation-skip</i> [Huang, 2016d]
Error Detection	double sampling transition detection	double sampling transition detection	algorithmic inference	double sampling transition detection
Accurate detection?	yes	yes	no (inference)	yes
When detector samples?	after clock edge	after clock edge	at clock edge	after clock edge
Error correction	counter-flow instruction replay bubble insertion	delay clock for one phase	use approximated Q	partial computation-skip in the next cycle
Guaranteed quality?	yes	yes	no (error accumulated)	no (error not accumulated)
Guaranteed throughput?	no (1+ cycles penalty)	no (1 clock phase penalty)	yes	yes
Clock overhead	moderate	heavy	none	none
Logical gate overhead	small	small	moderate	small
Time borrowing?	yes	yes	no	yes
Application independent?	yes	yes	no	no (recursive applications only)

correction. On the other hand, comparing with other ANT techniques, the *computation-skip* scheme simplifies the error-detection design by adopting the FF based approach.

From the implementation point of view, the state-of-the-art canary FF error detection scheme, as well as the newly proposed *computation-skip* scheme, were implemented and verified on silicon. Both circuits were processed in 28nm CMOS with standard digital design flow. This eliminates the conventional semi- (or even fully-) customized design effort for in-situ error detection circuits.

5.3 Error resilient circuits implementations

The circuit-level implementations of error resilient circuits are described in this section. This section describes the concept of Canary FF circuits. Moreover, it explains the design trade-offs of the computation-skip scheme.

A conventional pipeline circuit diagram is outlined in Fig. 5.1. The signal arrival time for the *D* port of an MSFF (Master-Slave Flip-Flop) is constrained by i) its hold-time plus the fast corner timing variation guard-band, and ii) its setup time plus the slow corner PVT guard-band. These corner-based guard-bands limit the capability of power saving.

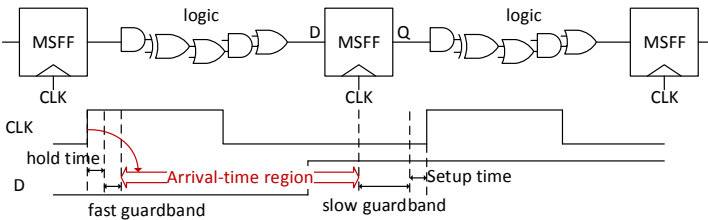


Figure 5.1: The conventional pipeline scheme requires worst-case design margins.

5.3.1 Circuits with Canary FF

An in-situ *canary FF* based circuit applies a second *shadow MSFF* to detect dangerous (slow) timing at critical V_{dd} or f_{clk} (Fig. 5.2). When the signal arrival on *D* is critical, its delayed signal *D'* will violate its constraint. As a consequence, the *main MSFF* and the *shadow MSFF* capture different values. This triggers a local warning (W_{local}) since the *shadow MSFF* has failed. Because the *main MSFF* does not fail, the situation is only reported as a warning, rather than

an error. This critical operation condition is hence called the PoFW (Point of the First Warning). However, if signal delays are enlarged under worse conditions, *main MSFF* will fail, and functionality errors will occur. The operation condition that the first error emerges is called the PoFF (Point of the First Failure).

The width between the PoFW and the PoFF is the error detection window, which is tuned by the delay element. The circuit usually operates around the PoFW condition. The error detection window in *canary FF* allows infrequent warnings and also ensures correct circuit functionality. The PVT slow guardband is exploited as each chip operates with just needed energy.

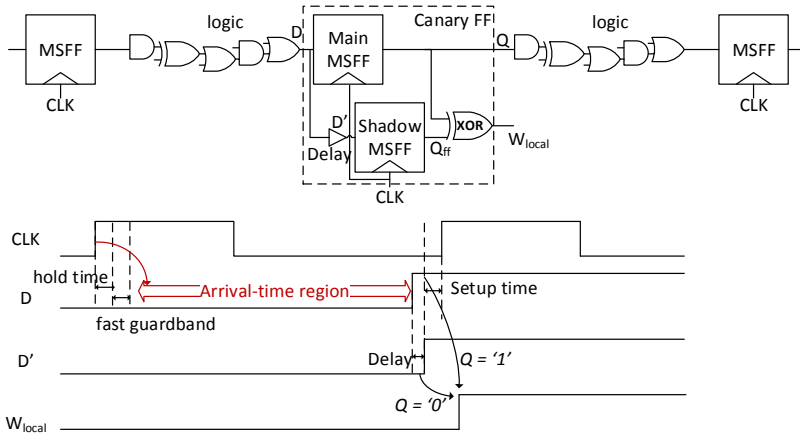


Figure 5.2: Circuits with in-situ *canary FF* that shave worst-case design margins.

5.3.2 Circuits with *computation-skip* scheme

The *computation-skip* error resilient scheme [Huang, 2016d] utilizes DSTB (Double Sampling with Time Borrowing) [Bowman, 2009b] for error detection (Fig. 5.3). A DSTB (Fig. 5.3) consists of an enable-high *latch*, an *shadow MSFF*, and an *XOR comparator*.

If a signal arrives late than the clock rising edge, the latch captures the correct signal (with time borrowing) while the shadow MSFF captures an incorrect one, an error flag E_{local} is produced. The operating condition that the first error emerges is PoFF. If the chip only operates around the PoFF, the circuit behaves similarly to *canary FF* based circuits. However, by investing the signal processing algorithms, a further design margin can be exploited. [Huang, 2016d] proposed that for recursive applications, e.g. CORDIC and Viterbi, a part

of computations can be approximated, or even totally skipped, under the worst-case condition. The final output is slightly degraded, which can often be tolerated or even compensated by the upper layer of the system, e.g. by using more quantization bits or stronger ECC. The system can thus decide whether to continue to work with the degraded circuit or to improve the operating condition (reducing f_{clk} and increasing V_{dd}) to prevent further degradations.

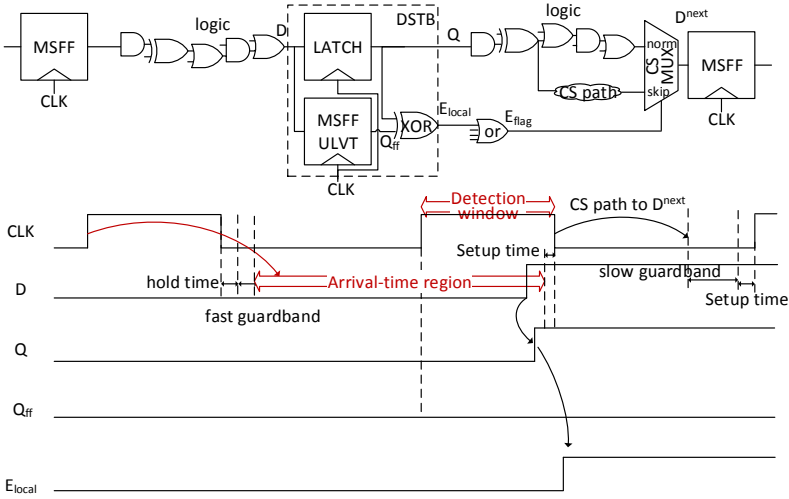


Figure 5.3: Circuits with *computation-skip* scheme, which utilizes *DTSBs* for error-detection, eliminate worst-case design margins.

Fig. 5.3 shows the timing diagram of a circuit with *computation-skip* scheme. Comparing with *canary FF*, it eliminates not only the PVT slow guard-band but also the delay margin between PoFW and PoFF. More importantly, a path with long propagation delay can borrow time from the next upcoming clock cycle, relaxing its timing constraint by a duty cycle of the clock ($\tau \cdot t_{clk}$). To compensate for the time borrowing, a short version of the upcoming computational path, i.e. a *computation-skip* path, is selected to meet the constraint. However, a disadvantage for circuits using DSTB or Razor is the minimum delay requirement. This requirement overburdens the hold time fixing during layout. Unfortunately, this disadvantage is also present in this *computation-skip* scheme. Bubble Razor [Fojtik, 2013] eliminates the minimum delay issue by adopting a two-phase latch design. However, these two-phase latches are still not fully compatible to mainstream EDA tools.

Once a timing violation is detected by the DSTB, Q is fed to the MUX by the *computation-skip* path. This is because re-computing the next logic with

the late-arrived correct Q is impossible, due to the setup timing constraint for the following cycle. As a consequence, the correct signals from the previous clock are preserved. More importantly, further timing violation is eliminated by bypassing with the *computation-skip* path. Note that only part of the *logic* is skipped by the *computation-skip* path. This avoids heavy quality degradations of skipping the whole logic.

This mitigation can be regarded as a naive implementation for the approximated version of the logic. For recursive applications, the bypassing can be as simple as a direct copy of the previous signal. Another benefit of the bypassing is that no accumulating approximation errors are introduced. The skipping only leads to less performed iterations. The skipped computations are delayed into future cycles. If extra cycles are allowed in the future, errors can be totally eliminated. However, this work aims to maintain a constant throughput. No time penalties are permitted. Therefore, the *computation-skip* error mitigation scheme alleviates timing-errors into insufficient iteration errors.

5.3.3 *Computation-skip* scheme settings

The *computation-skip* scheme tolerates longer propagation delay in data-path than a nominal circuit. In other words, the circuit can even operate at V_{dd} that lead to setup timing violations, or sub-critical V_{dd} , to save energy.

The detailed timing constraint is demonstrated in Fig. 5.4. By tuning the clock duty cycle factor τ , digital circuit designers can extend the nominal delay constraint $t_{max_original}$ to the error mitigation timing constraint t_{max_em} :

$$\begin{aligned} t_{max_ori} &= t_{clk} - t_{setup_FF} \\ t_{max_em} &= t_{clk} + \tau \cdot t_{clk} - t_{setup_latch} \end{aligned} \quad (5.1)$$

where t_{clk} is the clock duration, t_{setup_FF} represents the setup time for a normal FF, and t_{setup_latch} represents the setup time for the main latch. During logic synthesis, the relaxed timing constraint will enable EDA tool to use smaller but slower gates, which reduces area and power consumption.

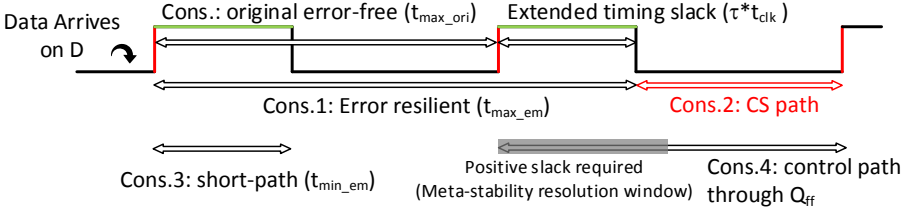


Figure 5.4: Timing constraints for circuits with the *computation-skip* scheme. The scheme relaxes the setup constraint by accepting late arrival signals.

The speed constraint relaxation ratio R is defined as:

$$R \triangleq \frac{1/t_{max_em}}{1/t_{max_ori}}. \tag{5.2}$$

Substituting t_{max_em} and t_{max_ori} from (5.1) obtains

$$R = \frac{t_{clk} - t_{setup_FF}}{t_{clk} + \tau \cdot t_{clk} - t_{setup_latch}} \approx \frac{1}{1 + \tau}. \tag{5.3}$$

A lower R represents loose constraint for data, and hence easier for setup timing closure.

At sub-critical V_{dd} situations, circuits delay increase that the data input port of the shadow MSFF might change around clock rising edges. So these FF might become meta-stable. Therefore, paths starting from Q_{ff} are guarded with positive slacks, serving as the resolution window for the FF to settle.

To find out the meta-stability resolution constant, Monte-Carlo simulation by Spice is performed on the ULVT FF. The meta-stability resolution is 20ps at nominal voltage. To mitigate the meta-stability issue, this chapter set an extra timing slack of 700ps. The worst case for MTBF estimation happens when exactly one signal at the D port of the FF changed close to the clock rising edge within setup time. Therefore,

$$MTBF = \frac{e^{\frac{700\ ps}{20\ ps}}}{f_{clk}} \approx 41\ days, \quad when\ f_{clk} = 450\ MHz. \tag{5.4}$$

It is sufficiently large to guarantee mean time before failure (MTBF) requirement for the system due to meta-stability. Note that the main latch is designed to

never fail at even sub-critical situations, the data-path and *computation-skip* path are immune from meta-stability, which is a big advantage for the DSTB. In reality, the system MTBF is much better than 41 days because any other timing failure will set E_{flag} to '1', masking the effect of the local meta-stability fault.

As the *main latch* is still sensitive after the clock rising edge throughout the first half of the clock, if it captures the newly arrived signal too early, the signal from the previous cycle is flushed. In this situation, the error detection circuit might indicate a false error. Therefore, for the paths to the *main latch*, a short-path timing constraint is required:

$$t_{min_em} = \tau \cdot t_{clk} + t_{hold_latch}, \quad (5.5)$$

where t_{hold_latch} is the hold time for the main latch. This short path constraint is guaranteed by inserting buffers during placement & route.

The timing constraint is summarized in Tab. 5.2. With the *computation-skip* scheme, the normal setup constraint is much relaxed with the ratio of R . This makes the timing much easier to meet, and hence EDA tools have the option to choose smaller and slower cells to save chip area and power consumption.

Table 5.2: Timing constraints for the *computation-skip* error mitigation scheme.

Constraints	Path
error mitigation setup	$Q \rightarrow \text{Logic1} \rightarrow \text{Logic2} \rightarrow D$
<i>computation-skip</i> path setup	$Q \rightarrow \text{CS path} \rightarrow D$
Q_{ff} control path setup	$Q_{ff} \rightarrow D$
short-path hold	$Q \rightarrow D$
	Value
error mitigation setup	$\leq t_{clk} \cdot (1 + \tau) - t_{setup_FF}$
<i>computation-skip</i> path setup	$\leq t_{clk} \cdot (1 - \tau) - t_{setup_FF}$
Q_{ff} control path setup	$\leq t_{clk} - t_{meta_window} - t_{setup_FF}$
short-path hold	$\geq \tau \cdot t_{clk} + t_{hold_latch}$
	Remarks
error mitigation setup	The timing is relaxed for normal data-path
<i>computation-skip</i> path setup	Skip constraint for error recovery
Q_{ff} control path setup	Positive slack as meta-stability resolution
short-path hold	Short path constraint for latch

5.4 Case study on a CORDIC hardware accelerator

A CORDIC hardware accelerator is selected for a case-study to validate the error resilient design techniques. The canary FF case and the *computation-skip* are applied to the *core1 canary* and the *core2 CS*, respectively. These two cores are synthesized and processed in a standard 28nm CMOS technology.

5.4.1 CORDIC algorithm

A CORDIC [Volder, 1959] is a simple and efficient implementation to calculate trigonometric functions. A typical application is to compute the magnitude M and initial angle ϕ of a complex input vector $[x_0, y_0]$:

$$\begin{aligned}
 M &\triangleq \sqrt{x_0^2 + y_0^2}; \\
 \text{Cos}(\phi) &\triangleq x_0 / \sqrt{x_0^2 + y_0^2}; \\
 \text{Sin}(\phi) &\triangleq y_0 / \sqrt{x_0^2 + y_0^2}.
 \end{aligned}
 \tag{5.6}$$

The CORDIC operation is an iterative process: the input vector $[x_0, y_0]$ is rotated recursively by micro-rotations. The *CORDIC cell* that performs one micro-rotation (iteration) is shown in Fig. 5.5.

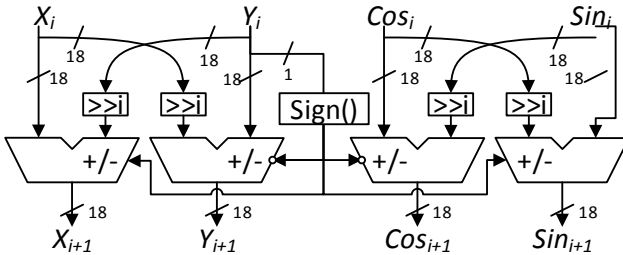


Figure 5.5: A *CORDIC cell* that performs a CORDIC iteration. The example word-length is 18-bit.

The total number of iterations for a CORDIC operation is denoted as n . After n iterations, y_i converges to 0, and the resulted x_n is the magnitude, with a scaling factor $K \approx 0.6072$. The angle (ϕ) is represented in the sine (Sin_n) and cosine (Cos_n) format.

Because of non-ideal in the CORDIC algorithm, the resulted x_n , Sin_n , and Cos_n can never reach the true values (M , $Cos(\phi)$, $Sin(\phi)$). The computation errors can be measured as the mean square error (MSE), the error vector magnitude (EVM), the signal-to-noise ratio (SNR), and the effective number of bits (ENOB). The computation procedures are given below.

The MSE directly depicts the magnitude of errors:

$$MSE = \frac{1}{T} \sum_{t=1}^T \left[(x_0 - x_{out})^2 + (y_0 - x_{out})^2 \right] \quad (5.7)$$

$$= \frac{1}{T} \sum_{t=1}^T \left[(x_0 - k \cdot x_n \cdot Cos_n)^2 + (y_0 - k \cdot x_n \cdot Sin_n)^2 \right], \quad (5.8)$$

where T is the number of samples for calculation. x_0 and y_0 are inputs for sample t . The x_{out} and y_{out} terms are the reverted format of CORDIC outputs (x_n , Sin_n , and Cos_n). The EVM compares the MSE with the signal power:

$$EVM(dB) = 10 \log_{10} \frac{MSE}{P_{signal}} \quad (5.9)$$

$$= 10 \log_{10} \frac{MSE}{\frac{1}{T} \sum_{t=1}^T [x_0^2 + y_0^2]}. \quad (5.10)$$

The computation error is treated as noise. So the output quality can also be represented by SNR:

$$SNR(dB) = 10 \log_{10} \frac{P_{signal}}{MSE} \quad (5.11)$$

$$= 10 \log_{10} \frac{\frac{1}{T} \sum_{t=1}^T [x_0^2 + y_0^2]}{MSE}. \quad (5.12)$$

The SNR and EVM are in inverse linear relation, as illustrated in [Shafik, 2006]. In this chapter, the output quality of the CORDIC unit is denoted as ENOB [Geerts, 2006]:

$$ENOB = \frac{SNR - 1.76dB}{6.02}. \quad (5.13)$$

Fig.5.6 shows that the ENOB for the CORDIC evolves with each iteration. This evolution characteristic provides space for trading off the iteration number during VOS. Thus, this chapter proposes to apply the proposed *computation-skip*

scheme on the CORDIC application. Once a timing violation is detected, the chip skips part of the computation in the next cycle and adjusts the iteration counter. Due to the requirements of constant CPI, the final iterations, which contribute less to the EVM, are skipped to ensure that previous computations are guaranteed even in sub-critical situations. This is equivalent to computing with reduced iterations.

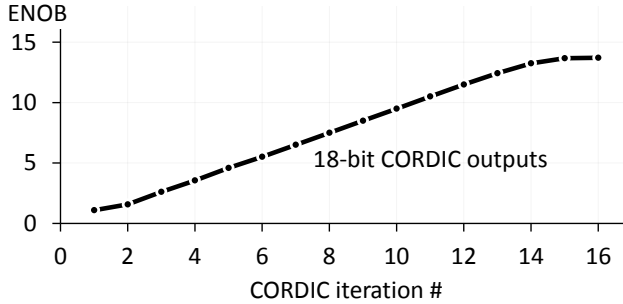


Figure 5.6: The CORDIC output ENOB evolves with numbers of iterations n .

5.4.2 State-of-the-art *Core1* with Canary FF

The first implementation option (*core1*) of the CORDIC accelerator is demonstrated in Fig. 5.7. The internal word size is 18-bit. 16 iterations are performed for each operation. The accelerator contains four *CORDIC cells*; each performs a CORDIC iteration. Therefore, four clocks cycles are required to finish a CORDIC operation (CPI=4). Computation results are stored in the sequential cells (e.g. FF), whose outputs serve as the inputs for the next cycle. The iteration counter counts the CORDIC iterations and controls the barrel shifters in the CORDIC cells. For each cycle, the iteration counter is counted up by 4, meaning that 4 CORDIC iterations are finished in each cycle. Aiming for a high-performance design specification, the operating frequency is 450 MHz by design. This requires intensive gate up-sizing, as the speed target is challenging if worst-case design margins are considered.

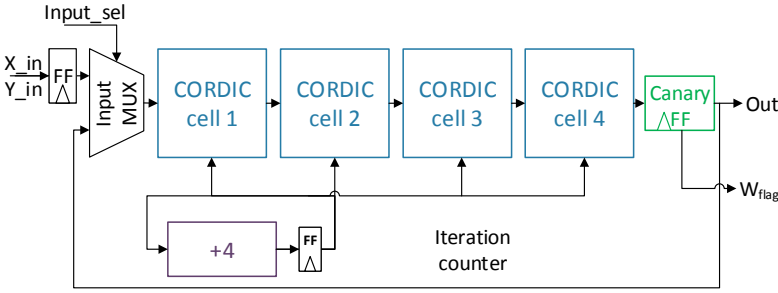


Figure 5.7: *Core1* hardware CORDIC accelerator replaces output FF with *Canary FF*.

The *canary FF* were built out of two MSFF, a *delay cell*, and a *XOR gate* (see Fig. 5.2). The delay of the *delay cell* is 150 ps, or 7% of the clock period, under nominal conditions. EDA is disallowed from changing the *canary FF*, otherwise the *delay cells* will be deleted (optimized) for timing relaxation (and thus timing-error detection capabilities are lost). Besides, only the timing constraints for paths to *main MSFF*, not for paths to *shadow MSFF* (disallowing timing checks), are guaranteed, to avoid over-constraining.

The design was implemented in standard RTL. Afterward, the most timing-critical FF are replaced by *canary FF*. Only the 9 MSBs from *CORDIC cell 4* are substituted since they are associated with the longest delays, as suggested by the EDA tool (STA checks performed with Primitime). This reduces the overall area overhead (of applying *canary FF*) to 1.3%, compared with the alternative 7% full substitute overhead.

5.4.3 Proposed *Core2* with *computation-skip* scheme

The second implementation option (*core2*) is presented in Fig. 5.8. Similar to the *core1*, FF that connect to *CORDIC cell 4* are replaced by *DSTB*.

Knowing that the output quality depends on the # of CORDIC iterations finished, the *computation-skip* approach introduced a skipping path that shorts the *CORDIC cell 3 and 4* when a short propagation delay is required. It is activated whenever the time is borrowed in the previous clock cycle (*DSTB* will set a timing-error flag). Once the skip path is activated, the iteration counter is counted up by 2 (instead of 4), to provide correct right-shift commands for future iterations. The 9 MSBs of the *CORDIC cell 4* were substituted by *DSTB* to reduce substitution overhead. As the remaining 9 LSBs will fail

under aggressive conditions, MSFF of these LSBs were replaced by latches to enable time borrowing. In summary, all outputs of *CORDIC cell 4* allow time borrowing and only the 9 MSBs are responsible for timing-error detection.

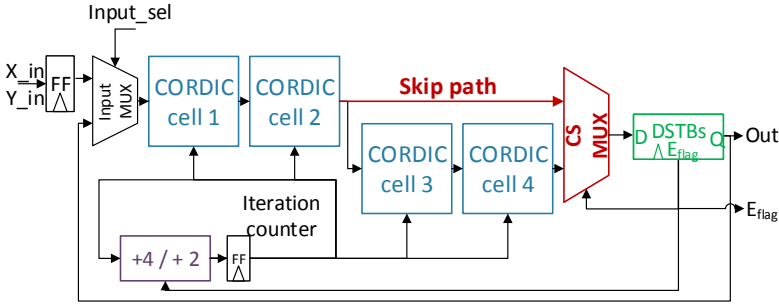


Figure 5.8: *Core2* features the proposed *computation-skip* scheme.

When a setup timing violation (for the previous cycle) is detected at the beginning of the current cycle, the circuit skips the last two CORDIC iterations (cell 3 and cell 4), due to insufficient processing time. Therefore, the iteration counter is counted up by two instead of by four. As a consequence, the intended four CORDIC iteration computation will eventually conduct two iterations. These skipped computations will be performed in later stages, as the iteration counter is only counted up by two. As a result, only the final computations are skipped because of the constraint of fixing the CPI to 4.

Fig.5.9 shows the timing diagram of the proposed CORDIC processor. When no error is detected, a complete CORDIC operation takes four cycles, computing 16 iterations. When one error is detected for cycle 2, the DSTB latches the late-arriving data and triggers the *computation-skip* path in cycle 3. After four cycles, only 14 iterations instead of 16 are performed. This result serves as a reduced-quality output to maintain the throughput. When the timing situation is very severe that all cycles fail, the performed iteration number is 12. This is because the *computation-skip* path (red in Fig.5.9) are constrained to never fail.

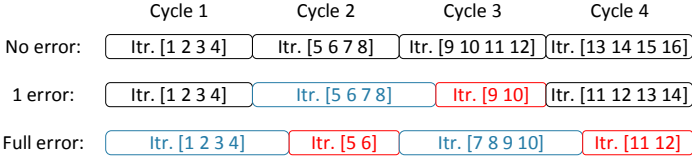


Figure 5.9: Example timing diagrams of the proposed *computation-skip* CORDIC. It shows 3 cases: 1) no error detected; 2) error detected in one cycle (cycle 2); 3) every cycle triggers the E_{flag} . Timing-errors leads to reduced iterations.

The timing constraints, as specified in Table 5.2, were applied during synthesis and layout. The following explains these constraints in the CORDIC context.

- The max time borrow constraint was set to 0 for the latches in *DSTB*. This ensures no time borrowing under the nominal condition.
- Max delay for paths through *CORDIC cell 3 and 4* ($Q \rightarrow cell_3 \rightarrow D$), which is also called error mitigation setup in Table 5.2, is $t_{clk} \cdot (1 + \tau) - t_{setup_FF}$. The max delay is relaxed as they are designed for a better-than-worst-case situation, with the unlikely worst situations protected by time borrowing and the computation-skip. This enables area and power saving by gate-downsizing. The clock duty-cycle τ is set to 25%, which results in 34% area saving.
- Max delay for *computation-skip* paths ($Q \rightarrow skip\ path \rightarrow D$) is $t_{clk} \cdot (1 - \tau) - t_{setup_FF}$. This constraint ensures computation skip during timing-borrowing.
- Max delay for paths started from the *MSFF* in *DSTB* ($Q_{ff} \rightarrow D$) is $t_{clk} - t_{meta_window} - t_{setup_FF}$. As the *MSFF* in *DSTB* might fail under aggressive operating conditions, $t_{meta_window} = 700$ ps is reserved for meta-stability resolution. To ensure stability resolution, ULVT MSFF are used instead of SVT ones, as they have higher loop gains and hence lower resolution time constant (20 ps).
- Minimum delay for paths to *DSTB* ($Q \rightarrow D$) is $\tau \cdot t_{clk} + t_{hold_latch}$. This is achieved by automatic delay cell inserting during routing. This accounts for 8% area overhead.

5.4.4 Pre-silicon analysis

To make a fair comparison between *core1 canary* and *core2 CS*, they are designed to fail (produce errors) at the same frequency and voltage condition. This is achieved by synthesizing the circuit delay of the CS core (Cons.1 in Fig. 5.4) the same as the clock delay of the Canary core.

Fig. 5.10 shows the synthesizing frequency and energy consumption relation for the conventional CORDIC (without the modifications into *Canary* nor *CS*). Without many surprises, higher clock frequency leads to automatically gate-upsizing (by the synthesizing EDA tools), and hence the power consumption is increased.

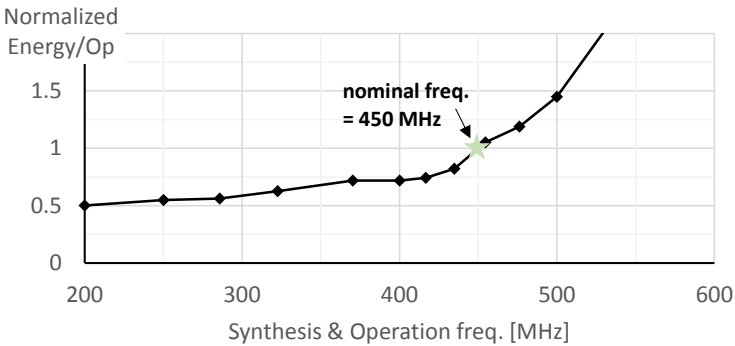


Figure 5.10: 450 MHz is selected for the operating condition. The figure shows the normalized energy consumption per CORDIC operation for the conventional CORDIC. The energy is measured with simulated gate-level toggling information.

This chapter chooses 450 MHz as the nominal operation frequency for both *core1 canary* and *core2 CS*, as shown in Fig. 5.10. The reasons are i) the speed meets the high-speed requirement, and ii) the gate-upsizing is moderate, and hence the energy consumption is not increased dramatically, compared with over-relaxed situations (e.g., 200 MHz).

Both designs are synthesized at the frequency of 450 MHz. After synthesize, placement and routing, the cell areas of the original CORDIC and its variants, *core1 canary* and *core2 CS*, are shown in Fig.5.11. The conventional CORDIC suffers from tight timing constraint to meet the worst-case corner, making the area very large. The *core1 canary* CORDIC utilizes more area. This is because its sequential logic area is almost doubled by replacing normal MSFF with *canary FF*.

For *core2 CS*, when τ is large, the area increase because of timing constraint (light blue bar in Fig. 5.11) is very small since the timing constraint is relaxed. This also exhibits a wider range of V_{dd} drop monitoring. However, higher τ calls for more delay cells to fix the short path issue, which leads to more area cost. For a balance between area (and hence energy) and V_{dd} drop monitoring, $\tau=25\%$ is chosen in this chapter. $\tau=25\%$ also makes the clock generation easy because it can be accomplished with a 4x frequency divider. According to Fig.5.11, the delay cells account for 8% of the total area when $\tau=25\%$.

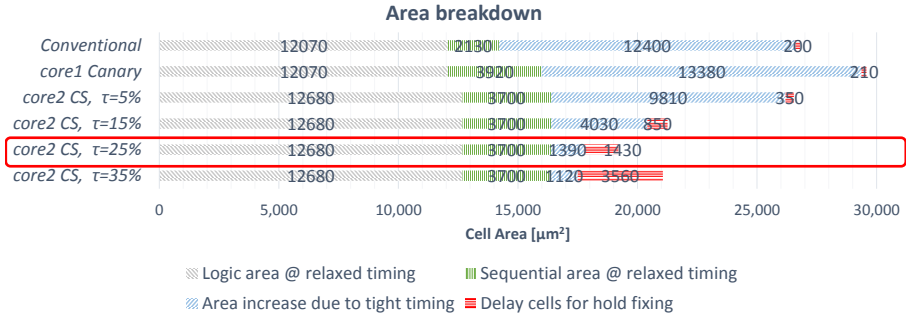



Figure 5.11: Cell area breakdown of the conventional and the proposed hardware CORDIC accelerators designed at 450 MHz. The *core2 CS* is designed with different duty cycle (τ). The area increase due to tight timing is computed against the cell area under 1/10 clock frequency (45 MHz). *cs* reduces area by loosen the timing constraints.

5.4.5 Post-silicon Measurement

The CORDIC accelerators were processed in a standard 28nm CMOS technology. A micro-graph of the chip is shown in Fig. 5.12. The complete chip measures $1 * 0.2 \text{ mm}^2$. The area for the *core1 canary* is 0.022 mm^2 . The area for *core2 CS* is 0.016 mm^2 . The area reduction comparing to *core1* is due to the relaxed timing constraints.



<i>Application</i>	CORDIC accelerator @ 4 CPI
<i>Process Technology</i>	Standard 28-nm CMOS
<i>Nominal condition</i>	450MHz @ 0.9V
<i>ENoB</i>	13.7 bits
<i>Area</i>	<i>Core1 canary</i> : 0.022 mm ² <i>Core2 CS</i> ($\tau=25\%$): 0.016 mm ²
<i>Energy/ENoB</i>	<i>Core1 canary</i> @ 0.9V: 6.6 pJ/bit <i>Core CS</i> @ 0.77V: 3.5 pJ/bit

Figure 5.12: Die photo and chip information.

To measure the output SNR and ENOB, random stimuli are generated in a desktop computer and send to an FPGA controller. The FPGA then writes/reads the data in the on-chip memory serially, before/after a CORDIC execution. Matlab and Python scripts are used for ENOB computation. For energy consumption measurement, testing vectors are stored in the on-chip memory. So the chip is filled with data and will run continuously. This makes the energy consumption results realistic.

The test chip was measured at 450 MHz. Fig. 5.13 shows the timing violations for both cores. The violations are regards as warnings for *core1 canary*. For *core2 CS*, they are actual errors. The *core1 canary* produces a warning when V_{dd} is lower or equal to 0.785V (PoFW). For *core2 CS*, when V_{dd} is lower than 0.805V (PoFF) or higher than 1.020V (PoFF), timing violations are asserted. The prior case is because of timing failure in the error mitigation path. The later case is because of the short path failure.

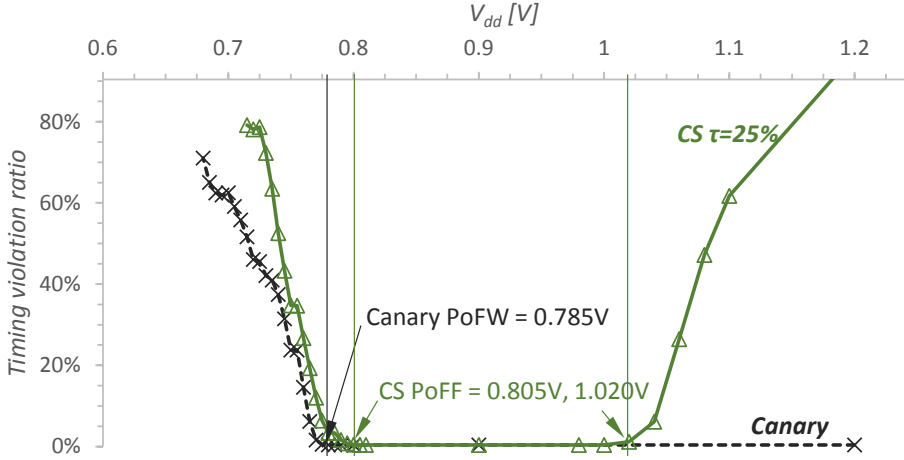


Figure 5.13: Measured timing violation ratio. A violation is observed if any path violates the timing. For *canary*, the first timing violation condition is PoFW. For the proposed CS (*computation-skip*), the first timing violation condition is PoFF.

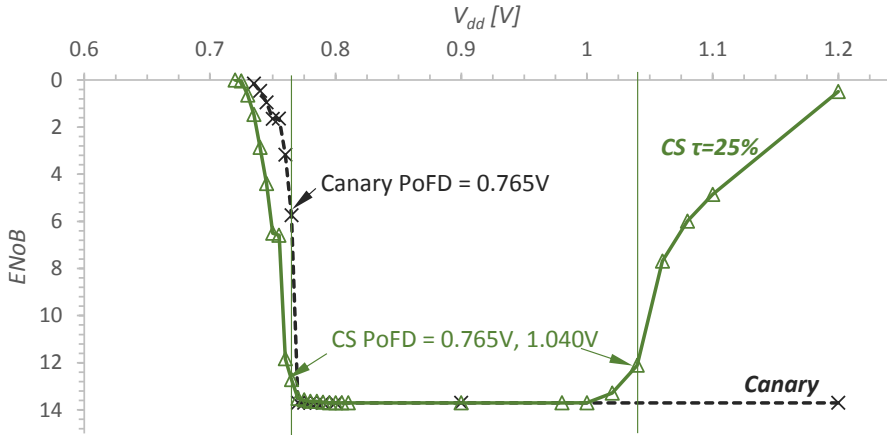


Figure 5.14: Measured CORDIC ENOB. For *canary*, the PoFD coincides with the PoFF. For the proposed CS (*computation-skip*), the PoFF does not result to PoFD, as the timing-errors are mitigated by the *computation-skip* scheme.

To quantify the error degradation, the notion of PoFD (Point of the First noticeable Degradation) is introduced. It represents the critical situation when

noticeable degradations are observed at outputs. Note that PoFD does not always coincide with the PoFF, as errors might be mitigated gracefully. Fig. 5.14 shows the ENOB as a measure of the output SNR. The ENOB is unaltered until the 0.785V (PoFW), as no actual timing-error is introduced. The PoFD for *core1 canary* is 0.765V. For *core2 CS*, when V_{dd} is lower than 0.805V (PoFF), *computation-skip* paths are activated and thus the ENOB slightly drops. The PoFD for *core2 CS* is marked on 0.765V. The PoFD for both cores are equal. This is because they are designed with the same frequency constraint, and are experiencing similar PVT conditions. Beyond the PoFD, the *computation-skip* paths and control paths both fails and hence the error tolerance capability becomes invalid. If the V_{dd} is higher than 1.040V, the ENOB decreases, due to minimum delay violations for DSTB. In summary, the ENOB performance is comparable for *core1* and *core2*, except for the high V_{dd} situation.

The energy consumption per CORDIC operation per effective bit is shown in Fig. 5.15. Under the nominal condition (0.9V), the *core2 CS* reduces energy consumption per ENOB by 28%, compared to the *core1 canary* (6.6 pJ/bit), which is attributed to relaxed timing constraint. For both cores, reducing the V_{dd} decreases the energy/ENOB. Going beyond the PoFDs leads to drastic energy/ENOB increase. The *core1 canary* saves 25% energy at the PoFW (0.785V), comparing with the 0.9V nominal case. This is error-free power saving that shaves the design margin. For the *core2 CS*, the energy reduction is measured at 42% at 0.805V (PoFF), where the ENOB is imperceptibly reduced. The energy/ENOB for *core2 CS* keeps reducing when going beyond PoFF ($V_{dd} \leq 0.805V$), despite the fact that the ENOB is slightly reduced due to computation skips. The minimum energy/ENOB is 3.5 pJ/bit at 0.770V, which is 46% lower than the nominal *core1 canary*. The corresponding ENOB is reduced from 13.7 bits to 13.5 bits.

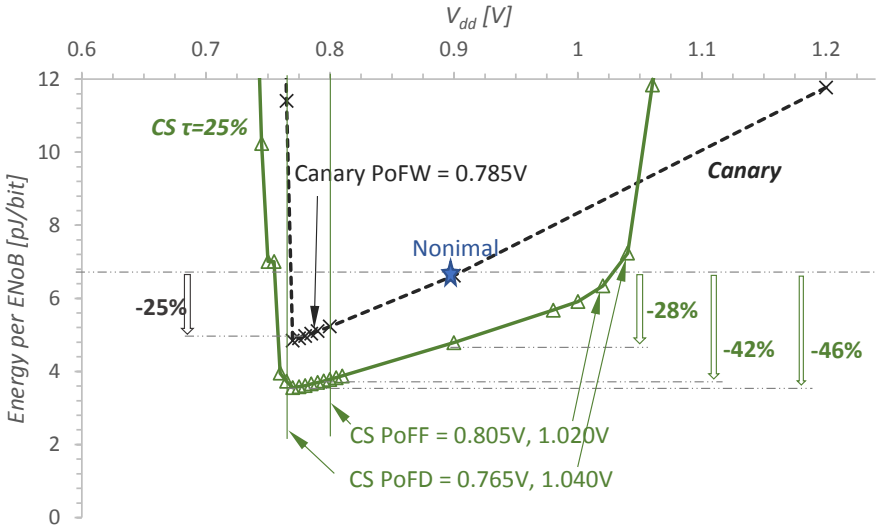


Figure 5.15: Comparison of the *core1 canary* (implementing the baseline state-of-the-art technique) and *core2 CS* (implementing the proposed scheme) CORDIC accelerators during voltage scaling. *core2* saves power by timing constraint relaxation, error-free adaptive scaling, and error-resilient VOS.

5.5 Conclusion

The CMOS variability margin leaves great room for energy savings. The timing-errors incurred by supply voltage over-scaling is most easily detected at the circuit-level. By providing this information to the algorithm-level, energy saving is achieved with gracefully output quality degradation.

In this chapter, error resilient techniques to set V_{dd} for variability margin saving, i.e., Canary, Razor, TIMBER, ANT, and computation-skip, are analyzed. In particular, the *computation-skip* scheme is discussed to mitigate timing-errors introduced by VOS. This scheme suits for evolutionary algorithms. It relaxes the timing constraint for a conventional circuit and hence saves power. The error flag can provide statistical timing-error rate, which indicates the output quality as well as the distance to the V_{dd} scaling limit.

The canary and the *computation-skip* schemes are applied to a recursive CORDIC processor. Effectively, the last CORDIC iterations are skipped once timing-errors are detected in this proposed *computation-skip* CORDIC. A 28% energy

consumption per bit saving due to relaxed timing constraint (design margin shaving at design time) is observed. The energy/ENOB saving improves to 42% because of adaptive scaling (error-free design margin shaving at run-time). Moreover, a total of 46% saving is possible, with a 0.2-bit precision loss.

Chapter 6

Application-level error-resilience on massive MIMO base stations

This chapter investigates application-level error absorption and handling. The considered errors are generated at the circuit-level and the algorithm-level errors by hardware operating at risky conditions. This chapter focuses on a massive MIMO communication system case-study. It shows that the perceived performance will hardly be affected by sparse processing failures, while the power consumption can be considerably reduced as error resilient hardware are utilized. Furthermore, this chapter assesses antenna outage impacts and proposes damage control strategies. The work in this chapter is published in [[Huang, 2017](#)].

The rest of this chapter is organized as follows: Section 6.1 introduces the opportunities brought by the massive MIMO system, as well as the cross-level optimization demands. Section 6.2 highlights the contribution. The system description from a functional processing point of view is presented in Section 6.3. Section 6.4 models the algorithm-level effects of circuit-level errors, or more specifically the VOS errors and the antenna outage error. The application-level impacts on the massive MIMO system are evaluated in Section 6.5. Section 6.6 proposes an approach to enhance the massive MIMO system performance under errors. Finally, the major conclusions of this chapter are summarized and relevant directions for future elaboration of the proposed concept are outlined

in Section 6.7.

6.1 Opportunities and challenges of massive MIMO

Massive MIMO is the currently most compelling sub-6 GHz physical-layer technology for future wireless access. The main concept is to use large antenna arrays at base stations to simultaneously serve many autonomous terminals, as illustrated in Fig. 6.1.

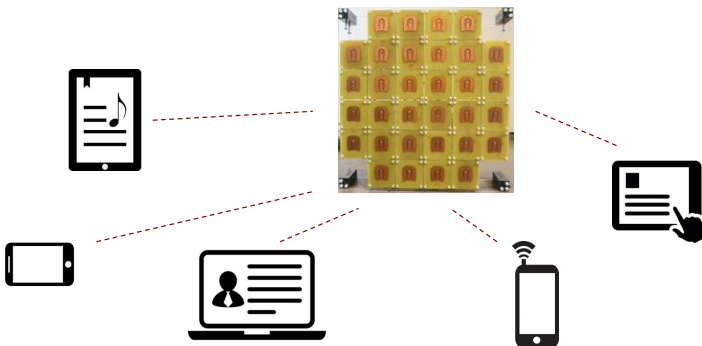


Figure 6.1: Massive MIMO exploits large antenna arrays to spatially multiplex many terminals [Larsson, 2017].

Since its inception about a decade ago, the massive MIMO concept has evolved from a wild “academic” idea to one of the hottest research topics in the wireless communications community, as well as the main work item in 5G standardization. The time for massive MIMO has come at this moment for two reasons: First, conventional technology has proven unable to deliver the spectral efficiencies that 5G applications are calling for. Second, the confidence in the exceptional value of the technology has spread rapidly since impressive real-life prototypes showed record spectral efficiencies, and the robust operation with low-complexity RF and baseband circuits has been substantiated [Prabhu, 2017].

6.1.1 Massive MIMO, the highly-demanded future technology

Massive MIMO opens up a new dimension of wireless communications by using an excess of base station (BS) antennas, compared to the number of active

terminals. This technique allows for very efficient spatial multiplexing, attainable using linear processing in a time-division duplex mode [Larsson, 2014].

Conceptually, massive MIMO achieve a 10x or more increase in system capacity. What is even more important is the gain in reliability due to flattening out of deep fades, hardening of the channel, and array gain. This especially benefits the cell edge users and could be essential for low power terminals as in Machine Type Communications (MTC).

This stunning improvement of massive MIMO results from the fact that much less transmitted power is needed thanks to the array gain. It also benefits from the utilization of low complexity hardware [Gunnarsson, 2017], as the individual antenna signals do not need to be of high precision [Desset, 2015; Gustavsson, 2014; Björnson, 2014].

6.1.2 Power consumption, a design challenge in massive MIMO digital processing

However, an obvious concern is how a large number of antennas (and associated transceivers and signal processing) will affect the complexity and energy consumption of the BS. [Desset, 2014] anticipates that the overall complexity and energy consumption in terms of J/bit can be lowered by a factor of 10 to 100 compared with current BS.

The energy consumption issue in digital systems is alleviated by the CMOS scaling. The scaling has brought steady power reduction for many generations, thanks to the decrease of the supply voltage that shows up as a quadratic factor in the dynamic power formula. However, Integrated circuits are facing ever increasing variability challenges in recent technology nodes (65nm and smaller). The process, voltage, and temperature (PVT) variability are considered as the three main contributors to circuit variability. Conventionally, to cope with this variability challenge, ICs are designed at the worst PVT corners, to ensure that they always operate correctly.

However, this approach brings considerable margins, leading to reduced peak performance and wasted power consumption. For instance, [Huang, 2016d] shows for 28nm technology, the performance difference (in terms of speed) is as large as 2.2x between the typical case and the worst-case. To reduce the margins, dynamical scaling techniques manage power dissipation and temperature using variable V_{dd} . The most adventurous methods are the error resilient techniques. They scale down the V_{dd} more aggressively (VOS) while accepting that errors might occur on individual chips. These methods have been proven to enable

significant energy savings while maintaining excellent performance for wireless communication [Huang, 2016d; Hegde, 2004].

In this context, considerable energy reduction potential is expected for massive MIMO if its low accuracy need is extended to the CMOS implementation of digital signal processing – to the point of sporadically processing distortion or even full failure of one or a few individual antenna signals. It, therefore, opens the door to much narrower design margins (comparing with the traditional semiconductor specification set at design-time). The circuits can operate at the lower supply voltage (and hence power).

6.2 Contribution of this chapter

This work combine the results form error resilient hardware with the inherent antenna redundancy in massive MIMO. It focuses on the TDD option of 3GPP LTE in a massive MIMO context. [Björnson, 2014] demonstrated the resilience to analog non-ideal hardware (e.g. nonlinearity).

To embrace unreliable hardware, this work proposes to consider the digital computation as a faulty process. It demonstrates that if a limited number of circuit-level and algorithm-level computational errors can be tolerated at the application-level, the safety margins can be reduced significantly. This will bring considerable power saving with minor performance degradation.

Apart from demonstrating the inherent error resilience, this work also proposed methods to detect extensive errors and adjust the massive MIMO application to prevent further quality loss.

6.3 Masive MIMO system introduction

In a massive MIMO system, the BS is equipped with M antennas and serves K single-stream users simultaneously, each equipped with a single antenna. Unless otherwise specified, $M = 100$ and $K = 10$ is set as typical values in this chapter. Fig. 6.2 illustrates the BS architecture of a massive MIMO system.

The BS consists of central digital modem functionality, the per-antenna processing including (I)FFT operations for OFDM (de)modulation, digital front-end (DFE), analog front-end (AFE) and power amplifier (PA). The signal processing complexity and power consumption of the inner-modem digital

processing scale linearly with K , while the (I)FFT, the DFE, and the AFE complexity all scale linearly with M , and the pre-coder scales with $M * K$.

The massive MIMO digital processing complexity [Desset, 2014] is summarized by billion complex floating-point arithmetic operations per second in Table 6.1. The data transfer overhead is included. For a typical massive MIMO system, the digital processing effort is dominated by the per-antenna functionality (mainly FFT and DFE filtering operations). This is due to the linear dependence of system power consumption to the massive BS antenna number M .

The digital modem is split in the outer modem (processing information bits through channel coding/decoding) and the inner modem (in multi-carrier systems performing frequency-domain operations such as channel estimation and massive MIMO precoding). To determine the signal processing demands for a typical BS, the complexity of the digital components of massive MIMO [Desset, 2014] is summarized in Table 6.1. These complexity values estimate the number of billion complex floating-point arithmetic operations performed per second for each specific digital signal processing operation. They have been multiplied by an overhead factor (see [Desset, 2014]) to take data transfers (in memories and registers) into account. Because they take a big portion of the power consumption of digital systems.

Table 6.1: Complexity of digital components for a 100x10 massive MIMO system in each phase, with 20 MHz bandwidth, 3 bps/Hz (16-QAM, 3/4 coding rate, training not included)

<i>Subcomponent</i>	<i>Downlink (DL)</i> <i>[GOPS]</i>	<i>Uplink (UL)</i> <i>[GOPS]</i>	<i>Training</i> <i>[GOPS]</i>
Inner modern	175	520	290
Outer Modern	7	40	0
DFE incl. (I)FFT	920	920	920

The workload of the massive MIMO is partitioned into downlink (DL), uplink (UL), and training phases. The training phase uses the UL signals to perform channel estimation. Therefore, its digital processing components are similar to the UL phase. For a typical massive MIMO system, the digital processing effort (see Table 6.1) is dominated by the per-antenna functionality (mainly FFT and DFE filtering operations) [Desset, 2014]. Note that the downlink data traffic is usually larger (5 to 20 times more traffic than UL). In terms of overall complexity, a BS spends more effort on the UL phase.

To minimize the area cost and the energy budget of the BS, this chapter focuses

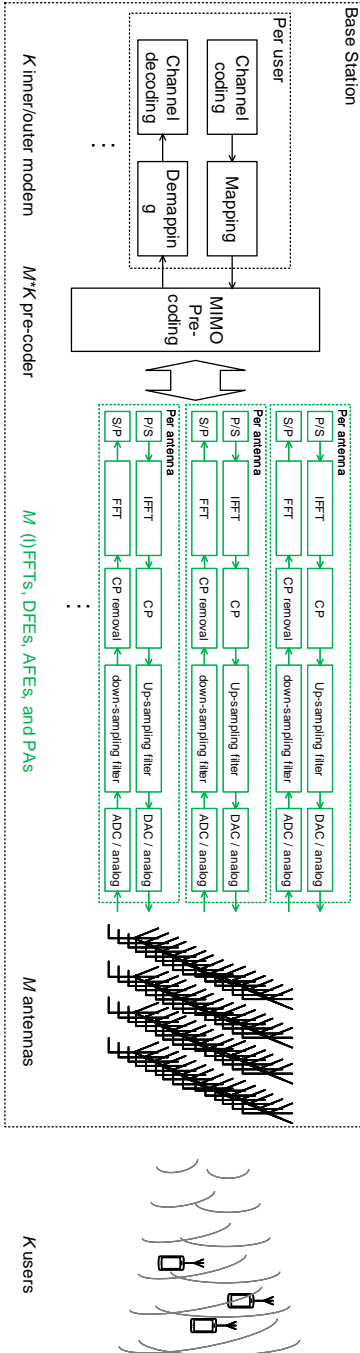


Figure 6.2: A massive MIMO BS equips with M antennas and serves K users. Typically, $M = 100$ and $K = 10$.

on demonstrating the possibilities of accepting intermittent digital hardware errors in the DFE (incl. FFT). The hardware error effects are considered within the context of DL massive MIMO. The reasons are i) The DL data size is usually larger, which makes the DL the dominant power consumer. ii) The DFE for UL has a lot of similarities to the DL, and thus exhibits similar effects on errors.

6.4 Digital hardware error impacts on signal quality

This section illustrates the most common sources of errors in digital signal processing. The impact of these circuit errors on the (I)FFT and other DFE hardware are then modeled for the later application-level assessment. Accordingly, the unreliability of the circuits is becoming a non-negligible issue [Rabaey, 2008].

For the massive MIMO system, the digital hardware errors in (I)FFT & DFE introduced by silicon unreliability and by adventurous design methodologies result in incorrect bit results during signal processing. This can be regarded as digital distortion noise. This work introduces a new metric to signal to digital distortion ratio (SDDR) to describe the quality of signal:

$$SDDR = 10 \cdot \log \frac{\sigma_s^2}{\sigma_d^2}, \quad (6.1)$$

where σ_s^2 and σ_d^2 are the powers of error-free DFE output, and the noise power of digital distortion due to circuit unreliability, respectively. The digital distortion noise results from circuit-level errors. Based on its mechanism, it is categorized into two class, i.e., VOS (temporary and local errors), and antenna outage (hard and full antenna errors).

The error-free output of a DFE, y_s , is contaminated by the VOS distortion n_d . Therefore, the final contaminated DFE output

$$\tilde{y} = \begin{cases} y_s, & \text{if } error - free \\ y_s + n_d, & \text{if } VOS \text{ errors} \\ 0 \text{ (fixed value)}, & \text{if } antenna \text{ outage} \end{cases} \quad (6.2)$$

The VOS errors n_d of is modeled with a zero-mean Gaussian distribution [Liu, 2010] with σ_d^2 as the error power. The effect of VOS can thus be molded with SDDR. For the scenario of antenna outage, the DFE output stuck at a fixed value (1 or 0). The information from the input signal y_s is completely lost. The rest of this section discuss the VOS and antenna outage effects at the algorithm-level.

Regardless of the DFE contamination class, the received signals at the MIMO receiver (y) are

$$\begin{bmatrix} y_1 \\ \vdots \\ y_K \end{bmatrix} = \begin{bmatrix} h_{1,1} & \cdots & h_{1,M} \\ \vdots & \ddots & \vdots \\ h_{K,1} & \cdots & h_{K,M} \end{bmatrix} \begin{bmatrix} \tilde{y}_1 \\ \vdots \\ \tilde{y}_M \end{bmatrix} + \begin{bmatrix} n_1 \\ \vdots \\ n_N \end{bmatrix} \quad (6.3)$$

$$y = H\tilde{y} + n, \quad (6.4)$$

where H denotes MIMO channel matrix. The vector of received symbols y is distorted by the noise vector n . Note that the hardware distortion n_d and n are different in nature, as n_d impacts individual transmitter antenna, n suffers from the channel and receiver.

6.4.1 VOS (Voltage-OverScaling) impacts

Section 2.2 reviewed the techniques to scale down the voltage to exploit design margins and save power consumption. In this chapter, their impacts on SDDR is re-evaluated, in the context of the Massive MIMO digital front-end.

The power saving consists of two part: error-free power saving and error-resilient power saving. The error-free part can be achieved usually by the Razor techniques. For instance, [Bull, 2011] reports 30% and 52% power consumption saving on a typical die and a fast die, respectively; [Fojtik, 2012] achieves 54% saving on a typical die and 60% saving on a fast one. The infrequent errors timing-errors are fully resolved by the micro-architectural level error correction schemes, and produce no errors to the signal output.

The error-resilient techniques, as reviewed in Section 2.3.2, reduces the power consumption of digital signal processors by gracefully sacrificing the SDDR, admitting that a certain amount of errors might occur. For example, Chapter 5 saves 45% power consumption for CORDIC applications, at the cost of 1 ENdB degradation. However, if the supply voltage is further reduced for more aggressive power savings, the SDDR is reduced dramatically. e.g. lower than 0 dB, because a lot of setup paths are failed and the circuit cannot operate correctly.

In summary, state-of-the-art algorithm-level error-resilient techniques save around 40% power, at the cost of potential sparse antenna processing distortion. The SDDR depends on the operating V_{dd} , the process variability and the environment temperature. This means that even with the same design, different (I)FFT & DFEs of the massive MIMO might exhibit vastly different SDDR behavior. When the circuit is mainly subject to random SEU errors, designs

can choose to either carry on using the erroneous signal, or selectively harden the most critical component using the knowledge presented in Chapter 3. In addition, error mitigation techniques, e.g. the recursive mitigation scheme presented in Chapter 5, are encouraged in this work to avoid dramatic SDDR degradation.

If the SDDR of an individual functional block cannot be sustained in a cost efficient way, application-level redundancy is preferred. Section 6.5 analyzes the massive MIMO application-level effects of VOS errors and assures that circuit degradation on a small portion of antennas can be absorbed in the massive MIMO system.

6.4.2 Antenna outage impacts

Another hardware failure scenario for the DFE is the antenna outage (antenna is completely non-operational). This happens when the power supply systems are broken, or when a circuit controlling signal is corrupted, e.g. failure to wake-up the digital circuit.

In an antenna outage scenario, the DFE output is stuck at a fixed value, which is assumed to be the maximum value (DFE output $Y = \text{maximum}$). The SDDR of the outage antenna is $-\infty$, as the signals from the victim antennas are completely lost. This model is regarded as one of the most pessimistic hardware failures. Note that the $-\infty$ SDDR does not imply infinite noise to the whole system, as only the victim antennas are affected and their PA powers are normalized among all antennas. Therefore, several antenna outages will not fail the system entirely.

6.5 Random antenna error impacts assessment

Consider a TDD massive MIMO system in DL with $M = 100$ and $K = 10$, where the channel estimation and the minimum mean square error (MMSE) MIMO pre-coding are free from digital hardware errors. The performances over a Rayleigh 20-tap i.i.d. channels are simulated. The system is OFDM-based according to LTE parameters, i.e., 1200 loaded subcarriers in a 20 MHz band.

The channel is estimated through uplink pilots associated to the different user equipments (UE) in a round robin fashion, i.e., one pilot every 10 subcarriers for a given UE. Since the channel estimation is assumed to be perfect. The simulation in this work cannot take advantage of the MMSE pre-coding that would limit the digital distortion errors. This is because the digital distortion

is not present in the channel training phase. SNR is defined based on a total transmit power normalized to 0 dB per user. The emitted power is normalized for each antenna. The simulations do not apply error correction coding (ECC), except for Fig. 6.8 where the effects of coding on digital hardware errors are studied.

6.5.1 Error effects on uncoded QPSK 100x10 massive MIMO

This subsection discusses the effect of digital hardware errors, when some per-antenna digital processing units are suffering from errors.

Assume that due to local PVT variation and semiconductor aging effects, a portion of the antenna IFFT & DFE are suffering from slight VOS hardware errors, i.e. SDDR = 10 dB for victim antennas, and no digital hardware errors occur for the remaining antennas.

The system bit error rate (BER) degradation because of antenna errors is illustrated in Fig. 6.3. The BER performance drops slightly as more antennas are affected. Nevertheless, the degradation is small even with 50% antennas affected.

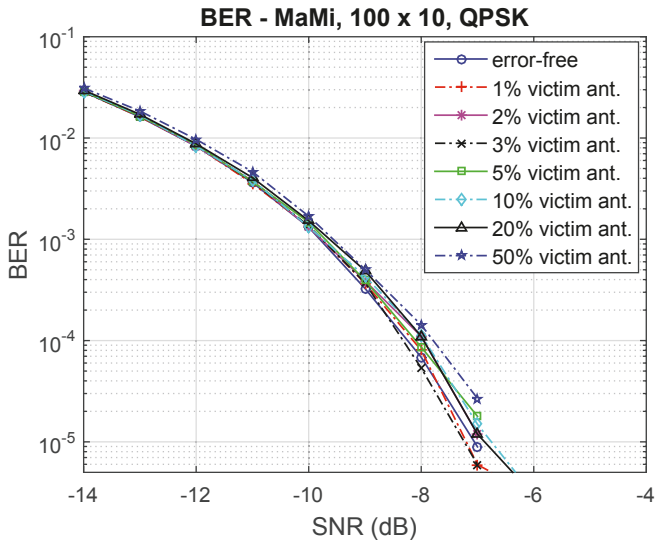


Figure 6.3: Slight VOS digital distortion errors, i.e. SDDR = 10 dB, only degrades the system BER marginally. This is observed by the massive MIMO system BER vs. channel SNR plot. Randomly chosen victim antennas are suffering from VOS errors. The remaining antennas are free from these errors. The pre-coding scheme is MMSE.

Supposing the VOS distortion noise is larger as designers further exploit the design margin, the digital hardware errors become more frequent and hence the SDDR is smaller for each given antenna, e.g. 0 dB. The resulting BER performance with the same settings is shown in Fig. 6.4. For a target BER of 10^{-4} , massive MIMO with 20% antenna failing only requires a channel SNR of -7.4 dB, as opposed to -8.4 dB for the error-free case. This shows that the massive MIMO system still will operate correctly even if a noticeable amount of antennas suffer from digital hardware errors.

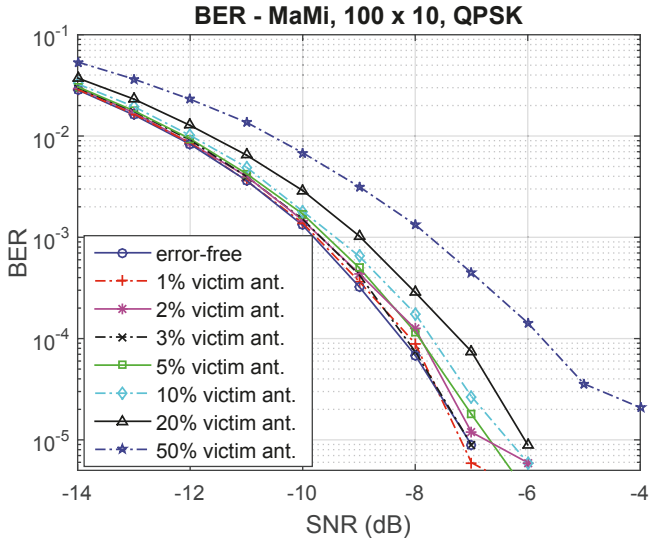


Figure 6.4: If the massive MIMO system is suffering from extensive VOS digital errors on some antennas, i.e. SDDR = 0 dB, the output is still manageable.

In Fig. 6.5, the massive MIMO BER when applying the most pessimistic antenna outage model is shown. For the victim antennas, the useful signals are completely lost and a constant value is output from the DFE and emitted by the PA. This corresponds to an infinitely small SDDR for these victim antennas. The resulting BER performance shows larger SNR degradation for the same BER target. Nevertheless, the massive MIMO system can still cope with the antenna outage error thanks to the redundancy of antennas in the BS, at least for a failure rate up to 10%.

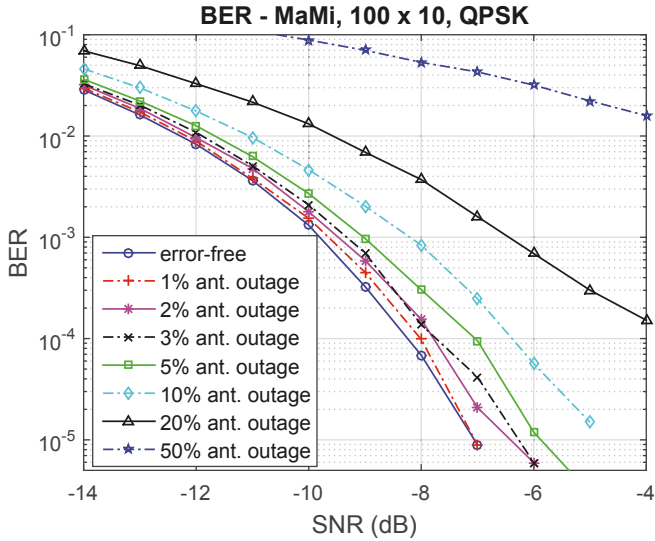


Figure 6.5: The system can tolerate some antenna outage (DFE output stuck at a fixed value), i.e. $SDDR = -\infty$.

6.5.2 Antenna outage effects for other massive MIMO setups

This subsection discusses the BER performance of massive MIMO for different settings. To analyze the most pessimistic situation, the antenna outage model is used.

Fig. 6.6 displays that for the 100x10 massive MIMO, 10% antenna outage leads to slightly more BER degradation for QPSK, comparing with BPSK. This is due to the larger error margin for simpler modulation scheme. For the more sensitive 16-QAM modulation scheme, 10% antenna outage leads to a huge degradation in DL BER. This implies that for communication systems where channel SNR is worse and simple modulation schemes are used, the reliability requirement of the antennas can be relaxed, to simplify the (I)FFT & DFE design and reduce the power consumption budget.

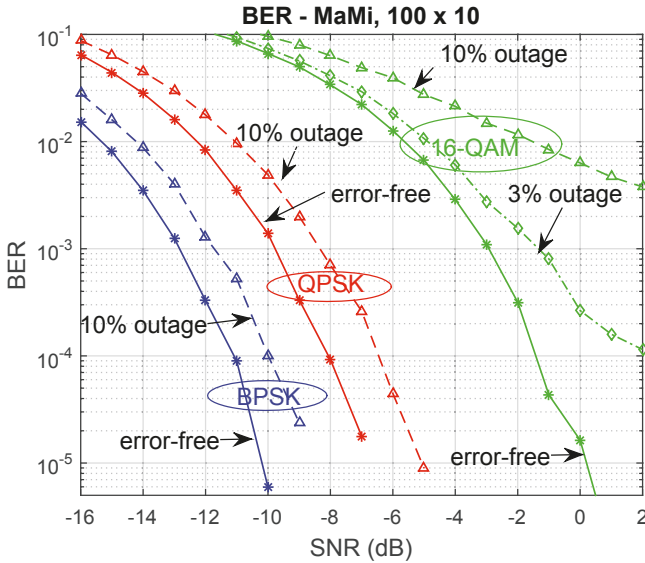


Figure 6.6: Massive MIMO system with random antenna outage errors for different modulation schemes, i.e. uncoded BPSK, QPSK and 16-QAM. Simple modulation schemes are more resilient to antenna outage.

The BS antenna redundancy is reduced if the load of the massive MIMO system increases (the number K of served users or streams is increased). In this scenario, the tolerance for antenna outage is decreased, compared to systems with small K (Fig. 6.7). Nevertheless, For massive MIMO systems where $M \gg K$, the amount of antenna redundancy is sufficient to provide opportunities for antenna unreliability.

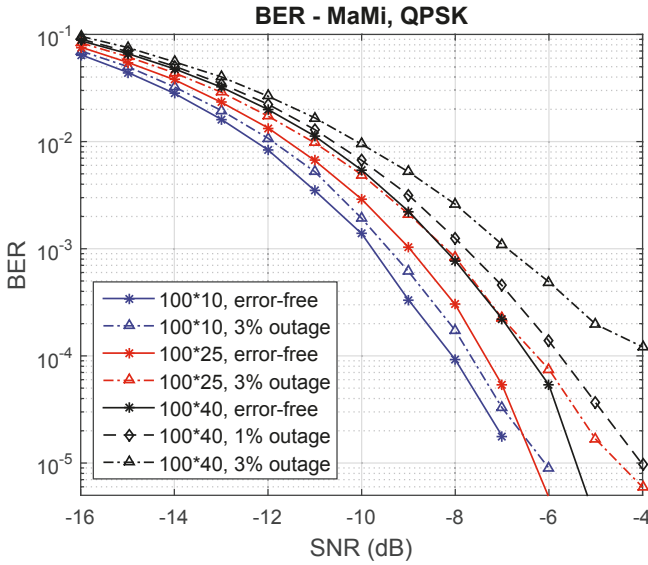


Figure 6.7: Massive MIMO system with random antenna outage errors for various loads, i.e. 100×10 , 100×25 , 100×40 . Lower loads leave more spaces for error absorption.

So far uncoded results were presented. However, errors in massive MIMO systems can be mostly corrected by error correction codes, e.g. convolutional codes and LDPC codes. Fig. 6.8 shows the BER improvement when $3/4$ soft decoded LDPC code is utilized in the massive MIMO system. At the targeted BER of 10^{-4} , the SNR is 6 dB lower for the coded QPSK, compared with an uncoded case. For such BER, the SNR difference when considering antenna outage is smaller for the coded massive MIMO system, compared to the uncoded one, although a limited degradation always remains.

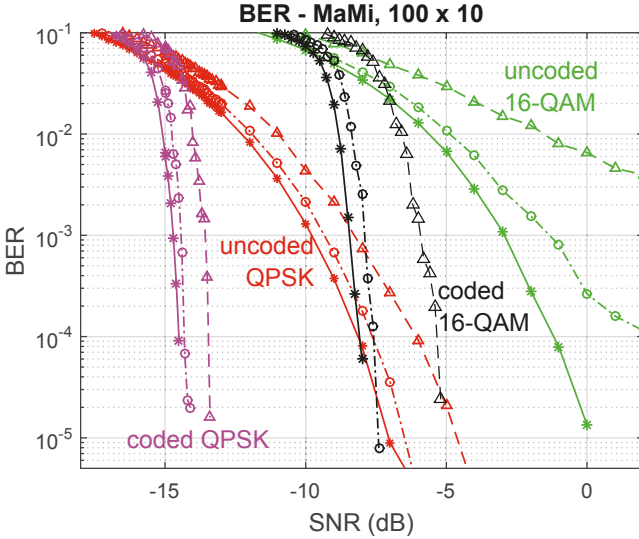


Figure 6.8: Massive MIMO system with random antenna outage errors for uncoded and coded (3/4 soft LDPC) QPSK, and uncoded and coded (3/4 soft LDPC) 16-QAM. The legend denotes: i) error-free (star shapes), ii) 3% victim antennas (circle shapes), and iii) 10% victim antennas (triangle shapes). Coding and uncoded degradations are similar.

6.6 Controlled antenna outage

According to Section 6.4.1, VOS with error-resilient techniques bring up to 40% power saving, at the risk of failures for few antennas. The simulation results from Section 6.5 illustrates performances when no error detectors are equipped. In other words, the massive MIMO system is operating as usual, regardless of hardware errors. In this situation, the massive MIMO manages to sustain system performance even if several antennas are non-operational (outage) due to aggressive VOS or completely DL failure. In order to improve the reliability of the system under hardware errors, this work proposes to firstly detect hardware errors, and then either correct errors, or circumvent the defective hardware if correction is not possible. It is worth noting that the distortion originating from digital circuit failures fundamentally differs from the random noise introduced in communication channels. While CMOS process variations may feature continuous random distributions, their effects typically lead to discrete antenna error events.

Dedicated monitoring circuit can be established to detect these errors and thus these erroneous bits can be labeled unreliable and potentially be corrected. Eventually, if some circuit errors get too large or systematic, measures at the system level can be taken to discard this hardware and increase the overall robustness.

If digital hardware designs provide monitors [Ernst, 2003; Bowman, 2011; Bull, 2011; Fojtik, 2012] for each (I)FFT & DFE, the massive MIMO system can equip a closed loop for error detection and correction. The error-resilient designs shown in Chapter 5 can detect and mitigate some errors at the algorithm level. Signals from erroneous antennas can thus be exploited, as the errors are small. The error effects are shown in Section 6.5.

Another countermeasure is to disable the victim antennas temporarily. Therefore, the channel estimation (and hence the precoding, and data transmission) are accomplished the remaining error-free antennas only. This method is equivalent to operating with a reduced number of error-free BS antennas M . For systems with large redundancy, e.g. using simple BPSK modulation, this method will hardly impact system quality. In Fig. 6.9, erroneous antennas are taken out completely. This leads to BER worse than Fig. 6.3 and Fig. 6.4, where antennas are affected with moderate digital hardware errors and the signal from these victim antennas are still exploited for communication.

This shows that by detecting the degree of antennas failing (noise power), designers have the option to determine whether to exploit the victim antennas or to discard them, for better performance. The noise power can be estimated from circuit-level or algorithm-level error monitors, or from the system level measurement that combines channel noise, e.g. by evaluating the measured channel information (CSI).

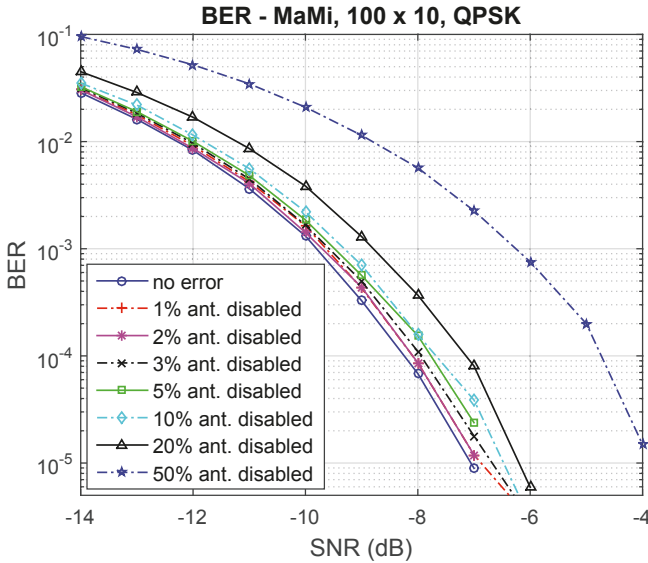


Figure 6.9: Massive MIMO system performance when victim antennas are discarded and the channel estimation and DL is carried out by the remaining error-free antennas. This situation is better than antenna outage.

This work proposes an error detection strategy to periodically check the antenna functionality by putting one antenna in the testing-mode at one time (Fig. 6.10). During the testing mode, per antenna DSP are supplied with testing inputs. The outputs are compared with the pre-computed data. If the results are vastly different, the antenna is detected erroneous, and thus the V_{dd} is increased to reduce errors and hence guarantee performance. Erroneous antennas are permanently disabled if they kept failing.

In this work, the periodical testing is scheduled when no data transmission is taken place. Moreover, it can even be performed on-the-fly during data transmission, since suppressing one (1% for massive MIMO with 100 antennas) antenna during DL into the testing-mode would not introduce huge degradation (see Fig. 6.9). This enables timely fine-grained V_{dd} adjustment, which maximizes power savings. If, however, the antenna is permanently damaged and thus cannot recover by increasing V_{dd} , the antenna will then be labeled as defected.

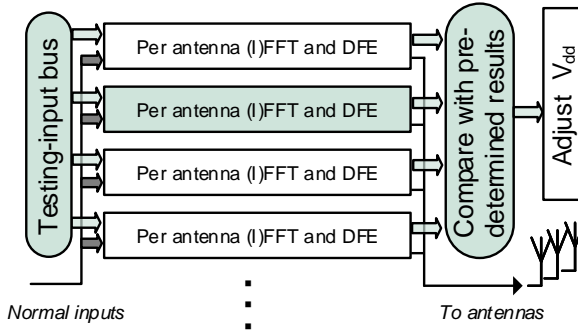


Figure 6.10: An error resilient adaptive scaling technique to manage hardware errors. This technique periodically checks DFE functionality, and adjusts V_{dd} accordingly. In this figure, the second antenna is in testing-mode.

The time interval of the periodical testing depends on the nature of the error occurrence and the changing environment (Fig. 6.11). The device process variability is usually determined after manufacturing. Thus, a pre-installation V_{dd} adjustment is sufficient to account for this variability. The CMOS aging effects are becoming more evident in the scaled technology, they depend on the work loads (voltage, frequency, and the relaxation duty cycle). If long-term aging is the only concern, checking the correctness of every hour is sufficient. If, however, the relaxation factor is considered, which alters the device characteristics in several-hundreds cycles, periodically checking the results by every millisecond is recommended. The temperature, which is mainly subject to the heating and dissipation efficiency, usually changes in the ranges of seconds. The voltage noise occurs in the picosecond range. Thus, voltage-noise incurred errors, if the occurrence is rare, can be absorbed in the resilient computing, without counter-measures after periodical testing. If the occurrence becomes more frequent, e.g. once in every thousand cycles, errors will be captured by the periodical testing, and hence the system can opt to either disable the unit or to increase the voltage. The SEU occurrence rate for on-ground application is low. Therefore, communication systems usually do not address this issue.

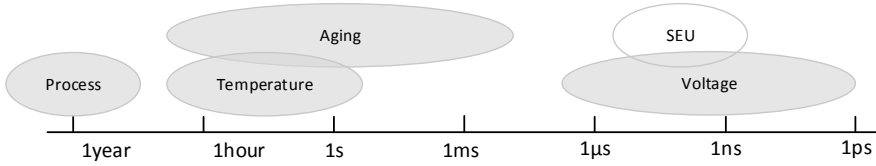


Figure 6.11: The frequency of changing of the process, aging, temperature, and voltage noise that affect the timing errors of circuits. Effects with slow changing rate can easily be resolved by periodical checking, e.g. process and temperature. Designs should either leave margins, or mitigate errors by resilient design toward slow changing effects, e.g. voltage noise and SEU.

6.7 Conclusion

This chapter examines the opportunity of using error-prone digital signal processing components in massive MIMO systems, and proposes a strategy to maximize power savings while still offering robust operation. The (I)FFTs & DFEs in massive MIMO are the most critical digital components in terms of area and power consumption as they scale linearly with the massive BS antenna count M . Hardware errors in a number of antennas' (I)FFT & DFE can be absorbed by the massive MIMO system thanks to the redundancy coming from the large antenna number. The massive MIMO system exhibits error resilience even for the worst-case antenna outage scenario.

When the hardware error distortion power is low, e.g. lower than 0 dB, the massive MIMO system should continue using the erroneous signal. The errors can be corrected at the application-level by other redundant antennas. It is proposed for systems to equip with on-chip monitors, so that they can detect hardware errors on-the-fly, and discard the error-prone antennas when their SDDR is large, e.g. when antenna outage, or when components are suffering from severe aging effects.

This provides opportunities for the digital hardware designers to embrace cost-efficient and reduced-power digital components at the expense of sacrificing individual antenna reliability, yet maintaining overall systems performance. Up to 40% power can be reduced for the considered digital processing components.

Chapter 7

Conclusions and future work

Digital circuit designs have benefited from the free lunch of technology scaling for many decades. Nevertheless, the free lunch starts to diminish as we are entering the deeply scaled era. Clever design optimizations are becoming more demanding than ever before. This work advocates a cross-layer to investigate through multiple design levels for power saving. This blurs the distinction between traditional design levels, especially in terms of handling variations, environmental and runtime uncertainties, and errors. This leads to global saving on power consumption as demonstrated in this work. This chapter concludes this thesis. Section 7.1 highlights the results of the thesis. Section 7.2 discusses the key messages from this work. Section 7.3 lists some future work suggests interesting further extension of this work.

7.1 Conclusions

In an era where performance and power are heavily constrained, the conventional worst-case design methodology no longer suffices. A cross-layer optimization is encouraged to quantify the actual need from all levels at design time to avoid over-design. Therefore, circuit and system designers should work more coherently to provide just-needed quality and minimize power consumption. This work advocated for a cross-layer optimization methodology for quality-power trade-offs.

In Chapter 2, approaches for power and quality trade-offs were reviewed. It evaluated the device-level phenomena of PVT variations and reliability threats.

It explained the limitation of the worst-case design approach. The chapter also reviewed the adaptive scaling method that uses replica circuits or in-situ error-detection flip-flops. The error-detection flip-flops offer large benefit, due to the capability of responding to fast variations and critical path activation differences. Finally, the motivation for cross-layer error resilience was presented. In a lot of applications, higher-level designs can easily absorb and handle errors that were seemingly inevitable at device and circuit-level.

Circuit-level: random error impact model

In Chapter 3, a circuit-level random error model was presented. The model predicts the impact of device errors on algorithms. In contrast to conventional models that require time-costly Monte-Carlo simulation, the proposed model uses an analytical approach. It ranks the flip-flop nodes in a digital circuit according to their contribution to the outcome. The contribution is defined as its *significance*. A flip-flop is significant if it tightly connects to a lot of significant flip-flops. The model is thus named as SERIAL, or SignificanceE RankIng ALgorithm.

This automation eliminates the need for conventional trial-and-error searching for suitable error hardening. Circuit designers have the opportunity to selectively ensure the most important FF (e.g. FF hardening, VOS margin), without excessive hardening overheads. The efficiency and effectiveness were shown on benchmark circuits. The design principles were applied to the design of a reliable FFT processor, which cuts the hardening overhead by a half.

Microarchitecture-level: fine-grain hardware switch for power saving

Chapter 4 proposed a microarchitecture-level fine-grain hardware-switch scheme to save power in embedded processors. This scheme exploits word-length optimization opportunities for a multiplication unit. The opportunities were justified on 11 typical signal processing applications. It is shown that half of the inputs of 32-bit multiplications are shorter than 8 bits.

Therefore, this work proposed a redundant short multiplier to perform these short applications. This leads to power saving as the toggle circuit complexity is reduced. The proposed hardware-switch scheme was validated on the OpenRISC platform. Without changing the software compiler, It brings 23.7% power saving for the multiplication unit, which accounts for 9.5% power saving for the whole execution unit.

Algorithm-level: computation-skip scheme to trade quality for power savings

Chapter 5 proposed an algorithm-level error mitigation method, computation-skip scheme. It trades quality for power savings in recursive applications without throughput penalties. Errors produced at the circuit-level are mitigated, without error accumulation, at the algorithm-level.

The chapter validated the power saving of the scheme on a CORDIC hardware accelerator. The accelerator is processed and verified in a standard 28nm CMOS process with only standard-cells. Using only standard-cells, this work eliminates the traditional semi- (or even fully-) customized design effort for in-situ error detection circuits. A 28% energy consumption per bit saving due to relaxed timing constraint (design margin shaving at design time) is observed. The energy/ENOB saving improves to 42% because of adaptive scaling (error-free design margin shaving at run-time). Moreover, a total of 46% saving is possible, with a 0.2-bit precision loss.

Application-level: embracing erroneous hardware in Massive MIMO systems

Finally, Chapter 6 investigated application-level error absorption and handling. The considered errors are generated at the circuit-level and the algorithm-level errors by hardware. The chapter focuses on a Massive MIMO communication system case-study. It shows that the perceived performance will hardly be affected by sparse processing failures, while the power consumption can be considerably reduced as error resilient hardware are utilized. Furthermore, this work assesses antenna outage impacts and proposes damage control strategies.

When the hardware error distortion power is low, e.g. lower than 0 dB, the Massive MIMO system should continue using the erroneous signal. The errors can be corrected at the application-level by other redundant antennas. It is proposed for systems to equip with on-chip monitors, so that they can detect hardware errors on-the-fly, and discard the error-prone antennas when their SDDR is large, e.g. when antenna outage, or when components are suffering from severe aging effects. This provides opportunities for the digital hardware designers to embrace cost-efficient and reduced-power digital components at the expense of sacrificing individual antenna reliability, yet maintaining overall

systems performance.

7.2 Key messages

The traditional method to handle process variations, environmental and runtime uncertainties, where excessive safety margins are added, result to huge power efficiency loss. One benefit of this conventional approach is that it simplifies design – if we are uncertain about some parameters, take a reasonable worst-case assumption and make sure that works. However, this must be changed as no precious power efficiency should be wasted because of a lack of engineering effort.

There are in general two general directions to advance. First, an accurate model is desired. This model should eliminate the unnecessary pessimism. For instance, as the transistor ages according to the operating voltage and frequency, the worst-case settings for systems where the supply voltage is low should be less pessimistic [Kükner, 2013]. This enables easier timing closure and hence power saving. The more accurate the model, the more information need to consider during the modeling, this calls for cross-layer modeling. Eventually, the effort in modeling will outweigh the power gain at some point. Another roadblock for pursuing a perfectly accurate model is that the environmental and runtime uncertainties vary over time and changes very fast. An accurate model at one point becomes invalid at another time. Again, safety margins remain.

Second, adjusting to the unpredictable changes when they arise is the advocated approach in this thesis. As it is fundamentally impossible to model precisely beforehand, the logical approach is dynamical changing according to the condition. Cross-layer information is encouraged as optimization locally within design levels are not enough.

The DVFS and the more aggressive AVFS falls in the second category. With speed detectors and fine-grained voltage regulator, the time-zero and time-dependent process variations can be largely eliminated. [Fojtik, 2012] reports 54% energy saving on an average die with this approach, without introducing any errors. Despite the remarkable benefits, the engineering challenges of i) the accuracy of speed monitor, ii) overhead of in-situ speed detection, iii) response time of the voltage-regulator are still the obstacles to mass adoption. Nevertheless, the huge energy saving has motivated engineers to exploit this opportunity, especially in the processor industry where a massive amount of chips are produced. The immense recursive benefit certainly outweighs the non-recursive engineering investment. The AVFS, realized by adaptive clocking distribution and power management, is becoming a common feature in modern

processors, e.g. [Gonzalez, 2017]. The other demanding application for AVFS is the ultra-low power IoT terminal devices, where every microwatt matters. It is almost impossible to find a sub-threshold computing device that does not equip AVFS to manage variability.

Apart from error-free DVFS, the error is another factor for cross-layer optimization. It is a pity that engineers spend so many energy to optimize power with the error-free constraint while the constraint is not necessarily needed. In the end, the digital devices serve to provide service, not to compute correctly. This work demonstrates huge potential in the wireless communication system. In those systems, errors are corrected by the ECC as long as they are small. Traditionally, circuit-level reliability hardening (FF hardening) are employed in mission-critical circuits to ensure error-free circuit. This is however unnecessary. In Chapter 3, a cross-layer optimization by selective FF hardening is performed on an FFT processor against random SEU. It increases reliability with much less hardening overhead. This methodology is advised in this work – when trying to improve reliably, optimize with minimum overhead, and do not assume that reliable means error-free circuits.

Similar design approaches are carried out on algorithm (Chapter 5) and application-level (Chapter 6) designs. The error-resilient approach advances from the canary FF method (an AVFS approach) in terms of power minimization, especially for high-speed circuits. This is because error-resilience relaxes the strict timing requirement in those circuits and uses slow but power-efficient digital cells.

The massive MIMO and the 5G technology will certainly demand massive digital solutions in the future. A good news is that the massive MIMO tolerates plenty of errors thanks to the antenna redundancy. Errors include conventional channel errors, analog component non-ideality, quantization errors, VOS errors and even hardware failure. Its resilience motivates the popularity of massive MIMO technology. Moreover, it encourages low-power but erroneous hardware solutions. The DVFS with sparse errors and the algorithm-level solution in Chapter 5 fits into this picture perfectly.

Overall, device uncertainties, that used to be a silicon-only problem, should be solved by joint force of device and application designers. A cross-layer optimization mentality should be included in the digital circuit and system development. This called for compound knowledge from the device to the

application during design.

7.3 Future work

Cross-layer optimization for quality and power remains a hot research domain. However, it should not stay in the research domain, yet actual industrial applications and deployments are anticipated. Fortunately, an early adoption of these techniques has been observed in the industry. The author firmly believes that the cross-layer methodology will become crucial to continue increasing the performance per watt for future digital circuits and systems. The following is a suggestion for future work in this area.

- **Extending the error effects model to non-uniform distributed errors.** The circuit-level model in this thesis covers the effects when errors are randomly generated. This is readily applicable to SEU effects where the error generation is uniform. For other non-uniform errors, especially time-dependent degradations, the toggling frequency, and operating voltage information, should never be dismissed. These non-uniformed error possibilities require special consideration not only in the error generation, but also error propagation. The significance factor can be extended to vulnerability factor, which is a product of the error generation possibility and error consequences. For instance, [Mukherjee, 2003] proposes an architecture vulnerability factor, which is the product of the error generation possibility and the error propagation possibility (but not error severity).
- **Complete hardware prototyping and system-level consideration of timing-error tolerant Massive MIMO systems.** The timing-error monitor on hardware is validated on recursive application CORDIC, and on non-recursive digital front-end. A complete system-level prototyping is possible. A key missing part is an integration with on-chip power regulators for autonomous voltage tuning. In addition, the proof of concept system-level demonstration, built with multiple chips, will allow consideration of realistic variations in environmental conditions. The considering of errors in uplink and channel estimations can also be researched. If that is considered, the co-optimization of digital front-end and channel pre-coder is much needed. It is believed by the author that the MMSE pre-coder might suppress the digital distortion errors more effectively than the ZF pre-coder. The increased channel estimation and pre-coding complexity should be carefully checked.

- **Considering workload-related CMOS aging effects.** Although the methods in this thesis can cope with the slow-changing CMOS aging effects, they cannot fully exploit the workload-related CMOS aging. Modern CMOS device is allocating more safety margins to the aging effects [Stamoulis, 2016]. Therefore, designing circuits with workload-related aging models will promote new proposals to exploit these margins, and saves power consumption.
- **Approximate computing.** The hardware switch in this work is only activated when no arithmetic errors will be produced. This can extend to situations when some small amount of errors occurs. For instance, when the input is slightly larger than the short multiplier size, saturation or truncation can be performed. This, however, needs compiler interplay to ensure application-level correctness. Moreover, this thesis has not covered the approximate computing hardware. An approximate computing device produces errors by design, which simplifies the computation process in return. The comparisons between the VOS and approximate multiplier [Liu, 2014] can be further investigated. The VOS and approximate hardware can also work together, its impact on application-level quality should be studied.
- **Stochastic computing.** Known for its low-complexity, stochastic computing devices represents and processes information in the form of digitized probabilities. However, it was seen as impractical because of very long computation times and relatively low accuracy. However, if future technologies continue to increase uncertainty in circuit behavior, it will imply a need to better understand, and perhaps exploit, probability in computation.
- **Systematical cross-layer design methodology.** This thesis performed ad-hoc cross-layer optimizations to digital circuits and systems, which spot and optimize the power / quality bottlenecks. Ideally, a more systematical cross-layer design methodology is welcomed. This task is not easy, as changing the whole design flow requires huge interplay from IC foundries to EDA vendors, designers, and system integrator. However, if the current design flow, combined with cross-layer optimizations, can not sustain the development of digital technologies, the systematical methodology is definitely one of the most promising solutions.
- **Impacts on future semiconductor devices.** The device uncertainties and management techniques in sub-5nm novel device architecture (e.g. gate-all-around transistors, carbon nanotubes), novel materials (e.g. GaN, GaAs), novel integration (3-D chip and package) and memory technologies (e.g. M-RAM, R-RAM) can be studied. Technology engineering is trying

to enhance the reliability of those new technologies. However, the progress is not always satisfying. The new device parameters in variations might demand new design methodologies.

- **Applications to other soft-output systems.** The case-study in this thesis is around the wireless communication systems. In these systems, the quality of service is never strict, as long as SNR, BER, throughput requirements (among others) are fulfilled. The knowledge gained in this thesis can also apply to the power and quality trade-offs in other soft quality systems, e.g., heuristic searching problems, approximate simulation of supercomputing tasks, and training and inference in artificial intelligence applications. In the deeply scaled semiconductor era, the advance depends more and more on the application. The digital neural network is well believed to the next killer application. The convolutional neural network, especially in computer vision applications, does not require precise solutions. The data-path error resilience power minimization can be substantial, considering the massive amount of processing carried out.

Appendix A

A digital front-end processor for 60 GHz polar transmitter

Abstract

A complete Digital Front-End (DFE) processor for the 60 GHz polar transmitter is presented. It avoids supply modulating, RF limiters, and AM detection circuits, compared to traditional analog-centric polar transmitter architectures.

The front-end processor consists of i) a poly-phase Cascaded Integrator-Comb (CIC) filter for spectrum shaping, ii) parallel COordinate Rotation DIGital Computers (CORDIC) for rectangular-to-polar conversion, and iii) Power Amplifier (PA) non-linearities pre-distortion units using Look-Up Tables (LUTs). It is designed in the two-phase latch-based pipeline to achieve a throughput of 4x1.76 Gbps. Implemented in a standard 28nm CMOS technology, the DFE processor occupies 0.031 mm^2 and consumes 39mW from 0.9V supply. This result outperforms previously reported architectures.

A.1 Introduction

In contrast with the scarcely available spectrum in the sub-10 GHz range, the 60 GHz frequency band provides 4 channels of 2.1 GHz bandwidth each, as specified by the IEEE802.11ad standard [Association, 2002]. This provides up to

6.75 Gbps data rate in Wireless Personal Area Network (WPAN). [Association, 2002]

However, due to the high free-space path loss, transmission at 60 GHz covers much less distance for a given power budget. This can be alleviated by employing phased array antennas. [Khalaf, 2016] Nevertheless, as the number of Power Amplifiers (PAs) increases, the power consumption grows drastically. The power issue is more severe given the fact that, the 60 GHz class-A linear mode PAs usually provide less than 5% efficiency.

Therefore, the polar architecture is proposed, which allows the PA to operate in the saturation region. In a polar transmission, the PHase (PH) and the AMplitude (AM) signals have separate paths before being combined by the PA. Conventional analog-centric polar modulation scheme suffers from several challenges, e.g. supply voltage linearity of the PA, AM-AM distortion, AM-PM distortion, nonlinearity of the envelope detector, and AM-PM distortion.

To cope with that, a digital-intensive transmitter architecture with the polar concept is explored at mm-waves high-bandwidth transmitters. Moreover, the polar concept is expanded to the whole transmitter, rather than only in the RF domain. The AM signal can then digitally modulate a variable-size PA. This avoids modulating the supply and also eliminates the need for an additional RF limiter and AM detection circuits, which would introduce extra nonlinearity and bandwidth limitations. Despite many advantages, the design of the digital front-end (DFE) processor is very challenging. For the 60 GHz application, the DFE processor mostly needed to work at a very high speed depending on the required oversampling factor. [Li, 2015a] discussed several design considerations for the polar conversion unit, without implementation.

This work presents the first DFE processor for such polar transmitter working in the 60 GHz band. It enables high-bandwidth data transmission with an output throughput of 7.04 Gbps (4x1.74 Gbps). The extensive measurement confirms the great potential of the polar architecture in an actual design. Section II discusses the system architecture. The implementation details are illustrated in Section III. In Section IV, the chip measurement results are presented.

A.2 System architecture

Fig. A.1 shows the high-level architecture of a polar transmitter system. The system consists of a DFE processor (which is presented in this work) and an analog front-end (described in details in [Khalaf, 2016]). The DFE comprises DSP for upsampling the rectangular (I & Q) signals, for I & Q signals to

AM and PH signals conversion, and for pre-distortion compensation. The 802.11ad standard specifies -21 dB EVM for single carrier QAM-16 modulation. Considering the variations in this deeply scaled 28nm CMOS technology, -31dB is taken as the design goal with a design margin of 10 dB.

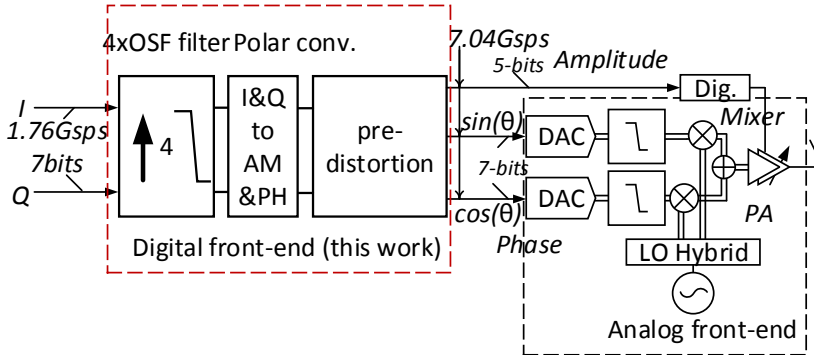


Figure A.1: Block diagram of the digital intensive 60 GHz polar transmitter.

A.2.1 Polar conversion

The rectangular to polar conversion takes in-phase I and quadrature Q signals, and provides the corresponding AM (A) and PH (in the form of $\sin(\theta(t))$ and $\cos(\theta(t))$):

$$\begin{aligned}
 A(t) &= \sqrt{I(t)^2 + Q(t)^2} \\
 \sin(\theta(t)) &= \sin(\arctan(\frac{Q(t)}{I(t)})) \\
 \cos(\theta(t)) &= \cos(\arctan(\frac{Q(t)}{I(t)}))
 \end{aligned} \tag{A.1}$$

This conversion involves multiple complex computations, e.g., square root, trigonometric and division computations. With the aim of energy efficient processing, this is achieved by deep-pipelined COordinate Rotation Digital Computers (CORDIC) [Li, 2015a]. Each CORDIC rotates the vector of the I

& Q signals iteratively until the vector angle reaches zero. The resulting vector amplitude and the rotated angle are recorded as the AM and PH signals.

A.2.2 Phase shaping filter

The polar conversion is a nonlinear computation, which broadens the spectrum. To avoid error vector magnitude (EVM) degradation from the spectrum overlap, signals are oversampled and digitally filtered before aliasing. Another reason for oversampling is to overcome alias generated by the RF-DAC in the analog stage. [Van Zeijl, 2007] To suppress the alias below the spectrum mask, a 4xOSF is investigated in [Li, 2015a], in combination with an analog Butterworth baseband filter in the PH path.

For pulse shaping, we utilized the Cascaded Integrator-Comb (CIC) filter, rather than the computation-intensive raised cosine filter. The structure and the transfer function of the CIC filter are illustrated in Fig. A.2. The oversampling is performed after the CIC filter, rather than before it, to reduce the operating frequency of the CIC filter.

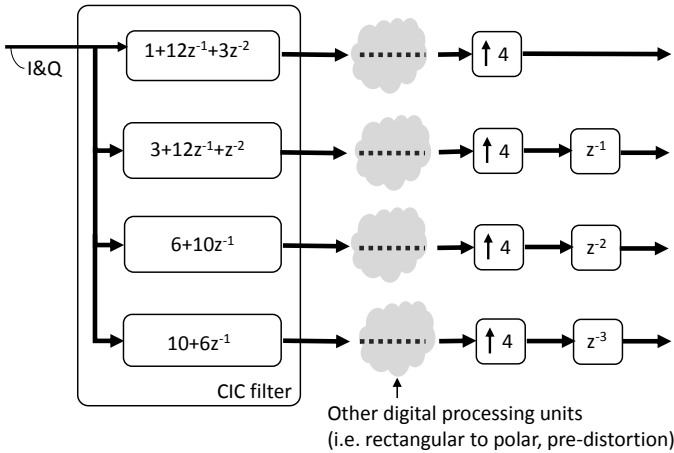


Figure A.2: Poly-phase implementation of the phase shaping filter.

With a targeted EVM of -30 dB, the PH and AM resolutions for the combination of the phase shaping filter and the polar conversion are traded-off in Matlab (Fig. A.3), which decides 5 bits for AM and 7 bits for PH.

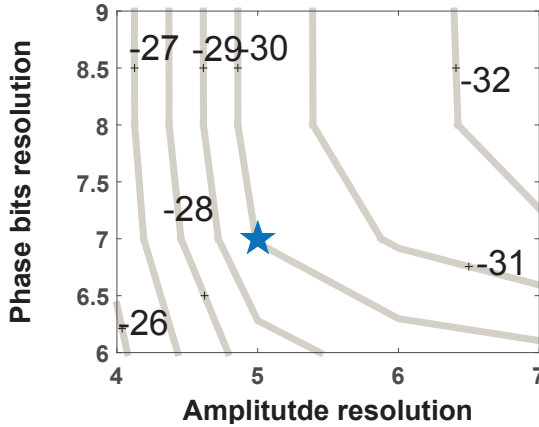


Figure A.3: EVM (in dB) vs. PH/AM resolution (in # of bits).

A.2.3 Pre-distortion

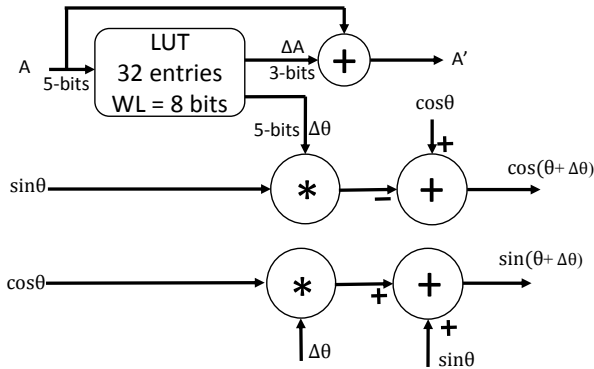


Figure A.4: Pre-distortion circuits for PA non-linearities.

The DSP output signals are distorted by the analog processing functions. This is due to i) nonidealities such as bandwidth limitations of the analog components in the amplitude and phase paths, ii) delay mismatch between the amplitude and phase paths, and iii) the RF-DAC non-idealities. These distortions cause spectral regrowth and devastate the constellation diagram. Therefore, a Pre-Distortion (PD) circuit is provided.

As shown in Fig. A.4, the pre-distortion unit is built with a lookup table (LUT), where the AM serves as the addressing index. The AM and PH signals are compensated with the derived Δ values from the LUT:

$$\begin{aligned}
 A(t)' &= A(t) + \Delta A \\
 \sin(\theta'(t)) &= \sin(\theta(t) + \Delta\theta(t)) \\
 &= \sin(\theta(t))\cos(\Delta\theta(t)) + \cos(\theta(t))\sin(\Delta\theta(t)) \\
 &\sim \sin(\theta(t)) + \cos(\theta(t))\Delta\theta(t) \\
 \cos(\theta'(t)) &= \cos(\theta(t) + \Delta\theta(t)) \\
 &= \cos(\theta(t))\cos(\Delta\theta(t)) - \sin(\theta(t))\sin(\Delta\theta(t)) \\
 &\sim \cos(\theta(t)) - \sin(\theta(t))\Delta\theta(t)
 \end{aligned} \tag{A.2}$$

The pre-distorted AM is created by summing up the AM and the derived $\Delta A(t)$ from the LUT. Similarly, the pre-distorted PH signals are obtained by operating on respective PH signals with and the $\Delta\theta(t)$. To avoid heavy computations, the PH pre-distortion is approximated in Equation A.2, provided that the $\Delta\theta(t)$ is small. An LUT consists of 32 entries (because the AM signal is 5-bit width). Each entry is of 8 bits width. The 3 most significant bits are assigned to the ΔA , and the 5 least significant bits to the $\Delta\theta(t)$.

A.3 Implementation details

Fig. A.5 shows the overall pipeline scheme. The DFE input throughput is 1.76 Gsps. The system comprises 4 parallel pipelined signal paths (each has a separate CIC filter, a CORDIC, and a pre-distortion unit), as the OSF is 4. Therefore, the output throughput is 7.04 Gsps (4x1.76 Gsps). The speed requirement is challenging, even with the 4x parallelism. Therefore, the DFE processor was implemented with a deep pipeline structure, i.e. pipelined after each addition.

Since the coefficients of the CIC filters are pre-determined, the multiplications in those filters were accomplished by shift-additions for power saving. Moreover, signals are added by customized carry-save adders and finally adding up by the vector merging adders (adding up the vectors of sums and carries) [Huang, 2015]. The pipeline breaks the CORDIC after each CORDIC rotation. As the AM

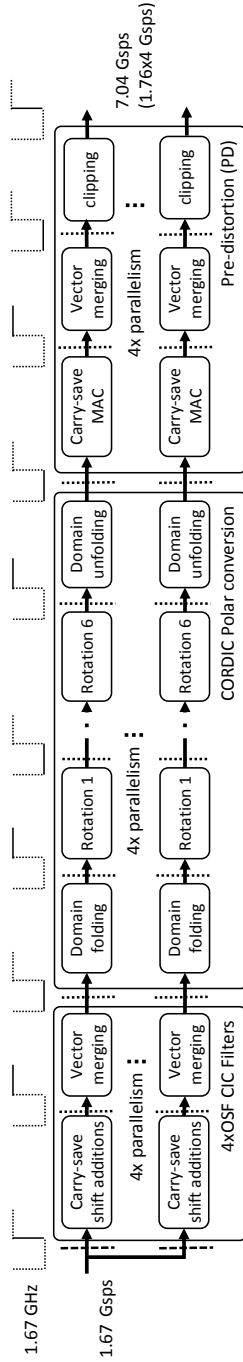


Figure A.5: Pipeline scheme of the proposed DFE processor.

resolution (5-bit) is less than the PH resolution (7-bit), the AM signals require fewer rotations. Therefore, they are ready before PH signals are produced. This is advantageous for the pre-distortion: by the time the PH signals are computed, the ΔA and the $\Delta\theta$ are already fetched by the AM signals from the LUT. The LUT is implemented with single-port RAMs.

To reduce the power consumption, level-triggered two-phase latches were chosen as the sequential component. The input sequential elements are rising-edge enabled flip-flops. The enabling signals for the latches are indicated by a solid line. For instance, the first latch in the pipeline is active-high, while the second latch is active-low. This complementary two-phase latch methodology eliminates often-encountered hold time problem in latch based designs. Advantageously, the proposed latch scheme allows time borrowing. For instance, the data can arrive later than the rising edge for an active-high latch, and thus borrows time from the next pipeline stage. The advantage of time-borrowing is two-fold. Firstly, it eases timing closure, because it can perform stage balancing automatically. This eliminates manually moving computation and/or logic elements from one stage to another. For the exemplified 1.76 GHz high-speed, it is especially beneficial. Secondly, the opportunistic time-borrowing principle addresses process and environmental variations. Due to such variations, even if the pipeline is carefully equalized at design time, the delay of each computation stage can vary in the fabricated chip, the effect of which becomes even more severe with technology scaling. In the DFE, time-borrowing allows for a slower computation stage to opportunistically borrow time from faster ones, which averages out some of the variations.

Even with the techniques mentioned above, the speed requirement cannot be achieved for the standard 28nm technology, with 0.9V as the standard V_{dd} . Accordingly, we applied the following modifications during library selection. i) For the CIC filter and the CORDIC polar converter: we utilized fast but leaky Low V_{th} (LVT) cells, rather than the Standard V_{th} (SVT). This leakage increase should not change the overall power consumption, as the circuit power is dominated by the dynamical part (since the clock frequency is very high). Moreover, the designed V_{dd} was increased to 1V for a higher speed. ii) For the pre-distortion unit: as the setup-timing requirement is not as difficult as the rest units, they were accomplished by SVT cells under 0.9V.

Therefore, we divided the design into two power domains: 1V for the CIC filter and the CORDIC polar conversion, and 0.9V for the pre-distortion unit. The pre-distortion unit can be switched off and by-passed for scenarios where

linearity is sufficient, for energy saving.

A.4 Measurement results

The DFE processor was processed in a standard 28 nm CMOS technology. A micrograph of the chip is shown in Fig. A.6. The complete design area is as small as 0.036mm^2 , of which the switchable pre-distortion unit utilizes 0.015mm^2 . The cell area breakdown is illustrated in Fig. A.7.

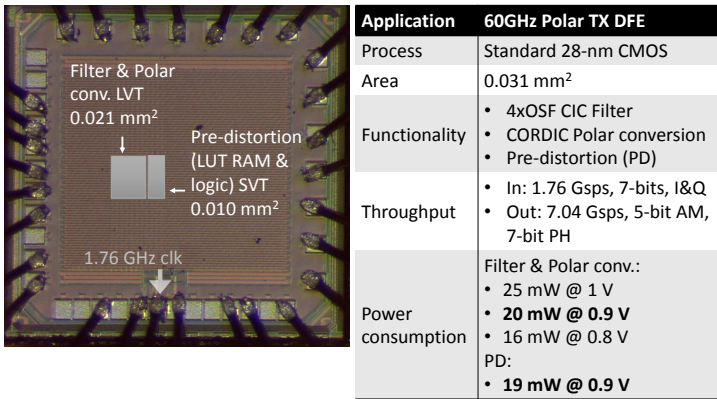


Figure A.6: Die photo and chip information of the proposed digital polar DFE processor.

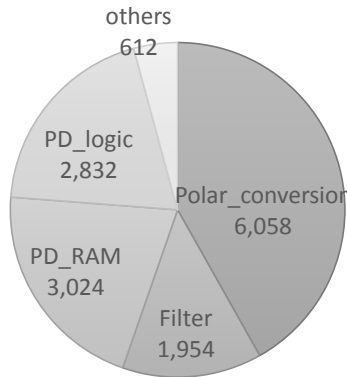


Figure A.7: Cell area breakdown (in μm^2) of the proposed DFE processor. N.B.: Routing spacing not included.

The circuit speed was measured for a typical-case chip at 25°C room temperature (Fig. A.8). Due to the inserted design margin, although the chip is designed to be 1.76 GHz @ 1V, a typical chip can operate correctly @ 3 GHz with the same V_{dd} , implying a throughput of 4x3 GHz. Alternately, it can also operate @ 0.8V V_{dd} with a fixed frequency of 1.76 GHz, which brings 33% power saving. The CIC filter and the CORDIC polar conversion consume 25mW @ 1V V_{dd} , 20mW @ 0.9V, or 16mW @ 0.8V. The pre-distortion unit consumes 19mW @ 0.9V. The overall leakage consumption power is less than 1mW at room temperature.

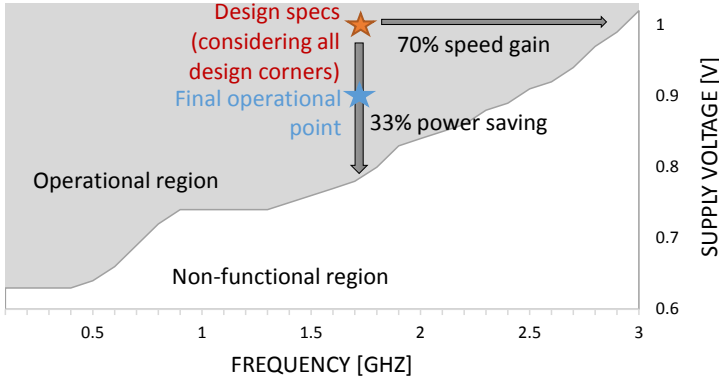


Figure A.8: Typical chip speed vs. V_{dd} @ 25°C .

The power spectrum density (PSD) of the DFE outputs is shown in Fig. A.9. Both the in-band PSD and the alias rejection are confirmed to be compliant with the spectrum mask. The EVM of the produced signal is measured to be -30.5 dB. Fig. A.10 plots the constellation for 16-QAM signals. It demonstrated clearly the nice purity of the signals.

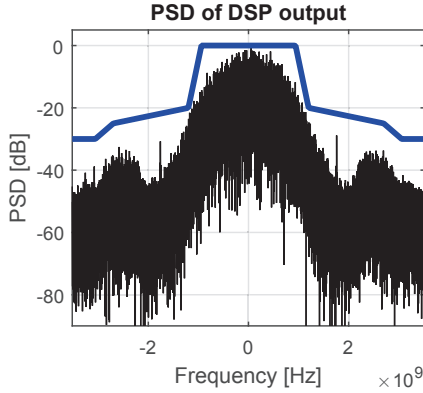


Figure A.9: Measurement DFE outputs PSD.

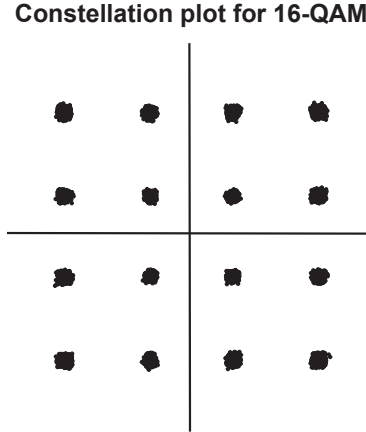


Figure A.10: Measurement DFE output 16-QAM constellation, with a corresponding EVM of -30.5 dB.

As this work is the first polar transmitter for high bandwidth 60 GHz system, benchmarking can be difficult. Nevertheless, Table A.1 compares state-of-the-art digital signal processors [Hwang, 2003; Strollo, 2008; Muller, 2012; Mehta, 2010] that has similar functionalities. The only work with comparable speed is [Muller, 2012], where merely the 4xOSF filtering function is provided. Even with the perfect scaling normalization, the only state-of-the-art work with comparable energy consumption is [Strollo, 2008], which only performs

Table A.1: Performance comparisons of digital polar front-end systems.

	Process	Input Bit Width	Supply	Functions	Output Throughput (Msps)	Power Cons. (mW)	¹ Energy (pJ/bit)	² E _{norm} (pJ/bit)
[Hwang, 2003]	250	14	2.5V	CORDIC	406	470	83	1.2
[Strollo, 2008]	250	13	2.5V	CORDIC	430	276	49	0.72
[Muller, 2012]	65	7	1.4V	4xOSF Filter	4x2400	400	6.0	1.1
[Mehta, 2010]	65	11	1.2V	CORDIC PD	125	³ 5	3.6	0.88
				32xOSF Filter	32x125	³ 121	2.7	0.66
This work	28	7	0.9V	4xOSF Filter	4x1760 (Max: 4x3000@1V)	20	0.41	0.41
				CORDIC PD		19	0.38	0.38

¹ Energy = Power / Frequency / Bits.

² E_{norm} = Energy x (V_{dd}/1V)² x (tech/28nm).

³ Estimated from a total current of 105mA.

CORDIC polar conversion. In summary, the comparisons show the proposed design has significant advantages.

A.5 Conclusions

This paper presents the first DFE processor for polar transmitter working in the 60 GHz band. It enables digital-intensive transmitter architecture with polar concept expanded to the whole transmitter, rather than conventionally only in radio frequency domain.

The DFE processor was processed in a standard 28 nm technology with 0.036 mm² area. The processor provides -30.5 dB EVM, with 4x1.76 Gsps output throughput. The throughput can reach to 4x3 Gsps when 1V V_{dd} is supplied. It consumes 39mW from 0.9V supply.

Bibliography

- [Abdallah, 2009] R. A. Abdallah and N. R. Shanbhag. “Error-resilient low-power viterbi decoder architectures”. In: *IEEE Transactions on Signal Processing* 57.12 (2009), pp. 4906–4917. ISSN: 1053587X. DOI: [10.1109/TSP.2009.2026078](https://doi.org/10.1109/TSP.2009.2026078).
- [Andrae, 2015] A. S. G. Andrae and T. Edler. “On global electricity usage of communication technology: trends to 2030”. In: *Challenges* 6.1 (2015), pp. 117–157.
- [Andrews, 2014] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang. “What will 5G be?” In: *IEEE Journal on selected areas in communications* 32.6 (2014), pp. 1065–1082.
- [Asenov, 2007] A. Asenov. “Simulation of statistical variability in nano MOSFETs”. In: *VLSI Technology, 2007 IEEE Symposium on*. IEEE, 2007, pp. 86–87.
- [Association, 2002] I. S. Association et al. *IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer*. IEEE, 2002.
- [Austin, 2008] T. Austin, V. Bertacco, S. Mahlke, and Y. Cao. “Reliable systems on unreliable fabrics”. In: *IEEE Design & Test of Computers* 25.4 (2008).
- [Baumann, 2002] R. Baumann. “The impact of technology scaling on soft error rate performance and limits to the efficacy of error correction”. In: *Electron Devices Meeting*,

2002. *IEDM'02. International*. IEEE. 2002, pp. 329–332.
- [Bawden, 2016] T. Bawden. *Global warming: Data centres to consume three times as much energy in next decade, experts warn*. 2016.
- [Best, 2014] J. Best. “The Race to 5G: Inside the Fight for the Future of Mobile as we Know it”. In: *San Francisco (CA, USA): Techrepublic-CBS Interactive*. [consult: 10/07/2015] (2014).
- [Biswas, 2008] A. Biswas, P. Racunas, J. Emer, and S. S. Mukherjee. “Computing accurate AVFs using ACE analysis on performance models: A rebuttal”. In: *IEEE Computer Architecture Letters* 7.1 (2008), pp. 21–24. ISSN: 15566056. DOI: [10.1109/L-CA.2007.19](https://doi.org/10.1109/L-CA.2007.19).
- [Björnson, 2014] E. Björnson, J. Hoydis, M. Kountouris, and M. Debbah. “Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits”. In: *IEEE Transactions on Information Theory* 60.11 (2014), pp. 7112–7139.
- [Bolchini, 2007] C. Bolchini, A. Miele, and M. D. Santambrogio. “TMR and Partial Dynamic Reconfiguration to mitigate SEU faults in FPGAs”. In: *IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems* (Sept. 2007), pp. 87–95. DOI: [10.1109/DFT.2007.25](https://doi.org/10.1109/DFT.2007.25).
- [Boning, 1996] D. S. Boning and J. E. Chung. “Statistical metrology: Understanding spatial variation in semiconductor manufacturing”. In: *Microelectronic Manufacturing 1996*. International Society for Optics and Photonics. 1996, pp. 16–26.
- [Bowman, 2002] K. A. Bowman, S. G. Duvall, and J. D. Meindl. “Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration”. In: *IEEE Journal of solid-state circuits* 37.2 (2002), pp. 183–190.
- [Bowman, 2009a] K. A. Bowman, A. R. Alameldeen, S. T. Srinivasan, and C. Wilkerson. “Impact of Die-to-Die and Within-Die Parameter Variations on the Clock Frequency and Throughput of Multi-Core Processors.” In: *IEEE Trans. VLSI Syst.* 17.12 (2009), pp. 1679–1690.

- [Bowman, 2009b] K. A. Bowman, J. W. Tschanz, N. S. Kim, J. C. Lee, C. B. Wilkerson, S. L. Lu, T. Karnik, and V. K. De. “Energy-efficient and metastability-immune resilient circuits for dynamic variation tolerance”. In: *Solid-State Circuits, IEEE Journal of* 44.1 (2009), pp. 49–63.
- [Bowman, 2011] K. A. Bowman, J. W. Tschanz, S. L. L. Lu, P. A. Aseron, M. M. Khellah, A. Raychowdhury, B. M. Geuskens, C. Tokunaga, C. B. Wilkerson, T. Karnik, and V. K. De. “A 45 nm Resilient Microprocessor Core for Dynamic Variation Tolerance”. In: *IEEE Journal of Solid-State Circuits* 46.1 (Jan. 2011), pp. 194–208. ISSN: 0018-9200. DOI: [10.1109/JSSC.2010.2089657](https://doi.org/10.1109/JSSC.2010.2089657).
- [Bowman, 2016] K. A. Bowman, S. Raina, J. T. Bridges, D. J. Yingling, H. H. Nguyen, B. R. Appel, Y. N. Kolla, J. Jeong, F. I. Atallah, and D. W. Hansquine. “A 16 nm all-digital auto-calibrating adaptive clock distribution for supply voltage droop tolerance across a wide operating range”. In: *IEEE Journal of Solid-State Circuits* 51.1 (2016), pp. 8–17.
- [Bravos, 2005] G. Bravos and A. G. Kanatas. “Energy consumption and trade-offs on wireless sensor networks”. In: *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*. Vol. 2. IEEE. 2005, pp. 1279–1283.
- [Brglez, 1989] F. Brglez, D. Bryan, and K. Kozminski. “Combinational profiles of sequential benchmark circuits”. In: *IEEE International Symposium on Circuits and Systems*. 217. 1989, pp. 1929–1934. DOI: [10.1109/ISCAS.1989.100747](https://doi.org/10.1109/ISCAS.1989.100747).
- [Bull, 2011] D. Bull, S. Das, K. Shivashankar, G. S. Dasika, K. Flautner, and D. Blaauw. “A power-efficient 32 bit ARM processor using timing-error detection and correction for transient-error tolerance and adaptation to PVT variation”. In: *Solid-State Circuits, IEEE Journal of* 46.1 (2011), pp. 18–31.
- [Burd, 2000] T. D. Burd, T. A. Pering, A. J. Stratakos, and R. W. Brodersen. “A dynamic voltage scaled microprocessor system”. In: *IEEE Journal of solid-state circuits* 35.11 (2000), pp. 1571–1580.

- [Calhoun, 2004] B. H. Calhoun and A. P. Chandrakasan. “Standby power reduction using dynamic voltage scaling and canary flip-flop structures”. In: *Solid-State Circuits, IEEE Journal of* 39.9 (Sept. 2004), pp. 1504–1511. ISSN: 0018-9200. DOI: [10.1109/JSSC.2004.831432](https://doi.org/10.1109/JSSC.2004.831432).
- [Calhoun, 2006] B. H. Calhoun and A. P. Chandrakasan. “Ultra-dynamic voltage scaling (UDVS) using sub-threshold operation and local voltage dithering”. In: *IEEE Journal of Solid-State Circuits* 41.1 (2006), pp. 238–245.
- [Carter, 2010] N. P. Carter, H. Naeimi, and D. S. Gardner. “Design techniques for cross-layer resilience”. In: *Proceedings of the Conference on Design, Automation and Test in Europe*. European Design and Automation Association. 2010, pp. 1023–1028.
- [Catthoor, 2010] F. Catthoor, P. Raghavan, A. Lambrechts, M. Jayapala, E. A. Kritikakou, and J. Absar. “Exploiting Word-Width Information During Mapping”. In: *Ultra-Low Energy Domain-Specific Instruction-Set Processors*. Springer, 2010, pp. 223–273.
- [Chang, 2012] H. Chang, V. Gerousis, S. Molakalapalli, and S. Shrivastava. *Design-specific on chip variation de-rating factors for static timing analysis of integrated circuits*. 2012.
- [Chaparro, 2007] P. Chaparro, J. González, G. Magklis, Q. Cai, and A. González. “Understanding the thermal implications of multi-core architectures”. In: *IEEE Transactions on Parallel and Distributed Systems* 18.8 (2007).
- [Chen, 2014] C.-C. Chen, S. Cha, T. Liu, and L. Milor. “System-level modeling of microprocessor reliability degradation due to BTI and HCI”. In: *Reliability Physics Symposium, 2014 IEEE International*. IEEE. 2014, CA–8.
- [Chiang, 2007] C. Chiang and J. Kawa. *Design for manufacturability and yield for nano-scale CMOS*. Springer Science & Business Media, 2007.
- [Choudhury, 2010] M. Choudhury, V. Chandra, K. Mohanram, and R. Aitken. “TIMBER: Time borrowing and error relaying for online timing error resilience”. In: *Design, Automation Test in Europe Conference Exhibition*

- (DATE), 2010. Mar. 2010, pp. 1554–1559. DOI: [10.1109/DATE.2010.5457058](https://doi.org/10.1109/DATE.2010.5457058).
- [Clark, 1995] J. A. Clark. *Fault Injection: A method for validating computer system dependability*. 1995.
- [Constantin, 2015] J. Constantin, L. Wang, G. Karakonstantis, A. Chattopadhyay, and A. Burg. “Exploiting dynamic timing margins in microprocessors for frequency-over-scaling with instruction-based clock adjustment”. In: *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015*. IEEE. 2015, pp. 381–386.
- [Corno, 2000] F. Corno, M. S. Reorda, and G. Squillero. “RT-level ITC’99 benchmarks and first ATPG results”. In: *IEEE Design & Test of Computers* 17.3 (2000), pp. 44–53. ISSN: 07407475. DOI: [10.1109/54.867894](https://doi.org/10.1109/54.867894).
- [Dai, 2001] W. Dai and H. Ji. “Timing analysis taking into account interconnect process variation”. In: *Statistical Methodology, IEEE International Workshop on, 2001 6yh*. IEEE. 2001, pp. 51–53.
- [Das, 2006] S. Das, D. Roberts, S. Lee, S. Pant, D. Blaauw, T. Austin, K. Flautner, and T. Mudge. “A self-tuning DVS processor using delay-error detection and correction”. In: *IEEE Journal of Solid-State Circuits* 41.4 (2006), pp. 792–804.
- [Dennard, 1974] R. H. Dennard, F. H. Gaensslen, V. L. Rideout, E. Bassous, and A. R. LeBlanc. “Design of ion-implanted MOSFET’s with very small physical dimensions”. In: *IEEE Journal of Solid-State Circuits* 9.5 (1974), pp. 256–268.
- [Desset, 2014] C. Desset, B. Debaillie, and F. Louagie. “Modeling the hardware power consumption of large scale antenna systems”. In: *Green Communications (OnlineGreenComm), 2014 IEEE Online Conference on*. Nov. 2014, pp. 1–6. DOI: [10.1109/OnlineGreenCom.2014.7114430](https://doi.org/10.1109/OnlineGreenCom.2014.7114430).
- [Desset, 2015] C. Desset and L. V. der Perre. “Validation of low-accuracy quantization in massive MIMO and constellation EVM analysis”. In: *Networks and Communications (EuCNC), 2015 European Conference on*. June 2015, pp. 21–25. DOI: [10.1109/EuCNC.2015.7194033](https://doi.org/10.1109/EuCNC.2015.7194033).

- [Dietel, 2014] S. Dietel, S. Hoppner, T. Brauninger, U. Fiedler, H. Eisenreich, G. Ellguth, S. Hanzsche, S. Henker, and R. Schüffny. “A compact on-chip IR-drop measurement system in 28 nm CMOS technology”. In: *Circuits and Systems (ISCAS), 2014 IEEE International Symposium on*. IEEE. 2014, pp. 1219–1222.
- [Djahromi, 2007] A. K. Djahromi, A. M. Eltawil, F. J. Kurdahi, and R. Kanj. “Cross layer error exploitation for aggressive voltage scaling”. In: *Quality Electronic Design, 2007. ISQED’07. 8th International Symposium on*. IEEE. 2007, pp. 192–197.
- [Dreslinski, 2010] R. G. Dreslinski, M. Wiecekowsi, D. Blaauw, D. Sylvester, and T. Mudge. “Near-threshold computing: Reclaiming moore’s law through energy efficient integrated circuits”. In: *Proceedings of the IEEE* 98.2 (2010), pp. 253–266.
- [Driscoll, 2003] K. Driscoll, B. Hall, H. Sivencrona, and P. Zumsteg. “Byzantine Fault Tolerance , from Theory to Reality 1 What You Thought Could Never Happen”. In: *Thought A Review Of Culture And Idea* 2 (2003), pp. 235–248. DOI: [10 . 1007 / 978 - 3 - 540 - 39878 - 3_19](https://doi.org/10.1007/978-3-540-39878-3_19).
- [Ebrahimi, 2015] M. Ebrahimi, A. Evans, M. B. Tahoori, E. Costenaro, D. Alexandrescu, V. Chandra, and R. Seyyedi. “Comprehensive analysis of sequential and combinational soft errors in an embedded processor”. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34.10 (2015), pp. 1586–1599.
- [EEMBC, 2006] EEMBC. *CoreMark, an EEMBC Benchmark. CoreMark scores for embedded and desktop CPUs*. 2006.
- [Ejlali, 2007] A. Ejlali, B. M. Al-Hashimi, P. Rosinger, and S. G. Miremadi. “Joint consideration of fault-tolerance, energy-efficiency and performance in on-chip networks”. In: *Proceedings -Design, Automation and Test in Europe, DATE*. 2007, pp. 1647–1652. ISBN: 3981080122. DOI: [10 . 1109 / DATE . 2007 . 364538](https://doi.org/10.1109/DATE.2007.364538).
- [Elgebaly, 2007] M. Elgebaly and M. Sachdev. “Variation-aware adaptive voltage scaling system”. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 15.5 (2007), pp. 560–571.

- [EMarketer, 2017] EMarketer. *US Adults Now Spend 12 Hours 7 Minutes a Day Consuming Media*. 2017.
- [Ernst, 2003] D. Ernst, N. S. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner, and T. Mudge. “Razor: a low-power pipeline based on circuit-level timing speculation”. In: *Microarchitecture, 2003. MICRO-36. IEEE/ACM International Symposium on*. Dec. 2003, pp. 7–18. DOI: [10.1109/MICRO.2003.1253179](https://doi.org/10.1109/MICRO.2003.1253179).
- [Esmaeilzadeh, 2011] H. Esmaeilzadeh, E. Blem, R. St Amant, K. Sankaralingam, and D. Burger. “Dark silicon and the end of multicore scaling”. In: *ACM SIGARCH Computer Architecture News*. Vol. 39. 3. ACM. 2011, pp. 365–376.
- [Evans, 2011] D. Evans. “The internet of things: How the next evolution of the internet is changing everything”. In: *CISCO white paper 1.2011* (2011), pp. 1–11.
- [Feng, 2010] S. Feng, S. Gupta, A. Ansari, and S. Mahlke. “Shoestring: probabilistic soft error reliability on the cheap”. In: *ACM SIGARCH Computer Architecture News*. Vol. 38. 1. ACM. 2010, pp. 385–396.
- [Fojtik, 2012] M. Fojtik, D. Fick, Y. Kim, N. Pinckney, D. Harris, D. Blaauw, and D. Sylvester. “Bubble Razor: An architecture-independent approach to timing-error detection and correction”. In: *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*. IEEE. 2012, pp. 488–490.
- [Fojtik, 2013] M. Fojtik, D. Fick, Y. Kim, N. Pinckney, D. M. Harris, D. Blaauw, and D. Sylvester. “Bubble Razor: Eliminating Timing Margins in an ARM Cortex-M3 Processor in 45 nm CMOS Using Architecturally Independent Error Detection and Correction”. In: *IEEE Journal of Solid-State Circuits* 48.1 (Jan. 2013), pp. 66–81. ISSN: 0018-9200. DOI: [10.1109/JSSC.2012.2220912](https://doi.org/10.1109/JSSC.2012.2220912).
- [Fujiwara, 1985] H. Fujiwara. *Logic testing and design for testability*. MIT press Cambridge, MA, 1985.

- [Gaisler, 2002] J. Gaisler. “A portable and fault-tolerant microprocessor based on the SPARC V8 architecture”. In: *Proceedings of the 2002 International Conference on Dependable Systems and Networks*. 2002, pp. 409–415. ISBN: 0769515975. DOI: [10 . 1109 / DSN . 2002 . 1028926](https://doi.org/10.1109/DSN.2002.1028926).
- [Geerts, 2006] Y. Geerts, M. Steyaert, and W. M. C. Sansen. *Design of multi-bit delta-sigma A/D converters*. Vol. 686. Springer Science & Business Media, 2006.
- [Ghosh, 2010] S. Ghosh and K. Roy. “Parameter variation tolerance and error resiliency: New design paradigm for the nanoscale era”. In: *Proceedings of the IEEE* 98.10 (2010), pp. 1718–1751.
- [GimmlerDumont, 2013] C. Gimmler-Dumont, M. May, and N. Wehn. “Cross-layer error resilience and its application to wireless communication systems”. In: *Journal of Low Power Electronics* 9.1 (2013), pp. 119–132.
- [Gonzalez, 2017] C. Gonzalez, E. Fluhr, D. Dreps, D. Hogenmiller, R. Rao, J. Paredes, M. Floyd, M. Sperling, R. Kruse, V. Ramadurai, et al. “3.1 POWER9™: A processor family optimized for cognitive computing with 25Gb/s accelerator links and 16Gb/s PCIe Gen4”. In: *Solid-State Circuits Conference (ISSCC), 2017 IEEE International*. IEEE. 2017, pp. 50–51.
- [Gunnarsson, 2017] S. Gunnarsson, M. Bortas, Y. Huang, C.-M. Chen, L. Van der Perre, and O. Edfors. “Lousy processing increases energy efficiency in massive MIMO systems”. In: *2017 European Conference on Networks and Communications (EuCNC)*. Oulu, Finland, June 2017, pp. 1–5. DOI: [10.1109/EuCNC.2017.7980739](https://doi.org/10.1109/EuCNC.2017.7980739).
- [Gustavsson, 2014] U. Gustavsson, C. Sánchez-Perez, T. Eriksson, F. Athley, G. Durisi, P. Landin, K. Hausmair, C. Fager, and L. Svensson. “On the impact of hardware impairments on massive MIMO”. In: *Globecom Workshops (GC Wkshps), 2014*. IEEE. 2014, pp. 294–300.
- [Gyselinckx, 2005] B. Gyselinckx, C. Van Hoof, J. Ryckaert, R. F. Yazicioglu, P. Fiorini, and V. Leonov. “Human++: autonomous wireless sensors for body area networks”. In: *Custom Integrated Circuits Conference, 2005. Proceedings of the IEEE 2005*. IEEE. 2005, pp. 13–19.

- [Haghi, 2009] M. Haghi and J. Draper. “The 90 nm Double-DICE storage element to reduce Single-Event upsets”. In: *Midwest Symposium on Circuits and Systems*. 2009, pp. 463–466. ISBN: 9781424444793. DOI: [10.1109/MWSCAS.2009.5236054](https://doi.org/10.1109/MWSCAS.2009.5236054).
- [Hamdioui, 2013] S. Hamdioui, D. Gizopoulos, G. Guido, M. Nicolaidis, A. Grasset, and P. Bonnot. “Reliability challenges of real-time systems in forthcoming technology nodes”. In: *DATE (2013)*, pp. 129–134. ISSN: 15301591. DOI: [10.7873/DATE.2013.040](https://doi.org/10.7873/DATE.2013.040).
- [Han, 2013] J. Han and M. Orshansky. “Approximate computing: An emerging paradigm for energy-efficient design”. In: *Test Symposium (ETS), 2013 18th IEEE European*. May 2013, pp. 1–6. DOI: [10.1109/ETS.2013.6569370](https://doi.org/10.1109/ETS.2013.6569370).
- [Hazucha, 2000] P. Hazucha and C. Svensson. “Impact of CMOS technology scaling on the atmospheric neutron soft error rate”. In: *IEEE Transactions on Nuclear science* 47.6 (2000), pp. 2586–2594.
- [Hegde, 1999a] R. Hegde and N. R. Shanbhag. “Energy-efficient signal processing via algorithmic noise-tolerance”. In: *Low Power Electronics and Design, 1999. Proceedings. 1999 International Symposium on*. Aug. 1999, pp. 30–35.
- [Hegde, 1999b] R. Hegde and N. Shanbhag. “Energy-efficient signal processing via algorithmic noise-tolerance”. In: *... of the 1999 international symposium on Low ... (1999)*.
- [Hegde, 2004] R. Hegde and N. R. Shanbhag. “A voltage overscaled low-power digital filter IC”. In: *Solid-State Circuits, IEEE Journal of* 39.2 (2004), pp. 388–391.
- [Hennessy, 2011] J. L. Hennessy and D. A. Patterson. *Computer architecture: a quantitative approach*. Elsevier, 2011.
- [Herbert, 2009] S. Herbert and D. Marculescu. “Variation-aware dynamic voltage/frequency scaling”. In: *High Performance Computer Architecture, 2009. HPCA 2009. IEEE 15th International Symposium on*. IEEE. 2009, pp. 301–312.

- [Herr, 1986] N. Herr and J. J. Barnes. “Statistical circuit simulation modeling of CMOS VLSI”. In: *IEEE transactions on computer-aided design of integrated circuits and systems* 5.1 (1986), pp. 15–22.
- [Holcomb, 2009] D. Holcomb, W. Li, and S. Seshia. “Design as you see FIT: System-level soft error analysis of sequential circuits”. In: *DATE* (2009).
- [Horvath, 2007] T. Horvath, T. Abdelzaher, K. Skadron, and X. Liu. “Dynamic voltage scaling in multitier web servers with end-to-end delay control”. In: *IEEE Transactions on Computers* 56.4 (2007).
- [Huang, 2014] Y. Huang, M. Li, C. Li, P. Debacker, and L. Van der Perre. “Computation-skip error resilient scheme for recursive CORDIC”. In: *2014 IEEE Workshop on Signal Processing Systems (SiPS)*. Belfast, Northern Ireland: IEEE, Oct. 2014, pp. 1–6. ISBN: 978-1-4799-6588-5. DOI: [10.1109/SiPS.2014.6986061](https://doi.org/10.1109/SiPS.2014.6986061).
- [Huang, 2015] Y. Huang, A. Kapoor, R. Rutten, and J. Pineda de Gyvez. “A 13bits 4.096GHz 45nm CMOS digital decimation filter chain with Carry-Save format numbers”. In: *Microprocessors and Microsystems* 39.8 (Nov. 2015), pp. 869–878. ISSN: 01419331. DOI: [10.1016/j.micpro.2014.11.003](https://doi.org/10.1016/j.micpro.2014.11.003).
- [Huang, 2016a] Y. Huang, C. Li, K. Khalaf, A. Bourdoux, J. Verschuere, Q. Shi, P. Wambacq, S. Polling, W. Dehaene, and L. Van der Perre. “A 28 nm CMOS 7.04 Gbps polar digital front-end processor for 60 GHz transmitter”. In: *2016 IEEE Asian Solid-State Circuits Conference (A-SSCC)*. Toyama, Japan: IEEE, Nov. 2016, pp. 333–336. ISBN: 978-1-5090-3699-8. DOI: [10.1109/ASSCC.2016.7844203](https://doi.org/10.1109/ASSCC.2016.7844203).
- [Huang, 2016b] Y. Huang, C. Li, M. Li, F. Catthoor, L. V. der Perre, and W. Dehaene. “An algorithm to identify cornerstones of digital circuits”. In: *ACM/IEEE Design Automation Conference (DAC), Work-In-Progress*. Austin, U.S., 2016.
- [Huang, 2016c] Y. Huang, C. Li, L. Meng, L. Van der Perre, and W. Dehaene. “Fine-grained hardware switching scheme for power reduction in multiplication”. In: *Electronics Letters* 52.16 (Aug. 2016), pp. 1374–1375. ISSN: 0013-5194. DOI: [10.1049/el.2015.3828](https://doi.org/10.1049/el.2015.3828).

- [Huang, 2016d] Y. Huang, M. Li, C. Li, P. Debacker, and L. Van der Perre. “Computation-skip Error Mitigation Scheme for Power Supply Voltage Scaling in Recursive Applications”. In: *Journal of Signal Processing Systems (JSPS)* 84.3 (Sept. 2016), pp. 413–424. ISSN: 1939-8018. DOI: [10.1007/s11265-015-1096-z](https://doi.org/10.1007/s11265-015-1096-z).
- [Huang, 2017] Y. Huang, C. Desset, A. Bourdoux, W. Dehaene, and L. Van der Perre. “Massive MIMO processing at the semiconductor edge: Exploiting the system and circuit margins for power savings”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New Orleans, U.S.: IEEE, Mar. 2017, pp. 3474–3478. ISBN: 978-1-5090-4117-6. DOI: [10.1109/ICASSP.2017.7952802](https://doi.org/10.1109/ICASSP.2017.7952802).
- [Hwang, 2003] D. D. Hwang, D. Fu, and A. N. Willson Jr. “A 400-MHz processor for the conversion of rectangular to polar coordinates in 0.25-um CMOS”. In: *Solid-State Circuits, IEEE Journal of* 38.10 (2003), pp. 1771–1775.
- [Index, 2016] C. V. N. Index. “Global Mobile Data Traffic Forecast Update, 2015–2020 White Paper”. In: *link: http://goo.gl/yITuVx* (2016).
- [Jahinuzzaman, 2009] S. M. Jahinuzzaman, D. J. Rennie, and M. Sachdev. “A soft error tolerant 10T SRAM bit-cell with differential read capability”. In: *IEEE Transactions on Nuclear Science* 56.6 (2009), pp. 3768–3773.
- [Jeon, 2012] D. Jeon, M. Seok, Z. Zhang, D. Blaauw, and D. Sylvester. “Design methodology for voltage-overscaled ultra-low-power systems”. In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 59.12 (2012), pp. 952–956.
- [Kahng, 2011] A. B. Kahng. “Design-Based “Equivalent Scaling” to the Rescue of Moore’s Law”. In: (2011).
- [Kao, 2002] J. T. Kao, M. Miyazaki, and A. P. Chandrakasan. “A 175-MV multiply-accumulate unit using an adaptive supply voltage and body bias architecture”. In: *Solid-State Circuits, IEEE Journal of* 37.11 (Nov. 2002), pp. 1545–1554. ISSN: 0018-9200. DOI: [10.1109/JSSC.2002.803957](https://doi.org/10.1109/JSSC.2002.803957).

- [Karakonstantis, 2009] G. Karakonstantis, D. Mohapatra, and K. Roy. “System level DSP synthesis using voltage overscaling, unequal error protection & adaptive quality tuning.” In: *2009 IEEE Workshop on Signal Processing Systems (SiPS)*. 2009, pp. 133–138.
- [Karakonstantis, 2012a] G. Karakonstantis, C. Roth, C. Benkeser, and A. Burg. “On the exploitation of the inherent error resilience of wireless systems under unreliable silicon”. In: *Design Automation Conference (DAC), 2012 49th ACM/EDAC/IEEE*. June 2012, pp. 510–515. DOI: [10.1145/2228360.2228451](https://doi.org/10.1145/2228360.2228451).
- [Karakonstantis, 2012b] G. Karakonstantis, C. Roth, A. Burg, and C. Benkeser. “On the exploitation of the inherent error resilience of wireless systems under unreliable silicon”. In: *Dac Acm* (2012), pp. 510–515. ISSN: 0738100X. DOI: [10.1145/2228360.2228451](https://doi.org/10.1145/2228360.2228451).
- [Karapetyan, 2015] S. Karapetyan and U. Schlichtmann. “Integrating aging aware timing analysis into a commercial STA tool”. In: *VLSI Design, Automation and Test (VLSI-DAT), 2015 International Symposium on*. IEEE. 2015, pp. 1–4.
- [Karl, 2005] E. Karl, D. Sylvester, and D. Blaauw. “Timing error correction techniques for voltage-scalable on-chip memories”. In: *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*. IEEE. 2005, pp. 3563–3566.
- [Kemp, 2017] S. Kemp. *Digital in 2017: global overview*. 2017.
- [Khalaf, 2016] K. Khalaf, V. Vidojkovic, K. Vaesen, M. Libois, G. Mangraviti, V. Szortyka, C. Li, B. Verbruggen, M. Ingels, A. Bourdoux, C. Soens, W. V. Thillo, J. R. Long, and P. Wambacq. “Digitally Modulated CMOS Polar Transmitters for Highly-Efficient mm-Wave Wireless Communication”. In: *IEEE Journal of Solid-State Circuits* PP.99 (2016), pp. 1–14. ISSN: 0018-9200. DOI: [10.1109/JSSC.2016.2544784](https://doi.org/10.1109/JSSC.2016.2544784).
- [Kleeberger, 2013] V. B. Kleeberger, C. Gimmler-Dumont, C. Weis, A. Herkersdorf, D. Mueller-Gritschneider, S. R. Nassif, U. Schlichtmann, and N. Wehn. “A cross-layer technology-based study of how memory errors impact system resilience”. In: *IEEE Micro* 33.4 (2013), pp. 46–55.

- [Knudsen, 2006] J. E. Knudsen and L. T. Clark. “An area and power efficient radiation hardened by design flip-flop”. In: *IEEE Transactions on Nuclear Science*. Vol. 53. 6. 2006, pp. 3392–3399. DOI: [10 . 1109 / TNS . 2006 . 886199](https://doi.org/10.1109/TNS.2006.886199).
- [Kraemer, 2007] S. Kraemer, R. Leupers, G. Ascheid, and H. Meyr. “Interactive presentation: Softsimd-exploiting sub-word parallelism using source code transformations”. In: *Proceedings of the conference on Design, automation and test in Europe*. EDA Consortium. 2007, pp. 1349–1354.
- [Kristiansson, 2013] S. Kristiansson and J. Baxter. *mor1kx - an OpenRISC Processor IP Core*. 2013.
- [Kükner, 2013] H. Kükner, P. Weckx, P. Raghavan, B. Kaczer, F. Catthoor, L. der Perre, R. Lauwereins, and G. Groeseneken. “Impact of duty factor, stress stimuli, gate and drive strength on gate delay degradation with an atomistic trap-based BTI model”. In: *Microprocessors and Microsystems* 37.8 (2013), pp. 792–800.
- [Kulkarni, 2006] S. H. Kulkarni, D. Sylvester, and D. Blaauw. “A statistical framework for post-silicon tuning through body bias clustering”. In: *Proceedings of the 2006 IEEE/ACM international conference on Computer-aided design*. ACM. 2006, pp. 39–46.
- [Kulkarni, 2015] J. P. Kulkarni, C. Tokunaga, P. Aseron, T. Nguyen, C. Augustine, J. Tschanz, and V. De. “4.7 A 409GOPS/W adaptive and resilient domino register file in 22nm tri-gate CMOS featuring in-situ timing margin and error detection for tolerance to within-die variation, voltage droop, temperature and aging”. In: *ISSCC, 2015 IEEE*. Feb. 2015, pp. 1–3. DOI: [10.1109/ISSCC.2015.7062936](https://doi.org/10.1109/ISSCC.2015.7062936).
- [Kunitake, 2011] Y. Kunitake, T. Sato, H. Yasuura, and T. Hayashida. “Possibilities to miss predicting timing errors in canary flip-flops”. In: *Circuits and Systems (MWS-CAS), 2011 IEEE 54th International Midwest Symposium on*. IEEE. 2011, pp. 1–4.
- [Lampret, 2014] D. Lampret and J. Baxter. *OpenRISC 1200 IP Core Specification (Preliminary Draft), 2014*. 2014.

- [Larsson, 2014] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta. “Massive MIMO for next generation wireless systems”. In: *Communications Magazine, IEEE* 52.2 (2014), pp. 186–195.
- [Larsson, 2017] E. G. Larsson and L. der Perre. “Massive MIMO for 5G”. In: *IEEE 5G Tech Focus* 1.1 (2017).
- [Lesea, 2005] A. Lesea, S. Drimer, J. J. Fabula, C. Carmichael, and P. Alfke. “The rosetta experiment: atmospheric soft error rate testing in differing technology FPGAs”. In: *IEEE Transactions on Device and Materials Reliability* 5.3 (2005), pp. 317–328.
- [Li, 2015a] C. Li, A. Bourdoux, M. Verhelst, Y. Huang, M. Li, L. Van Der Perre, and S. Pollin. “< 30 mW rectangular-to-polar conversion processor in 802.11ad polar transmitter”. In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. South Brisbane, Australia: IEEE, Apr. 2015, pp. 1022–1026. ISBN: 978-1-4673-6997-8. DOI: [10.1109/ICASSP.2015.7178124](https://doi.org/10.1109/ICASSP.2015.7178124).
- [Li, 2015b] M. Li, Y. Lee, Y. Huang, and L. Van der Perre. “Area and energy efficient 802.11ad LDPC decoding processor”. In: *Electronics Letters* 51.4 (Feb. 2015), pp. 339–341. ISSN: 0013-5194. DOI: [10.1049/e1.2014.4263](https://doi.org/10.1049/e1.2014.4263).
- [Li, 2015c] M. Li, J.-W. Weijers, V. Derudder, I. Vos, M. Rykunov, S. Dupont, P. Debacker, A. Dewilde, Y. Huang, and L. Van der Perre. “An energy efficient 18Gbps LDPC decoding processor for 802.11ad in 28nm CMOS”. In: *2015 IEEE Asian Solid-State Circuits Conference (A-SSCC)*. Xiamen, China: IEEE, Nov. 2015, pp. 1–5. ISBN: 978-1-4673-7191-9. DOI: [10.1109/ASSCC.2015.7387473](https://doi.org/10.1109/ASSCC.2015.7387473).
- [Li, 2015d] S. Li. *Structural and RTL mixed VHDL implementation of an FFT*. 2015.
- [Liu, 2010] Y. Liu, T. Zhang, and K. K. Parhi. “Computation error analysis in digital signal processing systems with overscaled supply voltage”. In: *IEEE Transactions on Very Large Scale Integration(VLSI) Systems* 18.1-4 (2010), pp. 517–526.

- [Liu, 2014] C. Liu, J. Han, and F. Lombardi. “A low-power, high-performance approximate multiplier with configurable partial error recovery”. In: *Proceedings of the conference on Design, Automation & Test in Europe*. European Design and Automation Association. 2014, p. 95.
- [Liu, 2017] T. Liu, C.-C. Chen, and L. Milor. “Comprehensive Reliability-Aware Statistical Timing Analysis Using a Unified Gate-Delay Model for Microprocessors”. In: *IEEE Transactions on Emerging Topics in Computing* (2017).
- [Marchiò, 2014] F. Marchiò, B. Vittorelli, and R. Colombo. “Automotive electronics: Application & technology megatrends”. In: *European Solid State Circuits Conference (ESSCIRC), ESSCIRC 2014-40th*. IEEE. 2014, pp. 23–29.
- [Martin, 2002] S. M. Martin, K. Flautner, T. Mudge, and D. Blaauw. “Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads”. In: *Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design*. ACM. 2002, pp. 721–725.
- [May, 2008] M. May, M. Alles, and N. Wehn. “A Case Study in Reliability-Aware Design: A Resilient LDPC Code Decoder”. In: *DATE* (Mar. 2008), pp. 456–461. DOI: [10.1109/DATE.2008.4484723](https://doi.org/10.1109/DATE.2008.4484723).
- [Mehta, 2010] J. Mehta, R. B. Staszewski, O. Eliezer, S. Rezek, K. Waheed, M. Entezari, G. Feygin, S. Vemulapalli, V. Zoicas, C.-M. Hung, et al. “A 0.8 mm² all-digital SAW-less polar transmitter in 65nm EDGE SoC”. In: *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*. IEEE. 2010, pp. 58–59.
- [Mei, 1974] K. C. Y. Mei. “Bridging and stuck-at faults”. In: *IEEE Transactions on Computers* 100.7 (1974), pp. 720–727.
- [METI, 2008] METI. *Green IT Initiative in Japan*. 2008.

- [Mintarno, 2013] E. Mintarno, V. Chandra, D. Pietromonaco, R. Aitken, and R. W. Dutton. “Workload dependent NBTI and PBTI analysis for a sub-45nm commercial microprocessor”. In: *Reliability Physics Symposium (IRPS), 2013 IEEE International*. IEEE. 2013, 3A–1.
- [Mirkhani, 2015] S. Mirkhani, B. Samynathan, and J. A. Abraham. “In-depth Soft Error Vulnerability Analysis using Synthetic Benchmarks”. In: *VLSI Test Symposium (2015)*, pp. 2–7. DOI: [10.1109/VTS.2015.7116254](https://doi.org/10.1109/VTS.2015.7116254).
- [MiroPanades, 2014] I. Miro-Panades, E. Beigné, Y. Thonnart, L. Alacoque, P. Vivet, S. Lesecq, D. Puschini, A. Molnos, F. Thabet, B. Tain, et al. “A Fine-Grain Variation-Aware Dynamic V_{dd} -Hopping AVFS Architecture on a 32 nm GALS MPSoC”. In: *IEEE Journal of Solid-State Circuits* 49.7 (2014), pp. 1475–1486.
- [Mitra, 2005] S. Mitra, N. Seifert, M. Zhang, Q. Shi, and K. S. Kim. “Robust system design with built-in soft-error resilience”. In: *Computer* 38.2 (2005), pp. 43–52. ISSN: 00189162. DOI: [10.1109/MC.2005.70](https://doi.org/10.1109/MC.2005.70).
- [Mitra, 2010] S. Mitra, K. Brelsford, and P. N. Sanda. “Cross-layer resilience challenges: Metrics and optimization”. In: *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2010*. IEEE. 2010, pp. 1029–1034.
- [Moore, 1998] G. E. Moore et al. “Cramming more components onto integrated circuits”. In: *Proceedings of the IEEE* 86.1 (1998), pp. 82–85.
- [Mukherjee, 2003] S. S. Mukherjee, C. Weaver, J. Emer, S. K. Reinhardt, and T. Austin. “A systematic methodology to compute the architectural vulnerability factors for a high-performance microprocessor”. In: *Proceedings of the Annual International Symposium on Microarchitecture, MICRO*. Vol. 2003-Janua. December. 2003, pp. 29–40. ISBN: 076952043X. DOI: [10.1109/MICRO.2003.1253181](https://doi.org/10.1109/MICRO.2003.1253181).
- [Muller, 2012] J. Muller, B. Stefanelli, A. Frappe, L. Ye, A. Cathelin, A. Niknejad, and A. Kaiser. “A 7-Bit 18th Order 9.6 GS/s FIR Up-Sampling Filter for High Data Rate 60-GHz Wireless Transmitters”. In: *IEEE Journal of*

- Solid-State Circuits* 47.7 (July 2012), pp. 1743–1756. ISSN: 0018-9200. DOI: [10.1109/JSSC.2012.2191677](https://doi.org/10.1109/JSSC.2012.2191677).
- [Murali, 2005] S. Murali, T. Theocharides, N. Vijaykrishnan, M. J. Irwin, L. Benini, and G. De Micheli. “Analysis of error recovery schemes for networks on chips”. In: *IEEE Design & Test of Computers* 22.5 (2005), pp. 434–442.
- [Narayanan, 2010] S. Narayanan, J. Sartori, R. Kumar, and D. L. Jones. “Scalable stochastic processors”. In: *Proceedings of the Conference on Design, Automation and Test in Europe*. European Design and Automation Association. 2010, pp. 335–338.
- [Nicolaidis, 2013] M. Nicolaidis. “ADDA: Adaptive Double-sampling Architecture for Highly Flexible Robust Design”. In: *Design Automation and Test in Europe Conference (DATE), Grenoble*. IEEE Computer Society. 2013, pp.
- [Nithin, 2010] S. K. Nithin, G. Shanmugam, and S. Chandrasekar. “Dynamic voltage (IR) drop analysis and design closure: Issues and challenges”. In: *Quality Electronic Design (ISQED), 2010 11th International Symposium on*. IEEE. 2010, pp. 611–617.
- [Nourani, 2006] M. Nourani and A. Radhakrishnan. “Testing on-die process variation in nanometer VLSI”. In: *IEEE Design & Test of Computers* 23.6 (2006).
- [Novo, 2010] D. Novo, A. Kritikakou, P. Raghavan, L. der Perre, J. Huisken, and F. Catthoor. “Ultra low energy Domain Specific Instruction-set Processor for on-line surveillance”. In: *Application Specific Processors (SASP), 2010 IEEE 8th Symposium on*. June 2010, pp. 30–35. DOI: [10.1109/SASP.2010.5521151](https://doi.org/10.1109/SASP.2010.5521151).
- [Nowka, 2002] K. J. Nowka, G. D. Carpenter, E. W. MacDonald, H. C. Ngo, B. C. Brock, K. I. Ishii, T. Y. Nguyen, and J. L. Burns. “A 32-bit PowerPC system-on-a-chip with support for dynamic voltage scaling and dynamic frequency scaling”. In: *IEEE Journal of Solid-State Circuits* 37.11 (2002), pp. 1441–1447.
- [Ohnari, 2013] Y. Ohnari, A. A. Khan, A. Dutta, M. Miura-Mattausch, and H. J. Mattausch. “Die-to-die and within-die variation extraction for circuit simulation with surface-potential compact model”. In:

- Microelectronic Test Structures (ICMTS), 2013 IEEE International Conference on.* IEEE. 2013, pp. 146–150.
- [Orshansky, 2007] M. Orshansky, S. Nassif, and D. Boning. *Design for manufacturability and statistical design: a constructive approach.* Springer Science & Business Media, 2007.
- [Osseiran, 2014] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, H. Tullberg, M. A. Uusitalo, B. Timus, and M. Fallgren. “Scenarios for 5G mobile and wireless communications: the vision of the METIS project”. In: *IEEE Communications Magazine* 52.5 (May 2014), pp. 26–35. ISSN: 0163-6804. DOI: [10.1109/MCOM.2014.6815890](https://doi.org/10.1109/MCOM.2014.6815890).
- [Ottavi, 2015] M. Ottavi, S. Pontarelli, D. Gizopoulos, A. Paschalis, C. Bolchini, M. K. Michael, L. Anghel, M. Tahoori, P. Reviriego, O. Bringmann, V. Izosimov, H. Manhaeve, C. Strydis, and S. Hamdioui. “Dependable multicore architectures at nanoscale: The view from Europe”. In: *IEEE Design and Test* 32.2 (2015), pp. 17–28. ISSN: 21682356. DOI: [10.1109/MDAT.2014.2359572](https://doi.org/10.1109/MDAT.2014.2359572).
- [Pavlov, 2008] A. Pavlov and M. Sachdev. *CMOS SRAM circuit design and parametric test in nano-scaled technologies: process-aware SRAM design and test.* Vol. 40. Springer Science & Business Media, 2008.
- [Pillai, 2001] P. Pillai and K. G. Shin. “Real-time dynamic voltage scaling for low-power embedded operating systems”. In: *ACM SIGOPS Operating Systems Review.* Vol. 35. 5. ACM. 2001, pp. 89–102.
- [Polian, 2011] I. Polian, J. P. Hayes, S. M. Reddy, and B. Becker. “Modeling and mitigating transient errors in logic circuits”. In: *IEEE Transactions on Dependable and Secure Computing* 8.4 (2011), pp. 537–547. ISSN: 15455971. DOI: [10.1109/TDSC.2010.26](https://doi.org/10.1109/TDSC.2010.26).
- [Prabhu, 2017] H. Prabhu, J. N. Rodrigues, L. Liu, and O. Edfors. “A 60pJ/b 300Mb/s 128x8 Massive MIMO precoder-detector in 28nm FD-SOI”. In: *Solid-State Circuits Conference (ISSCC), 2017 IEEE International.* IEEE. 2017, pp. 60–61.

- [Rabaey, 2008] J. M. Rabaey and S. Malik. “Challenges and solutions for late-and post-silicon design”. In: *IEEE Design & Test of Computers* 4 (2008), pp. 296–302.
- [Rachala, 2016] R. Rachala, M. Rodriguez, S. Kosonocky, and M. Trajkovic. “Modeling and implementation of a fully-digital integrated per-core voltage regulation system in a 28nm high performance 64-bit processor”. In: *Proceedings of the 2016 International Symposium on Low Power Electronics and Design*. ACM. 2016, pp. 88–93.
- [Ramanarayanan, 2003] R. Ramanarayanan, V. Degalahal, N. Vijaykrishnan, M. J. Irwin, and D. Duarte. “Analysis of soft error rate in flip-flops and scannable latches”. In: *SOC Conference, 2003. Proceedings. IEEE International [Systems-on-Chip]*. IEEE. 2003, pp. 231–234.
- [Rawat, 2014] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin. “Wireless sensor networks: a survey on recent developments and potential synergies”. In: *The Journal of supercomputing* 68.1 (2014), pp. 1–48.
- [Reddy, 2005] V. Reddy, A. T. Krishnan, A. Marshall, J. Rodriguez, S. Natarajan, T. Rost, and S. Krishnan. “Impact of negative bias temperature instability on digital circuit reliability”. In: *Microelectronics Reliability* 45.1 (2005), pp. 31–38. ISSN: 00262714. DOI: [10.1016/j.microrel.2004.05.023](https://doi.org/10.1016/j.microrel.2004.05.023).
- [Richter, 2008] M. Richter, K. Oberlaender, and M. Goessel. “New linear SEC-DED codes with reduced triple bit error miscorrection probability”. In: *On-Line Testing Symposium, 2008. IOLTS'08. 14th IEEE International*. IEEE. 2008, pp. 37–42.
- [Saha, 2010] S. K. Saha. “Modeling process variability in scaled CMOS technology”. In: *IEEE Design & Test of Computers* 27.2 (2010).
- [Samudrala, 2004] P. K. Samudrala, J. Ramos, and S. Katkoori. “Selective triple modular redundancy (STMR) based single-event upset (SEU) tolerant synthesis for FPGAs”. In: *IEEE Transactions on Nuclear Science* 51.5 IV (2004), pp. 2957–2969. ISSN: 00189499. DOI: [10.1109/TNS.2004.834955](https://doi.org/10.1109/TNS.2004.834955).

- [Shafik, 2006] R. A. Shafik, M. S. Rahman, and A. H. M. R. Islam. “On the extended relationships among EVM, BER and SNR as performance metrics”. In: *Electrical and Computer Engineering, 2006. ICECE’06. International Conference on*. IEEE. 2006, pp. 408–411.
- [Shim, 2004] B. Shim, S. R. Sridhara, and N. R. Shanbhag. “Reliable low-power digital signal processing via reduced precision redundancy”. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 12.5 (May 2004), pp. 497–510. ISSN: 1063-8210. DOI: [10.1109/TVLSI.2004.826201](https://doi.org/10.1109/TVLSI.2004.826201).
- [Shin, 2013] I. Shin, J.-j. Kim, Y.-s. Lin, and Y. Shin. “Architecture with 1-Cycle Timing Error for Low Voltage Operations Correction for Low Voltage Operations”. In: *Low Power Electronics and Design (ISLPED), 2013 IEEE International Symposium on*. Vol. 3. 2013, pp. 199–204. ISBN: 9781479912353.
- [Siewiorek, 2014] D. Siewiorek and R. Swarz. *Reliable Computer Systems: Design and Evaluation*. Digital Press, 2014.
- [Skadron, 2004] K. Skadron. “Hybrid architectural dynamic thermal management”. In: *Proceedings of the conference on Design, automation and test in Europe-Volume 1*. IEEE Computer Society. 2004, p. 10010.
- [Slayman, 2005] C. W. Slayman. “Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations”. In: *IEEE Transactions on Device and Materials Reliability* 5.3 (2005), pp. 397–404.
- [SPARC, 1994] I. I. SPARC and D. L. Weaver. *The SPARC architecture manual*. Prentice-Hall, 1994.
- [Stamoulis, 2016] D. Stamoulis, S. Corbetta, D. Rodopoulos, P. Weckx, P. Debacker, B. H. Meyer, B. Kaczer, P. Raghavan, D. Soudris, F. Catthoor, et al. “Capturing true workload dependency of BTI-induced degradation in CPU components”. In: *Proceedings of the 26th edition on Great Lakes Symposium on VLSI*. ACM. 2016, pp. 373–376.

- [Stine, 2007] J. E. Stine, I. Castellanos, M. Wood, J. Henson, F. Love, W. R. Davis, P. D. Franzon, M. Bucher, S. Basavarajaiah, J. Oh, et al. “FreePDK: An open-source variation-aware design kit”. In: *Microelectronic Systems Education, 2007. MSE’07. IEEE International Conference on*. IEEE. 2007, pp. 173–174.
- [Strojwas, 1989] A. J. Strojwas. “Design for manufacturability and yield”. In: *Design Automation, 1989. 26th Conference on*. IEEE. 1989, pp. 454–459.
- [Strollo, 2008] A. Strollo, D. De Caro, and N. Petra. “A 430 MHz, 280 mW Processor for the Conversion of Cartesian to Polar Coordinates in 0.25 CMOS”. In: *Solid-State Circuits, IEEE Journal of* 43.11 (2008), pp. 2503–2513.
- [Sutter, 2005] H. Sutter. “The free lunch is over: A fundamental turn toward concurrency in software”. In: *Dr. Dobb’s journal* 30.3 (2005), pp. 202–210.
- [Tam, 2000] S. Tam, S. Rusu, U. N. Desai, R. Kim, J. Zhang, and I. Young. “Clock generation and distribution for the first IA-64 microprocessor”. In: *IEEE Journal of Solid-State Circuits* 35.11 (2000), pp. 1545–1552.
- [Tsai, 2004] J.-L. Tsai, D. Baik, C. C.-P. Chen, and K. K. Saluja. “A yield improvement methodology using pre-and post-silicon statistical clock scheduling”. In: *Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design*. IEEE Computer Society. 2004, pp. 611–618.
- [Tschanz, 2002] J. W. Tschanz, J. T. Kao, S. G. Narendra, R. Nair, D. A. Antoniadis, A. P. Chandrakasan, and V. De. “Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage”. In: *IEEE Journal of Solid-State Circuits* 37.11 (2002), pp. 1396–1402.
- [Tschanz, 2007] J. Tschanz, N. S. Kim, S. Dighe, J. Howard, G. Ruhl, S. Vangal, S. Narendra, Y. Hoskote, H. Wilson, C. Lam, et al. “Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging”. In: *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest*

- of Technical Papers. IEEE International. IEEE. 2007, pp. 292–604.*
- [Unsal, 2006] O. S. Unsal, J. W. Tschanz, K. Bowman, V. De, X. Vera, A. Gonzalez, and O. Ergin. “Impact of parameter variations on circuits and microarchitecture”. In: *Ieee Micro* 26.6 (2006), pp. 30–39.
- [Van Zeijl, 2007] P. Van Zeijl and M. Collados. “A digital envelope modulator for a WLAN OFDM polar transmitter in 90 nm CMOS”. In: *Solid-State Circuits, IEEE Journal of* 42.10 (2007), pp. 2204–2211.
- [Vattikonda, 2006] R. Vattikonda, W. Wang, and Y. Cao. “Modeling and minimization of PMOS NBTI effect for robust nanometer design”. In: *Proceedings of the 43rd annual Design Automation Conference. ACM. 2006, pp. 1047–1052.*
- [Veendrick, 2008] H. Veendrick. *Nanometer CMOS ICs: from basics to ASICs*. Springer Publishing Company, Incorporated, 2008.
- [Volder, 1959] J. E. Volder. “The CORDIC trigonometric computing technique”. In: *Electronic Computers, IRE Transactions on* 3 (1959), pp. 330–334.
- [Wang, 2003] L. Wang and N. R. Shanbhag. “Low-power filtering via adaptive error-cancellation”. In: *IEEE Transactions on Signal Processing* 51.2 (2003), pp. 575–583.
- [Wang, 2011] X. Wang, A. R. Brown, B. Cheng, and A. Asenov. “Statistical variability and reliability in nanoscale FinFETs”. In: *Electron Devices Meeting (IEDM), 2011 IEEE International. IEEE. 2011, pp. 4–5.*
- [Wang, 2013] Z. Wang, K. Singh, C. Chen, and A. Chattopadhyay. “Accurate and efficient reliability estimation techniques during adl-driven embedded processor design”. In: *DATE. 2013, pp. 547–552. ISBN: 15301591 (ISSN); 9783981537000 (ISBN). DOI: [10.7873/DATE.2013.122](https://doi.org/10.7873/DATE.2013.122).*
- [Waterman, 2011] A. Waterman, Y. Lee, D. A. Patterson, and K. Asanovic. “The risc-v instruction set manual, volume i: Base user-level isa”. In: *EECS Department, UC Berkeley, Tech. Rep. UCB/EECS-2011-62 (2011).*

- [Whatmough, 2014] P. N. Whatmough, S. Das, and D. M. Bull. “A Low-Power 1-GHz Razor FIR Accelerator With Time-Borrow Tracking Pipeline and Approximate Error Correction in 65-nm CMOS”. In: *IEEE Journal of Solid-State Circuits* 49.1 (Jan. 2014), pp. 84–94. ISSN: 0018-9200. DOI: [10.1109/JSSC.2013.2284364](https://doi.org/10.1109/JSSC.2013.2284364).
- [Wilson, 2013] L. Wilson. “International technology roadmap for semiconductors (ITRS)”. In: *Semiconductor Industry Association* (2013).
- [Yamabe, 1985] K. Yamabe and K. Taniguchi. “Time-dependent-dielectric breakdown of thin thermally grown SiO₂ films”. In: *IEEE Transactions on Electron Devices* 32.2 (1985), pp. 423–428.

Curriculum Vitea

Yanxiang Huang received the B.Sc. degree in 2011 from Zhejiang University, China, and the M.Sc. degree in 2013 from Eindhoven Technology University (TU/e), the Netherlands, both in electrical engineering. Since November 2013, he has been working toward the Ph.D. degree in the Department of Electrical Engineering (ESAT) at University of Leuven (KU Leuven), with his research activities performed in collaboration with imec, Leuven, Belgium.

During 2012 to 2013, he served as a research intern in Research Division, NXP semiconductors, Eindhoven, the Netherlands. Since 2013, he has been working as a Ph.D. researcher in the Perceptive System Group, imec, Leuven, Belgium.

His research interest includes the design of digital circuits and systems, digital hardware units for wireless communication, IoT devices, multimedia accelerators, low-power computing, reliability, and algorithm-architecture co-exploration for digital computing.

List of publications

Journal (Authored and Co-Authored)

- Y. Huang, M. Li, C. Li, P. Debacker, and L. Van der Perre. “Computation-skip Error Mitigation Scheme for Power Supply Voltage Scaling in Recursive Applications”. In: *Journal of Signal Processing Systems (JSPS)* 84.3 (Sept. 2016), pp. 413–424. ISSN: 1939-8018. DOI: [10.1007/s11265-015-1096-z](https://doi.org/10.1007/s11265-015-1096-z).
- Y. Huang, C. Li, L. Meng, L. Van der Perre, and W. Dehaene. “Fine-grained hardware switching scheme for power reduction in multiplication”. In: *Electronics Letters* 52.16 (Aug. 2016), pp. 1374–1375. ISSN: 0013-5194. DOI: [10.1049/el.2015.3828](https://doi.org/10.1049/el.2015.3828).
- C. Li, Y. Huang, K. Khalaf, A. Bourdoux, M. Verhelst, L. Van Der Perre, and S. Pollin. “Energy-Efficient Digital Front-End Processor for 60 GHz Polar Transmitter”. In: *Journal of Signal Processing Systems (JSPS)* 1939.8115 (Dec. 2016), pp. 1–13. ISSN: 1939-8018. DOI: [10.1007/s11265-016-1213-7](https://doi.org/10.1007/s11265-016-1213-7).
- M. Li, Y. Lee, Y. Huang, and L. Van der Perre. “Area and energy efficient 802.11ad LDPC decoding processor”. In: *Electronics Letters* 51.4 (Feb. 2015), pp. 339–341. ISSN: 0013-5194. DOI: [10.1049/el.2014.4263](https://doi.org/10.1049/el.2014.4263).
- Y. Huang, A. Kapoor, R. Rutten, and J. Pineda de Gyvez. “A 13bits 4.096GHz 45nm CMOS digital decimation filter chain with Carry-Save format numbers”. In: *Microprocessors and Microsystems* 39.8 (Nov. 2015), pp. 869–878. ISSN: 01419331. DOI: [10.1016/j.micpro.2014.11.003](https://doi.org/10.1016/j.micpro.2014.11.003).

Patents (Authored and Co-Authored)

- Y. Huang, C. Li, and M. Li. *Error Resilient Digital Signal Processing Device*. 2015.

C. Li, Y. Huang, and A. Bourdoux. *A digital frontend system for a radio transmitter and a method thereof*. 2016.

Conferences and Workshops (Authored and Co-Authored)

- Y. Huang, C. Desset, A. Bourdoux, W. Dehaene, and L. Van der Perre. “Massive MIMO processing at the semiconductor edge: Exploiting the system and circuit margins for power savings”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New Orleans, U.S.: IEEE, Mar. 2017, pp. 3474–3478. ISBN: 978-1-5090-4117-6. DOI: [10.1109/ICASSP.2017.7952802](https://doi.org/10.1109/ICASSP.2017.7952802).
- S. Gunnarsson, M. Bortas, Y. Huang, C.-M. Chen, L. Van der Perre, and O. Edfors. “Lousy processing increases energy efficiency in massive MIMO systems”. In: *2017 European Conference on Networks and Communications (EuCNC)*. Oulu, Finland, June 2017, pp. 1–5. DOI: [10.1109/EuCNC.2017.7980739](https://doi.org/10.1109/EuCNC.2017.7980739).
- Y. Huang, C. Li, M. Li, F. Catthoor, L. V. der Perre, and W. Dehaene. “An algorithm to identify cornerstones of digital circuits”. In: *ACM/IEEE Design Automation Conference (DAC), Work-In-Progress*. Austin, U.S., 2016.
- Y. Huang, C. Li, K. Khalaf, A. Bourdoux, J. Verschueren, Q. Shi, P. Wambacq, S. Polling, W. Dehaene, and L. Van der Perre. “A 28 nm CMOS 7.04 Gbps polar digital front-end processor for 60 GHz transmitter”. In: *2016 IEEE Asian Solid-State Circuits Conference (A-SSCC)*. Toyama, Japan: IEEE, Nov. 2016, pp. 333–336. ISBN: 978-1-5090-3699-8. DOI: [10.1109/ASSCC.2016.7844203](https://doi.org/10.1109/ASSCC.2016.7844203).
- Y. Huang, M. Li, C. Li, and L. Van der Perre. “Partial Computation-skip Scheme for Power Supply Voltage Scaling”. In: *1st Workshop On Approximate Computing, HiPEAC (WAPCO)*. Amsterdam, Netherlands, 2015.
- C. Li, A. Bourdoux, M. Verhelst, Y. Huang, M. Li, L. Van Der Perre, and S. Pollin. “< 30 mW rectangular-to-polar conversion processor in 802.11 ad polar transmitter”. In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. South Brisbane, Australia: IEEE, Apr. 2015, pp. 1022–1026. ISBN: 978-1-4673-6997-8. DOI: [10.1109/ICASSP.2015.7178124](https://doi.org/10.1109/ICASSP.2015.7178124).

- M. Li, J.-W. Weijers, V. Derudder, I. Vos, M. Rykunov, S. Dupont, P. Debacker, A. Dewilde, Y. Huang, and L. Van der Perre. “An energy efficient 18Gbps LDPC decoding processor for 802.11ad in 28nm CMOS”. In: *2015 IEEE Asian Solid-State Circuits Conference (A-SSCC)*. Xiamen, China: IEEE, Nov. 2015, pp. 1–5. ISBN: 978-1-4673-7191-9. DOI: [10.1109/ASSCC.2015.7387473](https://doi.org/10.1109/ASSCC.2015.7387473).
- Y. Huang, M. Li, C. Li, P. Debacker, and L. Van der Perre. “Computation-skip error resilient scheme for recursive CORDIC”. In: *2014 IEEE Workshop on Signal Processing Systems (SiPS)*. Belfast, Northern Ireland: IEEE, Oct. 2014, pp. 1–6. ISBN: 978-1-4799-6588-5. DOI: [10.1109/SiPS.2014.6986061](https://doi.org/10.1109/SiPS.2014.6986061).
- A. Callejon, M. Li, Y. Huang, S. Pollin, and L. Van der Perre. “Gbps Throughput Architecture for Turbo Decoder”. In: *Information Theory in the Benelux and The 4th Joint WIC/IEEE Symposium on Information Theory and Signal Processing in the Benelux*. Eindhoven, Netherlands, 2014, p. 178.
- Y. Huang, A. Kapoor, R. Rutten, and J. Pineda de Gyvez. “A 13 bits 4.096 GHz 45 nm CMOS digital decimation filter chain using Carry-Save format numbers”. In: *31th Norchip Conference*. Vilnius, Lithuania: IEEE, Nov. 2013, pp. 1–4. ISBN: 978-1-4799-1647-4. DOI: [10.1109/NORCHIP.2013.6702042](https://doi.org/10.1109/NORCHIP.2013.6702042).

Reports

- Y. Huang. “Carry-Save Arithmetic for High-Performance DSP”. Master of Science Thesis. Eindhoven University of Technology, 2013, pp. 1–12.

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF ELECTRICAL ENGINEERING
ESAT
Kasteelpark Arenberg 10
B-3001 Leuven

