

Design Automation of Low Power Circuits in Nano-Scale CMOS and Beyond-CMOS Technologies

by

Elnaz Ansari Ogholbeik

**A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering)
in The University of Michigan
2016**

Doctoral Committee:

**Associate Professor David D. Wentzloff, Chair
Professor David Blaauw
Douglas Carmean, Microsoft Co
Professor Vineet Kamat
Associate Professor Zhengya Zhang**

© Elnaz Ansari

All rights reserved

2016

Dedicated to
my parents for their endless love
my husband for his great support and friendship
my sisters for the joy and happiness they brought to my life

ACKNOWLEDGEMENTS

First and foremost, I would like to express my special appreciation and thanks to my advisor Professor David Wentzloff, who has been a tremendous mentor and coach for me. I would like to thank him for encouraging me throughout my PhD. His advice on both research and my career has been priceless. I will always remember the joyful times I had during the team retreats and the adventurous vacations. I would also like to thank my committee members, Professor Vineet Kamat, Professor David Blaauw, Professor Zhengya Zhang, and Douglas Carmean, for serving as my committee members and providing valuable suggestions to further refine my dissertation. A special thanks to Douglas Carmean for his coaching and great advice on my last PhD project.

I would like to express my gratitude to EECS staff members who have helped me in many ways. Thank you Karen Liska, Beth Stalnaker, Steven Pejuan, Kyle Banas, Sarah Towler, Fran Doman, and Melanie Caughey.

I am thankful to the past and present members of our research group (WICS), Youngmin Park, Sangwook Han, Jonathan Brown, Dea Young Lee, Seunghyun Oh, Kuo-Ken Huang, Osama Khan, Muhammad Faisal, Nathan Robert, Ryan Rogel, Hyeongseok Kim, David Moore, Mike Kines, Avish Kosari, Yao Shi, Byron Tanous, Xing chen, Abdullah Alghaihab, Jaeho Im, Jiannan Huang, and Milad Moosavifard, with whom I have had a lot of great

discussions and fun times. In particular, I would like to thank Avish, Kuo-Ken, Muhammad, Osama, David, Nathan and Hyeongseok for their great friendship; I always enjoyed our conversations.

I would like to thank my brilliant classmates, and colleagues at the office EECS 2435, especially Chunyang Zhai, Rohit Deshpande, Russell Willmot, Yoonmyung Lee, Zhiyoong Foo, Gyouho Kim, Inhee Lee, Yejoong Kim, Nick Collins, Mohammad Ghahramani, and Jeffrey Fredenburg, and Jorge Pernillo with whom I have taken courses, done projects, played games, and had a lot of fun memories. I would also like to thank the member of Quantum Architecture (QuArc) Engineering team at Microsoft Research, for their invaluable inputs and feedbacks on my last PhD project.

My valuable friends in Ann Arbor have helped me remember that life is all about love and friendship. I wish to acknowledge them all for being there for me, especially Parisa Ghaderi, Mehrzad Samadi, Azadeh Ansari, Avish Kosari, Armin Jam, Hamidreza Tavafoghi, Vahed Qazvinian, Mohammadreza Imani, Parinaz Naghizadeh, Katayoon Sabet, Payam Mirshams Shahshahani, Mojtaba Mehrara, Nasibeh Nourbakhshnia, Mohammad Olfatnia, Mona Attarian, Amirhossein Hormati, Niloufar Ghafouri, Alireza Tabatabaeenejad, Maryam Arbabzadeh, Alireza Sarebanha, Sara Hadavi, Ali Besharatian, Mahta Mousavi, Mahmoud Barangi Ali Askarinejad, Mozhdeh Aminmadani, Yaser Zerehsaz, Shima Abadi, Hadi Katebi, Azadeh Haratian, Ehsan Nasr, Roshan Najafi, Nina Zabihi, Hedieh Alavi, Hossein Tamadoni, Frahad Shirani, Mohsen Heidari, Behnam Kamrani, Hassan Ghaed, Saeedeh Salimian, Parisa Faraji, Mohammadreza Kakoei, and Siamak Davarani.

I am extremely grateful to my parents, Robab Parvizi and Dariush Ansari, for their endless unconditional love and caring, as well as the sacrifices they made for my education and career. Thank you to my lovely little sisters, Sahar Ansari and Samin Ansari, who brought joy and excitement to my life from the moment they were born. I am grateful for their invaluable support and humor. I would like to thank my brother-in-law Calin Voichita, who brought great happiness to our family.

Above all, I want to express my most sincere appreciation to my loving, supportive, encouraging, and patient husband, Armin Alaghi, who has supported me throughout this process and has constantly encouraged me when the tasks seemed arduous and insurmountable. Thank you for the little things you have done like brining food and staying up with me during the tapeout deadlines. You were patient when I was frustrated, you celebrated with me when even the smallest things went right, and you were there whenever I needed you to just listen. Thank you for being my best friend.

TABLE OF CONTENTS

DEDICATION	ii
ACKNOWLEDGEMENTS	iii
LIST OF FIGURES.....	viii
LIST OF TABLES	xii
ABSTRACT.....	xiii
Chapter 1 Introduction	1
1.1. Moore’s Law.....	1
1.2. Denard’s Scaling Factor	3
1.3. More than Moore	5
1.4. Bell’s law.....	6
1.5. Digitally assisted analog designs	7
1.6. Internet of Things.....	9
1.7. Beyond CMOS.....	11
1.8. Contributions.....	12
Chapter 2 Very Large Scale Analog (VLSA) Design Methodology	15
2.1. EDA for Analog Designs	15
2.2. Previous Works on Analog Design Automation	17
2.3. Issues with Current Analog Design Automation Techniques.....	19
2.4. Proposed Very Large Scale Analog (VLSA) Design Flow	20
Chapter 3 Digital to Analog Converters Overview.....	25
3.1. Static Behavior.....	27
3.2. Dynamic Behavior	28
3.3. Popular DAC Topologies.....	30
Chapter 4 VLSA DAC.....	33
4.1. High-Level Architecture.....	33

4.2. Current Cells	35
4.3. Design Trade-Offs, Number of Cells and Look-Up Tables	37
4.4. Calibration	40
4.5. Measurement Results	46
4.6. Conclusions	50
Chapter 5 Baseband DSP for Ultra Wideband (UWB) Transceiver (TRX)	52
5.1. UWB Radio Architecture	55
5.2. Baseband Controller	56
5.3. Measurement Results	60
5.4. Conclusions	65
Chapter 6 Beyond CMOS	66
6.1. Motivation	66
6.2. Introduction to superconducting circuits	68
6.3. Top-down design steps: RTL Verilog to Josephson junctions	78
6.4. RQL design challenges and solutions	81
6.5. Design prototype	88
6.6. Conclusions	91
Chapter 7 Concluding Remarks	93
7.1. Conclusions	94
7.2. Future Directions	95
APPENDIX	97
REFERENCES	101

LIST OF FIGURES

Fig. 1-1 Cost vs. number of components per integrated circuit [1].....	1
Fig. 1-2 Cost and number of transistors in the past 50 years [3]	2
Fig. 1-3 Moore’s law and integrated circuit scaling [2]	3
Fig. 1-4 CMOS transistor scaling roadmap [8]	5
Fig. 1-5 Roadmap for semiconductors: miniaturization of the digital functions (“More Moore”) and functional diversification (More than Moore) [8]	6
Fig. 1-6 Bell’s Law [77].....	7
Fig. 1-7 Two different approaches for interface circuits: (a) precise implementation, and (b) digitally assisted implementation incorporating minimalistic interface components and additional pre- and post-processing units [14].....	8
Fig. 1-8 Internet of things (IoTs) [20].....	9
Fig. 1-9 Cost per transistor for deep nanometer CMOS technology nodes [74].....	10
Fig. 1-10 Emerging Technologies [88].....	11
Fig. 2-1 DRC and operation counts vs. advanced sub-micron technology nodes	16
Fig. 2-2 Analog vs. Digital	17
Fig. 2-3 Tightly coupled APRed digital and analog blocks	22
Fig. 2-4 Very large-scale analog (VLSA) synthesis design flow	23
Fig. 3-1 The Ideal Transfer Function of a DAC [54]	26
Fig. 3-2 Basic model of a DAC, with inputs for data and clock, and an analog output [56] ..	26
Fig. 3-3. Differential nonlinearity (DNL) in DACs [54]	27
Fig. 3-4. Integral nonlinearity (INL) in DACs [54]	28
Fig. 3-5 Example output spectrum of a DAC [56]	29
Fig. 3-6 R-2R ladder DAC [56].....	30

Fig. 3-7 A current-steering DAC architecture [56].....	30
Fig. 3-8 Oversampling DAC with semi-digital reconstruction filtering [56].....	31
Fig. 3-9 Charge-redistribution DAC [56]	32
Fig. 4-1 High level architecture of the synthesized DAC.....	34
Fig. 4-2 DAC current cells and look up tables (LUTs).	35
Fig. 4-3 Design trade-offs – Speed, and memory size vs. total number of current cells (calibration flexibility factor).....	37
Fig. 4-4 Tri-state DAC current cell for calibration purposes.....	42
Fig. 4-5 Calibrations setup	43
Fig. 4-6 DNL and INL measurement results before and after calibration – a) Before calibration, b) After calibration, first step: applying gain adjustment, cell re-ordering, and spare cells utilization. Second step: adding code swapping technique.....	44
Fig. 4-7 DAC output spectrum for low input frequency before and after calibration.....	45
Fig. 4-8 SFDR measured results vs. input frequency and comparison with some recent works; *: [59], **: [60], ***: [61].....	46
Fig. 4-9 Chip die photo – Fabricated in 65nm CMOS.....	47
Fig. 4-10 DNL plots for 3 different chips, before and after calibration (before calibration: blue plots, after calibration: red plots).....	48
Fig. 4-11 Synthesized-DAC analog and digital area-scaling rate from 65nm CMOS to 28nm SOI-CMOS	49
Fig. 5-1 Block diagram of a typical WSN node.....	53
Fig. 5-2 System block diagram of the entire crystal-less UWB radio	55
Fig. 5-3 High level state diagram for (a) transmission, (b) reception.....	57
Fig. 5-4 Bit-level active window of the RX when its clock is 1% faster than the TX clock; the transmitted pulses are not being tracked by the RX.	58
Fig. 5-5 Bit-level active window of the RX when its clock is 1% faster than the TX clock. The transmitted PPM pulses are being tracked by the receiver to maintain synchronization and a constant duty-cycling ratio of 6%	58
Fig. 5-6 Die photo of the radio.....	59
Fig. 5-7 Photo of the cubic-mm stacked WSN system.....	60

Fig. 5-8 Top level block diagram of the proposed wireless sensor node (left), wire-bond diagram, and size comparison of the autonomous node.....	61
Fig. 5-9 Power consumption profile of the radio layer in a basic transmission cycle	62
Fig. 5-10 Test setup, transmission characteristics, and received power at 1.5m. Node lifetime is measured while the sensor node transmits 1kb every 3 seconds at data rate of 30kbps.....	63
Fig. 6-1 Cost per transistor for deep nanometer CMOS technology nodes [74] (copy of Fig. 1-9).....	67
Fig. 6-2 Emerging technologies [88] (copy of Fig. 1-10)	68
Fig. 6-3 Design of an active Josephson transmission line (JTL) [86].....	72
Fig. 6-4 A shift register design, with JTLs that are driven by four-phase clock signals [86].	72
Fig. 6-5 Representation of the 1's and 0's in reciprocal quantum logic (RQL) [86]	73
Fig. 6-6 Timing diagram for a positive input pulse in a JTL	74
Fig. 6-7 Timing diagram for a negative input pulse in a JTL	74
Fig. 6-8 RQL logic gate examples [86].....	75
Fig. 6-9 Timing diagram for a positive followed by a negative input A pulse in the absence of input B	76
Fig. 6-10 Timing diagram for a positive input A pulse in after a positive input pulse at input B.....	76
Fig. 6-11 An example of a synthesized and timed design in RQL technology.....	77
Fig. 6-12 Automated RQL design flow	79
Fig. 6-13 (a) a JTL with missing wires in a congested routing are vs. (b) a successfully placed and wired JTL inside the DPC tool.....	81
Fig. 6-14 The size of standard cells are virtually increased for optimized placement.....	82
Fig. 6-15 Placed RQL standard gates (a)before optimization and (b) after optimization.....	83
Fig. 6-16 Placement and routing of an example design (a) before and (b) after optimization.	84
Fig. 6-17 Number of missing/unrouted wires for different designs with different gate utilizations before and after gate placement optimization	85
Fig. 6-18 Number of failures in meandering wires for different designs with different gate utilizations before and after gate placement optimization	86

Fig. 6-19 Total number of wires (missing wires +missing JTLs + failure in meandering) require rework for different designs with different gate utilizations before and after gate placement optimization.	87
Fig. 6-20 Gate placement and heat map of the wiring and JTL congestions before and after the optimization step.	89
Fig. 8-1 High level schematic of the GPS receiver	98
Fig. 8-2 Comparator block diagram	99
Fig. 8-3 GPS receiver die photo.....	100

LIST OF TABLES

Table 1-1 Danard and interconnect scale factors	4
Table 2-1 Qualitative comparison between the analog, VLSA, and VLSI design flows.....	21
Table 4-1 DAC specifications with segmented decoding (binary+thermometer) of the input code, m represents the number of binary bits	38
Table 4-2 DAC specifications with dividing the DAC cells into different banks, n represents the number of banks.....	39
Table 4-3 Comparison of the proposed DAC with previous designs.....	48
Table 5-1 Summary of the Radio Performance.....	64
Table 6-1 Summary of the implemented blocks in the RQL technology.....	90
Table 6-2 Power comparision for blocks implemented in RQL and 65nm CMOS technologies	91

ABSTRACT

Today's integrated system on chips (SoCs) usually consist of billions of transistors accounting for both digital and analog blocks. Integrating such massive blocks on a single chip involves several challenges, especially when transferring analog blocks from an older technology to newer ones. Furthermore, the exponential growth for IoT devices necessitates small and low power circuits. Hence, new devices and architectures must be investigated to meet the power and area constraints for wireless sensor networks (WSNs). In such cases, design automation becomes an essential tool to reduce the time to market of the circuits.

This dissertation focuses on automating the design process of analog designs in advanced CMOS technology nodes, as well as reciprocal quantum logic (RQL) superconducting circuits. For CMOS analog circuits, our design automation technique employs digital automatic placement and routing tools to synthesize and lay out analog blocks along with digital blocks in a cell-based design approach. This technique was demonstrated in the design of a digital-to-analog converter. In the domain of RQL circuits, the automated design of several functional units of a commercial Processor is presented. These automation techniques enable the design of VLSI-scale circuits in this technology.

In addition to the investigation of new technologies, several new baseband signal processor architectures are presented in this dissertation. These architectures are suitable for low-power mm³-scale WSNs and enable high frequency transceivers to operate within the power constraints of standalone IoT nodes.

Chapter 1

Introduction

1.1. Moore's Law

Half a century ago, Gordon Moore predicted a big trend in the circuit industry that the number of components per design will grow exponentially over time. He forecasted the high-level integration and growth of future electronic devices (such as home computers, mobile phones, automatic control systems for homes and cars, and etc.), as well as the cost

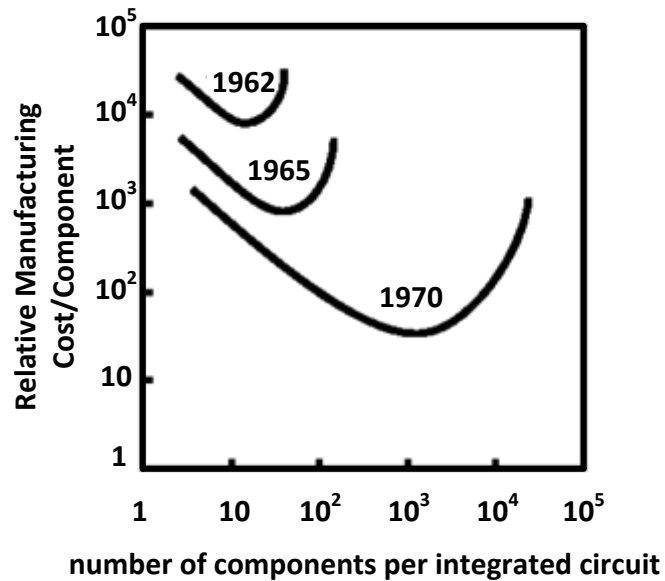


Fig. 1-1 Cost vs. number of components per integrated circuit [1].

per component reduction (see Fig. 1-1) [1]. Ten years after Moore's 4-page paper in trade magazine *electronics*, the exponential growth started; this phenomenon is dubbed as Moore's law, and it shows every 18-24 months the number of transistors per chip doubles. Fig. 1-3 shows the Moore's law and the scaling of integrated circuits [2].

Assuming the cost per wafer stays constant, as a result of Moore's law, the cost per transistor drops. Fig. 1-2 shows the growth in the number of transistors manufactured over the years, and the cost of per transistor [3]. The impact of Moore's law in our modern life cannot be neglected; it influenced many aspects of life such as transportation, phone calls, home appliances, security systems, internets, servers, and etc. [1]. Moore's law has remained somewhat true for the past 50 years, and at every generation, it has allowed higher levels of integration on the chips. Today's SoCs usually consist of billions of transistors accounting for both digital and analog blocks. However, the analog and digital circuitry integration on the same chip has several challenges [4].

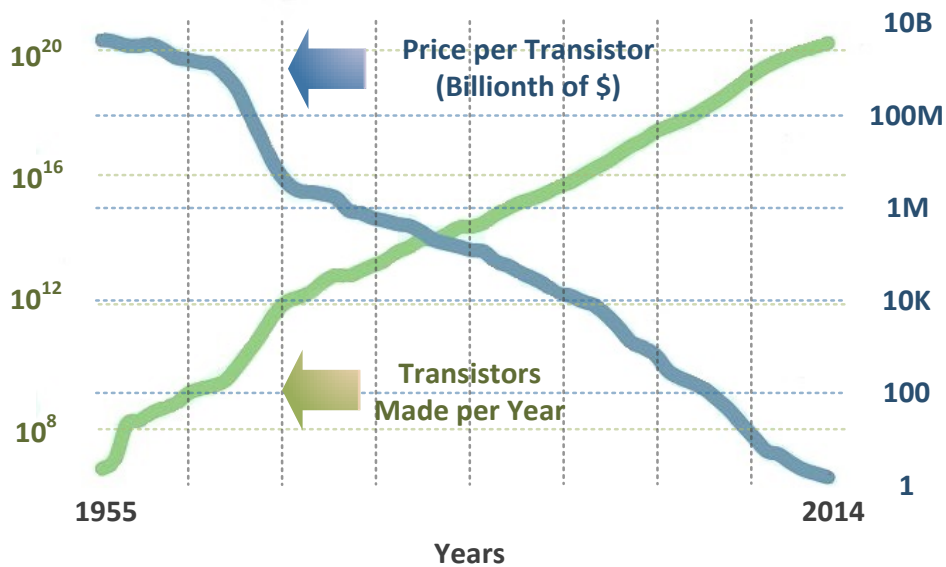


Fig. 1-2 Cost and number of transistors in the past 50 years [3]

1.2. Denard's Scaling Factor

Table 1-1 shows Denard's scaling for digital circuits in the CMOS submicron technologies down to 90nm. Despite the fact that interconnects do not have the same scaling factor in these process nodes, scaling to the smaller nodes has become feasible through design finesse [5].

The performance and supply voltage scaling in digital designs do not follow the Denard's scaling precisely (they scale at a lower rate). However, power, size, and design cost of digital circuits still benefit from scaling. For instance, digital blocks such as memories, processors, etc. benefit most from scaling, which allows migration to newer technology nodes.

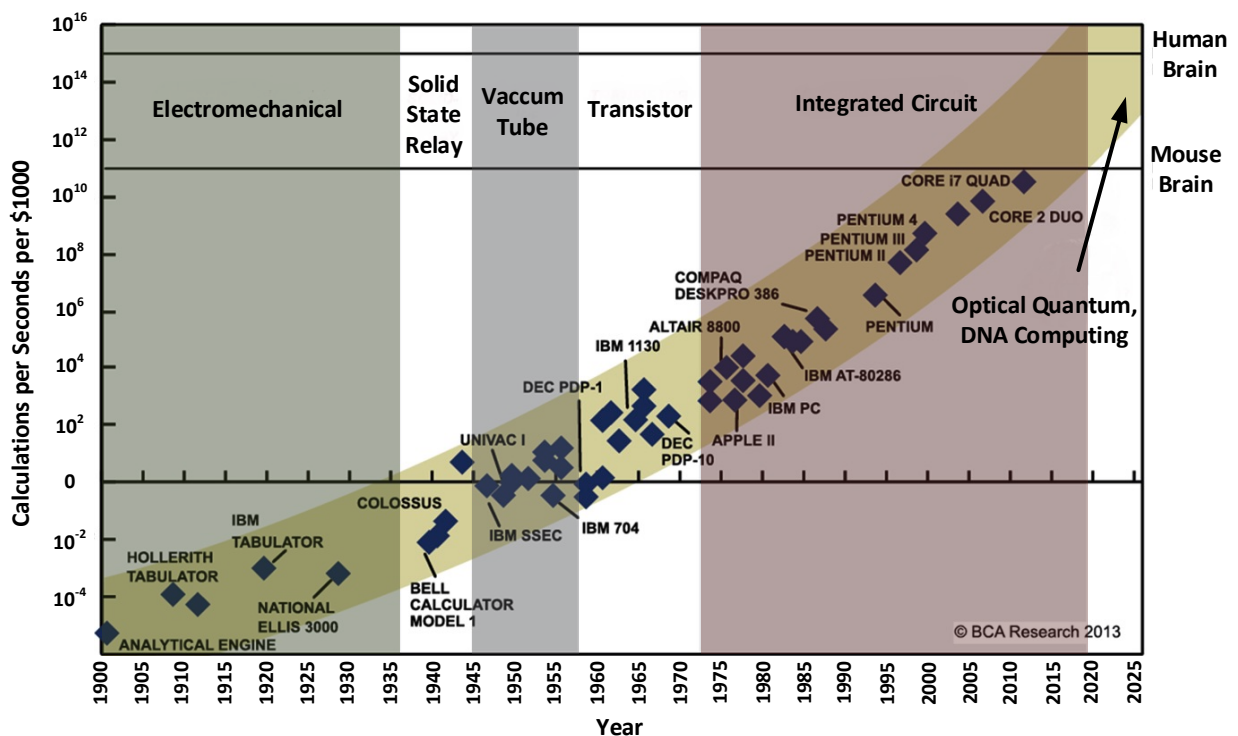


Fig. 1-3 Moore's law and integrated circuit scaling [2]

Table 1-1 Danard and interconnect scale factors

<i>Denard Device Scaling</i>		<i>Interconnect Scaling</i>	
<i>Parameter</i>	<i>Scaling factor</i>	<i>Parameter</i>	<i>Scaling factor</i>
<i>L, W, t_{ox}</i>	$1/\kappa$	<i>L, W, h</i>	$1/\kappa$
<i>Voltage, current</i>	$1/\kappa$	<i>Resistance</i>	κ
<i>Capacitance</i>	$1/\kappa$	<i>Capacitance</i>	$1/\kappa$
<i>Delay</i>	$1/\kappa$	<i>Delay</i>	1
<i>Power</i>	$1/\kappa^2$	<i>EM capability</i>	$1/\kappa^2$
<i>Density</i>	κ^2	<i>Density</i>	κ^2

Unlike digital circuits, analog circuits do not scale efficiently and suffer from low analog headroom (due to smaller supply voltages), increased variations in threshold voltage (V_t) (due to shrinkage in device sizes), smaller overdrive voltage (due to more scaling in supply voltage rather than V_t , $V_{dd}-V_t$), and etc. However, the analog blocks, e.g. RF blocks, power management units, sensors, actuators are as important as the digital systems on SoCs, because they are the interfaces to the real world [5]. Fig. 1-4 shows CMOS scaling over the past decades and its projection based on International Technology Roadmap for Semiconductors (ITRS) report [8].

Despite the analog design challenges of newer advance technology nodes, analog blocks are expected to be migrated in order to be integrated with digital designs that enjoy the scaling benefits. Thus analog designs also move towards smaller technology nodes, albeit with great design costs [6]-[7].

In summary, analog and digital integration should not be ignored, due to the benefits it has to offer, but migrating designs is not trivial in advanced technology nodes. There is a tradeoff between the performance of a design and its efforts and cost.

1.3. More than Moore

As mentioned, the semiconductor industry started following Moore’s law in 1975. Transistor feature size started to shrink over time, leading to an exponential growth in the number of components on the same die, and the level of integration. However, Moore’s law is predicted to slow down due to physical limitations. Therefore, a new era called “More-than-Moore” has started, where more functionality is added to devices that do not scale according to Moore’s Law [8]-[12].

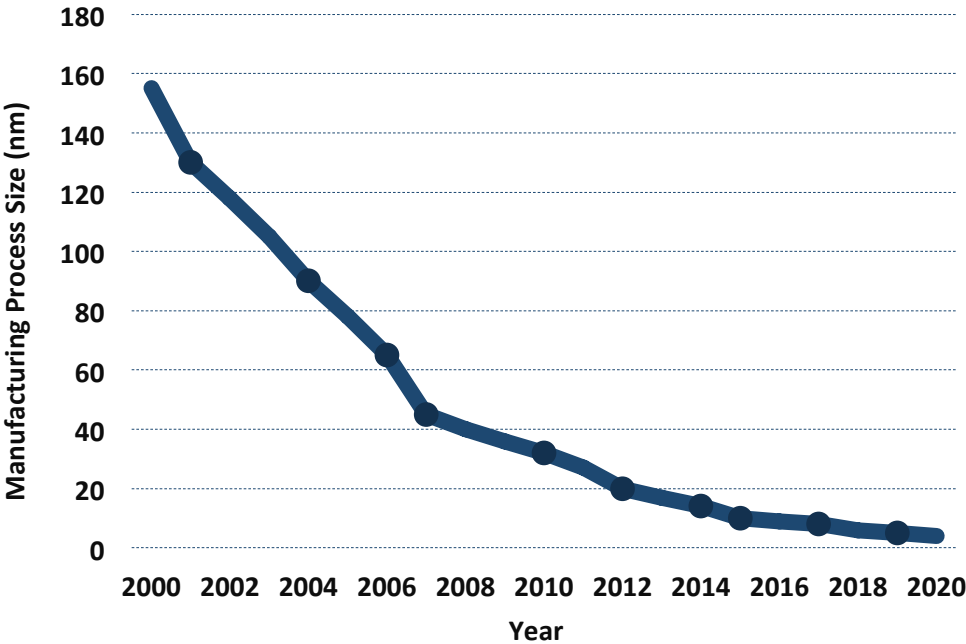


Fig. 1-4 CMOS transistor scaling roadmap [8]

Fig. 1-5 illustrates the International Roadmap Technology for semiconductors, which is the miniaturization of digital functions (“More Moore”) and functional diversification (More than Moore) [8].

For example, in a design approach called digitally assisted analog design, digital blocks that benefit from scaling are used in order to simplify the critical analog blocks that do not easily scale [13]. High-resolution data converters and high performance RF/analog front-ends often utilize this approach.

1.4. Bell’s law

Gordon Bell defines a computer class as a set of computers with similar cost, programming environment, network, and user interface, where each class undergoes a standard product life cycle of growth and decline. Based on prior market trends, a new computer class has

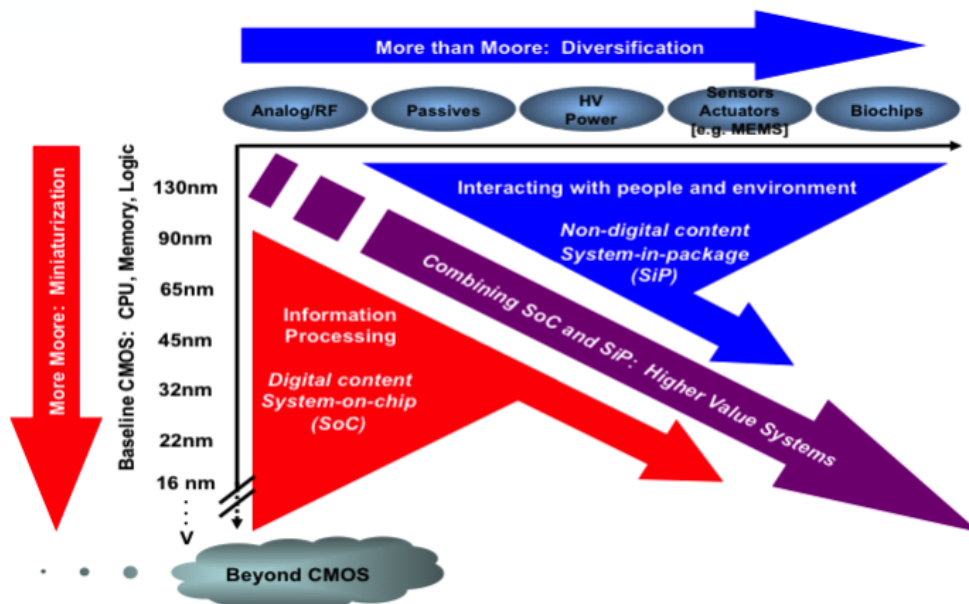


Fig. 1-5 Roadmap for semiconductors: miniaturization of the digital functions (“More Moore”) and functional diversification (More than Moore) [8]

come into existence approximately every decade, and each successive class has had a 100x reduction in volume (See Fig. 1-6) [16]. Each successive class has resulted in a reduction in unit cost and an increase in the volume of production [17]. Various applications, such as sensing, wireless communication, digital identification, and un-obtrusive surveillance, have driven the computing devices to a more compact regime with higher production volumes compared to prior computing classes. Wireless sensor networks (WSNs) are perceived as the next big step in this decades-long trend toward smaller, ubiquitous computing. They are projected to reach quantities of 1000 sensors per person by 2025 [18]-[19], [24].

1.5. Digitally assisted analog designs

As mentioned, digitally assisted analog refers to a set of techniques that help the analog designs to migrate to newer technologies and are mostly used in high-resolution data converters and high performance RF/analog front-ends. They involve automatic analysis

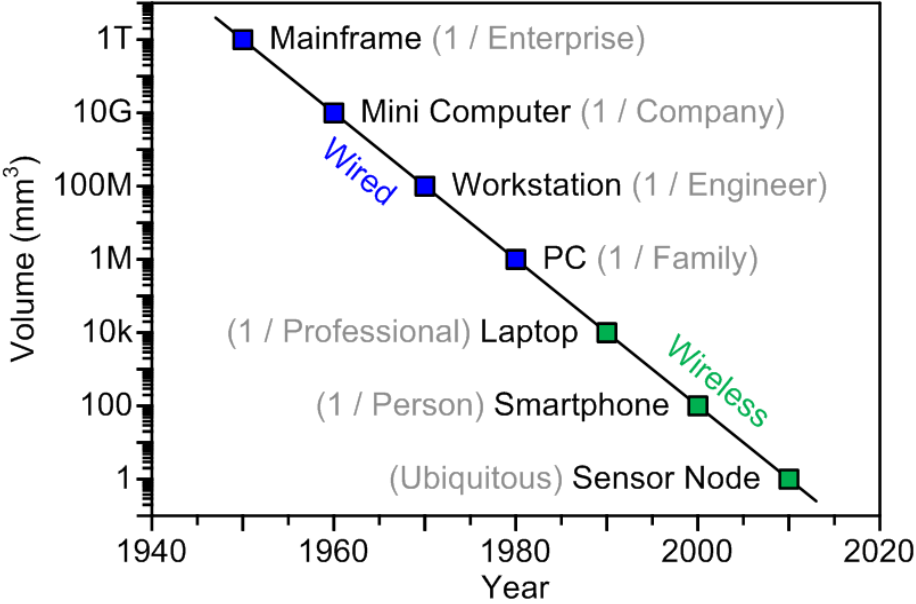


Fig. 1-6 Bell's Law [77]

and correction of analog circuit variations by employing digital circuits. For instance, calibration techniques for blocks that rely on matching (such as data converters) is a common example of a digitally assisted analog technique; an appropriate signal-processing algorithms are applied that runs before or at the same time of (in the background) or periodically through the analog block's normal operations to correct and compensate the deviations [7], [13] -[15].

Digital calibration techniques may be employed in various scenarios: tuning of analog filters, centering frequency range of voltage-controlled-oscillators (VCOs), DAC offset compensation, and tuning different parameters in RF/analog blocks (L, C, Gm, and etc.) Calibration is done at the blocks level (e.g. DAC, filter, and etc.), however there are other digitally assisted solutions for SoCs where the whole system (analog blocks as well as the

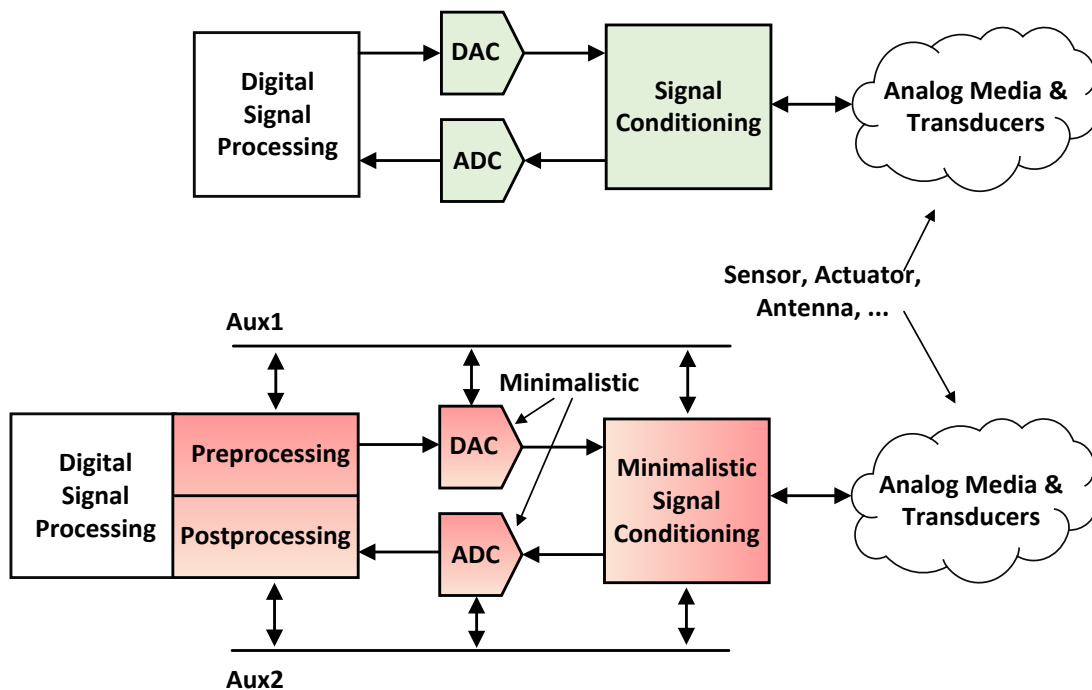


Fig. 1-7 Two different approaches for interface circuits: (a) precise implementation, and (b) digitally assisted implementation incorporating minimalistic interface components and additional pre- and post-processing units [14].

digital parts) is embedded on one chip, e.g. feedback loops that can estimate the quality of the signal and adjust the parameters in order to achieve the desired performance level [7][6].

1.6. Internet of Things

The new class of ubiquitous computing will give birth to what is dubbed as the Internet of Things (IoT) [21]. IoT devices will penetrate in every aspect of our lives and they will have the ability to sense, process, and communicate (Fig. 1-8).

The progression of IoT devices in our daily life will be feasible only if device technologies

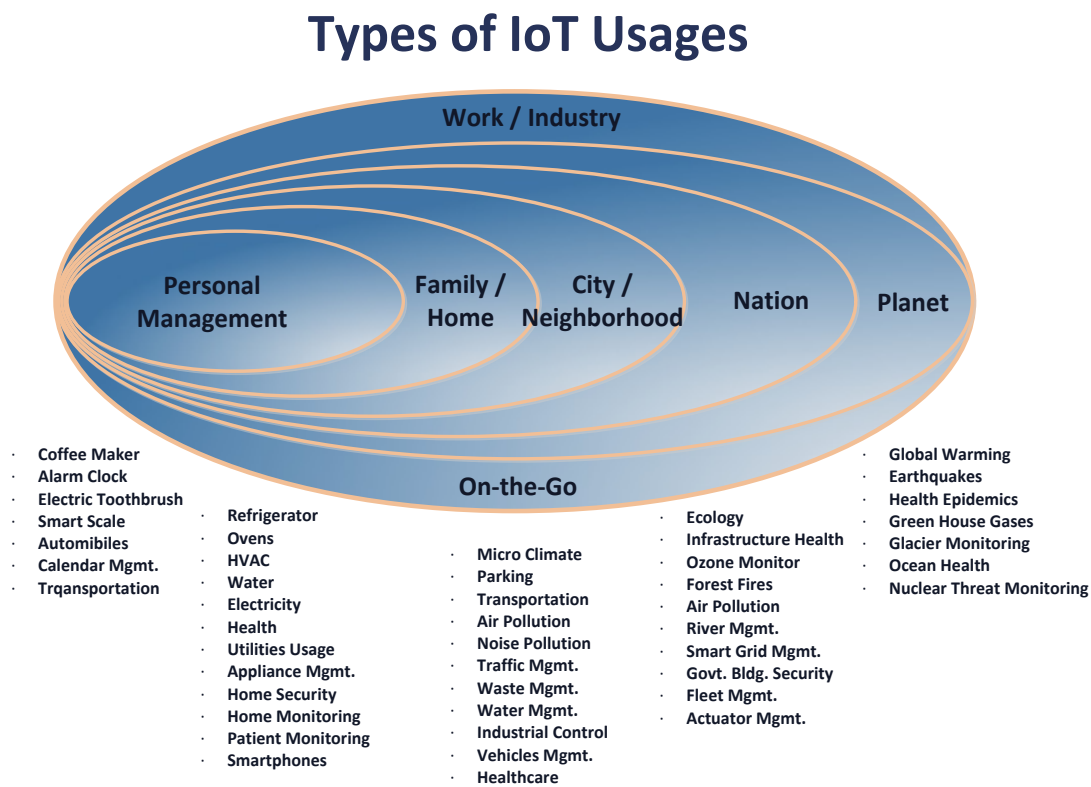


Fig. 1-8 Internet of things (IoTs) [20]

facilitate the wide spread proliferation [79]. This trend is predicted with different companies as listed below:

- Intel predictions 50 billion connected devices by 2020 [22].
- Cisco predicts 1 trillion connected devices by 2025 [23].
- Bosch predicts that the average person will carry up to 1000 sensors by 2025 [24].

In order to make IoT a reality, significant innovations are still required in many areas such as wireless communications, embedded processing, sensing, data management, software, and etc. as well as more efficient and cost effective applications. But it seems that the critical bottleneck in this process is the hardware due to many reasons, some of which are listed below [25]:

- Power consumption and battery limitations (size and life time)
- Relatively high priced electronic devices; for example having 1000 devices requires

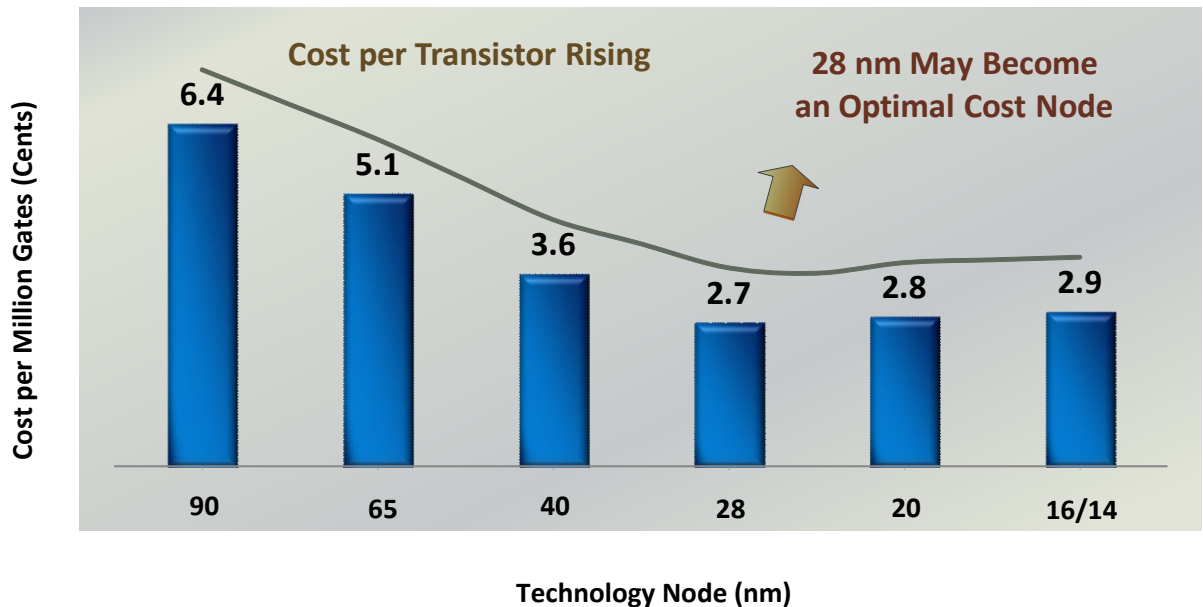


Fig. 1-9 Cost per transistor for deep nanometer CMOS technology nodes [74]

a significantly low unit price (cents per unit)

- Further miniaturization requirements in devices for most of the IoT applications
- Faster design time and time-to-market; which is a key factor to meet the high volume demand

1.7. Beyond CMOS

Smaller CMOS technologies provide higher levels of integration and processing speeds, but are not necessarily cheaper. In fact, the production cost of CMOS chips increase beyond 28nm [74], as shown in Fig. 1-9. In addition, the miniaturization of the newer technologies leads to increased wire resistances, higher delays, and heating problems. The latter is one of the biggest challenges of the current large scale computing systems. The required power

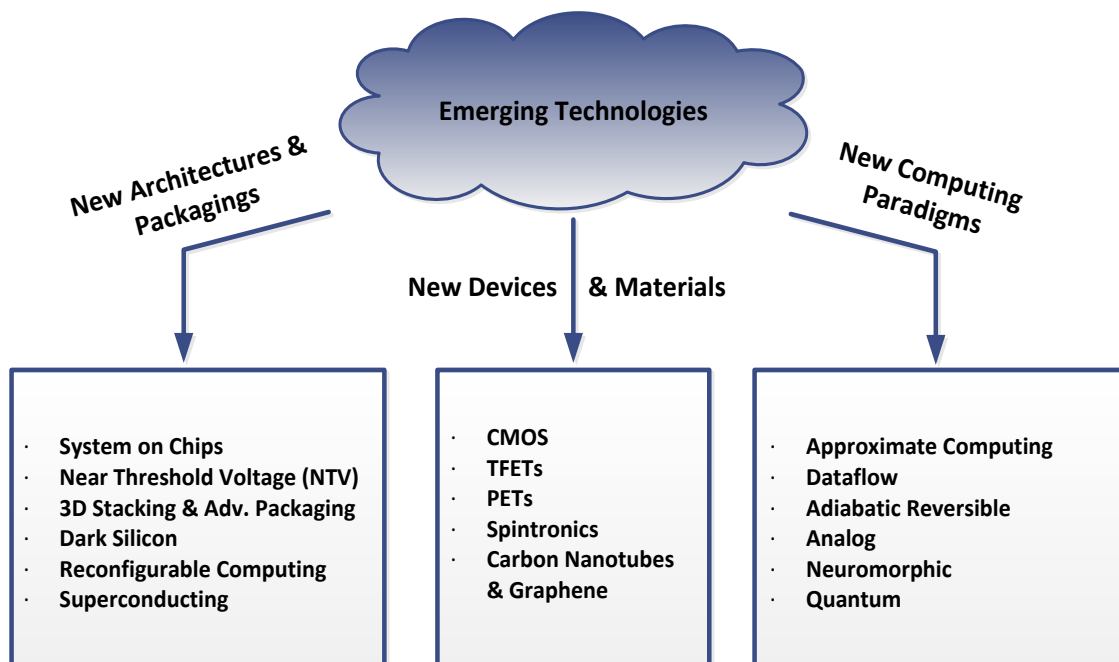


Fig. 1-10 Emerging Technologies [88]

for contemporary data centers is estimated to be around 12GW (which will be about 15% of the total power consumption of the world) [85]-[86].

Meanwhile the demands for more communication and computation systems increase exponentially and exceed the current technology node limitations. Therefore, beyond CMOS devices and novel computing systems must be employed. Fig. 1-10 shows several emerging candidates for new classes of devices, computing paradigms, architectures and packaging [88].

Superconductors are among the new technologies that are being re-evaluated for the future computing systems, mainly due to their high speed and low power characteristics. Since they let the electric current flow through the device without any energy loss, superconductors are extremely popular for low-power circuits. However, superconducting circuits suffer from scalability and to successfully implement contemporary computing tasks (VLSI-scale systems) their design process needs to be automated.

1.8. Contributions

In this dissertation we address some of the aforementioned challenges of contemporary circuit design. We improve the design process of RF/analog CMOS circuits that suffer from scalability, by introducing automated design and calibration methods. We also improved and automated the design process low power and high-speed superconducting circuits. Finally, we present novel architectures and functionalities that are suitable for extremely small and low-power IoT devices.

A. Very Large Scale Analog (VLSA) Methodology

With the number of design rule checks (DRCs) and operation counts increasing exponentially, cost and time-to-market of fully custom layouts increase, and manual designs of big analog circuitry becomes infeasible. In fact, only digital circuit designers exploit CMOS scaling to its full extent. Therefore analog design methodology should also change in a way that benefits from scaling. In other words, we need analog design techniques that remain compatible with nano-scale CMOS technologies. In this dissertation we introduce a design automation technique that employs digital automatic placement and routing (APR) tools and techniques to synthesize and lay out analog blocks along with digital blocks in a cell-based design approach, we call this approach very large scale analog (VLSA) methodology (Chapter 3).

B. VLSA Digital to Analog Converter

We also present a VLSA digital to analog converter (DAC) that benefits from the APR of unit-sized analog cells, along with digital standard cells. In this approach, analog complexity is relaxed in favor of automation, and as a result, the (re-)design cycle (including architecture and schematic designs, layout, parasitic extraction, and simulation) is dramatically shortened. The architecture of the proposed DAC is amenable to automatically place and route (APR) design flow (Chapter 4).

C. Optimization of Design Automation Techniques for the Reciprocal Quantum Logic Technology

The rapidly growing demands for wireless communications, computing units, and data centers, and etc. exceed the limitations of the existing technology nodes and systems. Superconductor technology is a potential candidate for future high speed, low power designs. However, to compete with CMOS circuits, they must scale up to large integrated systems. The integration could further be improved by automating the design and layout process of the superconducting circuits. In this dissertation we describe our contributions in automating the design process of the reciprocal quantum logic (RQL) technology, a promising superconductor technology. We design several blocks of a commercial Processor with the RQL technology while employing our optimization techniques (Chapter 6).

D. DSP and Communication Protocol for mm³-scale Wireless Sensor Nodes

In addition to the investigation of new technologies, several baseband DSPs, modulators and demodulators, and digital feedback gain controllers for digitally assisted transceiver's are presented in this dissertation. These architectures are suitable for low-power mm³-scale WSNs and enable high frequency transceivers to operate within the power constraints of standalone IoT nodes. In addition, we have developed a communication protocol suitable for these low power mm³-scale WSNs (Chapter 5 and Appendix).

Chapter 2

Very Large Scale Analog (VLSA) Design

Methodology

2.1. EDA for Analog Designs

Newer IC technologies enable better integration and possibly better performance, but they leave the IC designers with significant challenges, especially in the area of analog circuits where the most complicated ones are custom-designed.

In [26] they report that DRC complexity is increasing node over node, there are 25%-35% more DRC at each new technology node, and thus more operation counts for signing off the DRC decks that is mandatory by the foundries.

With the number of design rule checks (DRCs) and operation counts increasing exponentially (Fig. 2-1) [26]-[27], cost and time-to-market of fully custom layouts increase, and manual designs of big analog circuitry becomes infeasible. In fact, only digital circuit designers exploit CMOS scaling to its full extent. Therefore analog design methodology should also change in a way that benefits from scaling. In other words, we need analog design techniques that remain compatible with nano-scale CMOS technologies.

Electronic design automation (EDA) plays an essential role in today's electronic systems, especially very large-scale digital designs, e.g., processors with billions of transistors integrated on a chip.

The key step of EDA is the abstraction and reuse of regular common blocks such as standard cells, and at a higher level, arithmetic blocks. Analog circuits, on the other hand, typically require complicated design techniques with many optimization variables. This process is not easily automated, and results in critical, high-performance analog blocks still being designed manually [29]-[31]. In a typical ASIC chip, most of the area is dedicated to the digital portion of the design relative to the analog portion, while the majority of the design time is spent on the analog portion, since the digital portion exploits automation (Fig. 2-2) [32].

For decades, the separation of using EDA for digital circuits and doing full-custom design for analog circuits has worked. However, as mentioned earlier, the number of

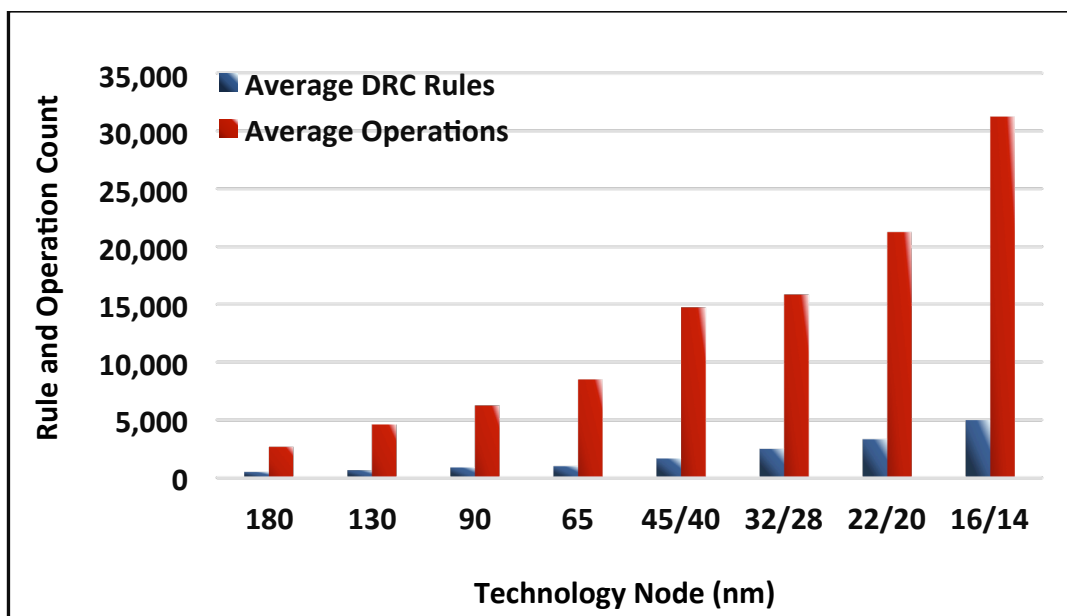


Fig. 2-1 DRC and operation counts vs. advanced sub-micron technology nodes

manufacturing design rules in modern CMOS processes is growing exponentially, and with it the time required to produce full-custom layout of high-performance analog blocks (e.g., the analog front-end of a digital-to-analog converter) is increased. This is especially true for designing large analog circuits that use numerous building blocks, e.g., high-resolution data converters. A 1-bit resolution increase in a digital-to-analog converter (DAC) or analog-to-digital converter (ADC), roughly doubles the number of blocks used in it, leading to a rapid increase in design time and effort when done manually. This favors a cell-based EDA approach that utilizes both digital and analog cells, and is much faster at producing an optimized layout. Cell-based EDA approach will be further discussed in Chapter 4.

2.2. Previous Works on Analog Design Automation

SoCs in advanced technology nodes (e.g., 14nm, 10nm, 7nm, and 5nm) include digital processors, memories, along with the mixed signal and analog blocks on one chip. This makes analog EDA tools a necessity for designing well performing circuits. The tools that generates the automated layout should be able to take care of all important analog designs

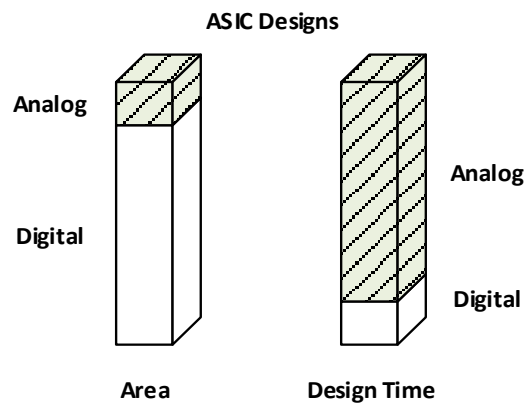


Fig. 2-2 Analog vs. Digital

parameters such as electromigration, signal and power routing to avoid DC drops, and self heating of power circuits [31]. There have been several studies on the analog design automation (DA) tools and techniques over the years.

In spite of all the efforts, existing tools do not cover all the steps required for analog designs [32]-[37]. They either focus on specific parts or require analog designer's interventions. Moreover they treat whole designs as single, large blocks (traditional design approach), making them impractical for complex systems. Complex SoC designs demand more hierarchical design levels in order to benefit from the divide-and-conquer strategy, as well as design re-use [35]

As mentioned, several tools have been designed for the analog automation purposes, each employing a different design methodology. For example, one classification is based on choice of topology derivation [35]:

- Topology selections: which is selected from a predefined library, based on the performance specifications provided, and provides the closest topology.
- Topology generation: a methodology that creates new topologies and explores the immense potential from low abstraction level. It starts from the small blocks and connects them in a bottom-up order and generates a new topology.

Another classification is based on the approaches for sizing and optimization of analog blocks, and/or their automated layout generation. These approaches are listed below:

- Knowledge-based approach

- Optimization-based approach
- Equation-based method
- Simulation-based method
- Learning-based method

2.3. Issues with Current Analog Design Automation Techniques

Analog design automation had progresses over the past decades, some of the analog DA tools are commercially available and they can handle device sizing, automatic layout, and even blocks designed with several transistors and components. However, they are not truly practical when it comes to the scaling and IoT applications, because analog designs lacks abstraction and cannot be automatically synthesized the same way as digital circuits. Some of the reasons are listed below [38]-[47], [79]:

- Nonlinearities, recursive iterations, optimization requirements, etc. which make the hierarchical abstractions challenging.
- Scalability challenges in deep submicron analog circuits, which make the analog designers to stick to the older processes rather than migrating to more advanced nodes. These challenges include reduced voltage headroom, exacerbated mismatches, increased leakage and second order effects in transistor behavior.
- PVT variations need to be carefully addressed in analog designs. Unlike in digital circuits, variation problems cannot be addressed only by overdesigning an analog circuit for the worst case. For example, an oscillator may not be stable over PVT variations even it is overdesigned.

- Analog designs are highly specialized and cannot be easily described with simple scripts and codes; they are highly dependent to the designer's skills
- Lack of innovations in analog blocks' architectures. Most of the analog blocks architectures are old and not amenable to abstraction

This research presents a different approach to analog synthesis. The design methodology is discussed in the following section in detail.

2.4. Proposed Very Large Scale Analog (VLSA) Design Flow

The design automation technique we introduce in this work employs digital automatic placement and routing (APR) tools and techniques to synthesize and lay out analog blocks along with digital blocks in a cell-based design approach. Digital APR techniques have been recently employed in several synthesized analog designs. Synthesized ADPLLs were first reported in 2011 [48] using only standard cell libraries. The authors of [49] used a similar approach for an ADC. More recently this approach has gained momentum, with standard cell based ADPLLs [50]-[51] and a blended approach of standard/custom cells to achieve higher performance [52]-[53]. In all cases a cell-based approach is taken, on the digital cell grid, with physical design automated by tools. We are calling this process Very-Large-Scale Analog (or VLSA). One benefit of such cell-based analog designs is that existing (commercial) VLSI tools can be easily used during their design process. Table 2-1 provides a qualitative comparison between the VLSA flow and those of VLSI and analog systems.

Table 2-1 Qualitative comparison between the analog, VLSA, and VLSI design flows

	<i>Analog (custom/ manual)</i>	<i>VLSA (automatic analog)</i>	<i>VLSI (automatic digital)</i>
<i>Design and optimization of building blocks</i>	All blocks are manually designed	A few custom blocks are manually designed and re-used	All blocks are designed automatically
<i>Block placement and routing</i>	Blocks are laid out manually	Critical blocks are placed manually; the rest are placed automatically; routing is done automatically	Blocks are placed and routed automatically
<i>Development time</i>	Months	Days/Weeks	Days
<i>(Re-)iteration time</i>	Days/Weeks	Hours	Hours

In this work we present the published results from a synthesized DAC that benefits from the APR of unit-sized analog cells, along with digital standard cells. In this approach, analog complexity is relaxed in favor of automation, and as a result, the (re-)design cycle (including architecture and schematic designs, layout, parasitic extraction, and simulation) is dramatically shortened. Furthermore, APR enables higher levels of integration and better scaling of analog designs, and thus benefits from Moore’s law without being hindered by the increasingly complex design rules. The drawback of this approach is the non-idealities and the mismatches introduced by (almost) randomly laid out cells (see Fig. 2-3), for which we rely on automated digital calibration techniques to digitally correct them.

The design flow of a VLSA (shown in Fig. 2-4), as expected, borrows many aspects of a digital design flow. While the building blocks of a digital circuit are mainly standard cells, a VLSA circuit utilizes tightly coupled analog sub-blocks alongside the digital standard cells. These analog sub-blocks may be available from a pre-generated library, possibly from a previous similar design, or may be designed manually. In the latter case, it is important to keep the number of distinct analog blocks at a low number to minimize the manual steps of the design flow.

We refer to these analog cells as analog standard cells. Analog standard cells are custom designed once and are reused in bigger designs. During their manual design step, these cells are fully characterized in terms of area, power, and variations; in the same way that

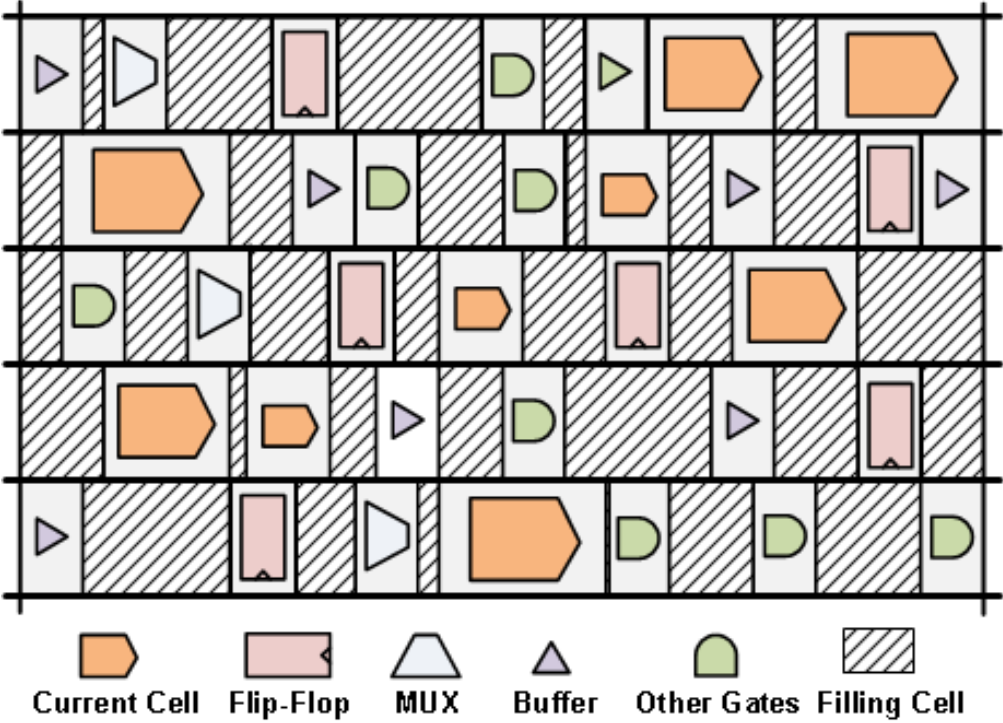


Fig. 2-3 Tightly coupled APRed digital and analog blocks

digital standard cells are designed and characterized.

Once all the analog sub-blocks are designed, they are added to a library of analog cells. The analog portion of the circuit, which utilizes only the pre-designed cells, is then expressed as a high-level structural Verilog. This is basically a high-level structure that shows the connections between the cells. The digital portion of the circuit, is usually expressed via a behavioral Verilog, and is converted to a gate-level using existing digital synthesis tools, e.g., Synopsys's Design Compiler. At this point, the analog and digital gate-level descriptions are automatically placed and routed via APR tools, such as Cadence's Encounter. Finally, existing CAD tools deal with post-APR tasks such as timing, filling, power calculation, floor-planning, etc.

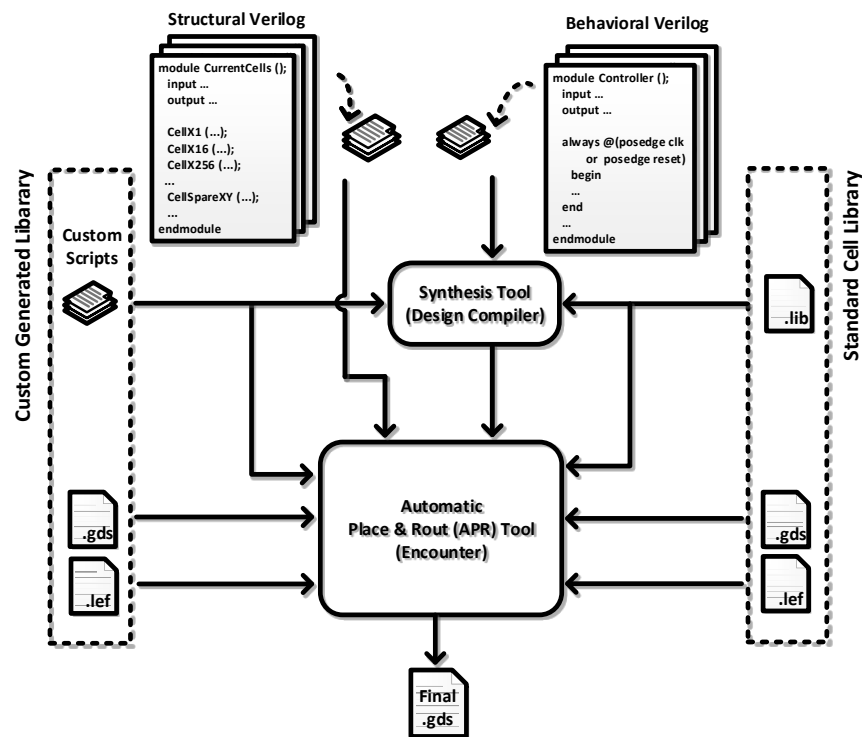


Fig. 2-4 Very large-scale analog (VLSA) synthesis design flow

The biggest advantage of the VLSA design flow is that the manual tasks are limited to the design of relatively small, reusable analog cells, exactly like what is currently done for digital standard cell libraries. As mentioned earlier, these cells may already exist in a library of earlier designs, or may require simple modifications of the library components. Analog cells are integrated on the same grid as the digital cells, allowing one APR tool to automate the physical design. Another advantage of VLSA is that many of the source codes and scripts used in the design flow can be re-used. By eliminating many manual steps, in particular the manual physical design, the (re-)design cycle of analog circuits in a new technology is significantly shortened. For instance, a small design change in analog circuit can lead to significant changes at its top-level layout, and hence may require a significant amount of design time.

In our VLSA approach, however, the final layout can be obtained within a few hours. Another benefit is the fact that digital circuitry is used as much as possible because it directly integrates with the analog cells and circuits, which allows better scaling and less layout constraints. This allows a higher level of integration, reducing area and cost. On the other hand, the automatic placement of many analog and digital cells leads to a somewhat random ordering of the cells, as opposed to a full-custom design where the blocks are usually manually placed in order. This leads to uneven wire delays and timing mismatch between various analog and digital signals. As a result, the final design may display unwanted non-linear behavior. The unwanted behavior is corrected after fabrication using digital calibration techniques. Some of these techniques are discussed in Chapter 4.

Chapter 3

Digital to Analog Converters Overview

In this chapter, we present the basic concepts of digital to analog converters (DACs), as well as the specifications of an ideal DAC. A DAC converts a discrete (finite precision) set of inputs to a continuous time real world (analog) signal. Fig. 3-1 shows the transfer function of a DAC, which consists of a set of discrete points that ideally fall on a straight line. The analog output step between two successive points is called the least significant bit (LSB) [54].

Most of the DACs have digital input code and clock, in order to sample the data at specific time intervals. For example at one edge of clock (e.g. the rise edge), digit input code is sampled and DAC output is updated accordingly (shown in Fig. 3-2), then this output stays the same until the next sampling point (e.g. next clock rising edge). This sample and hold process leads to the quantization noise even in the ideal DACs [55].

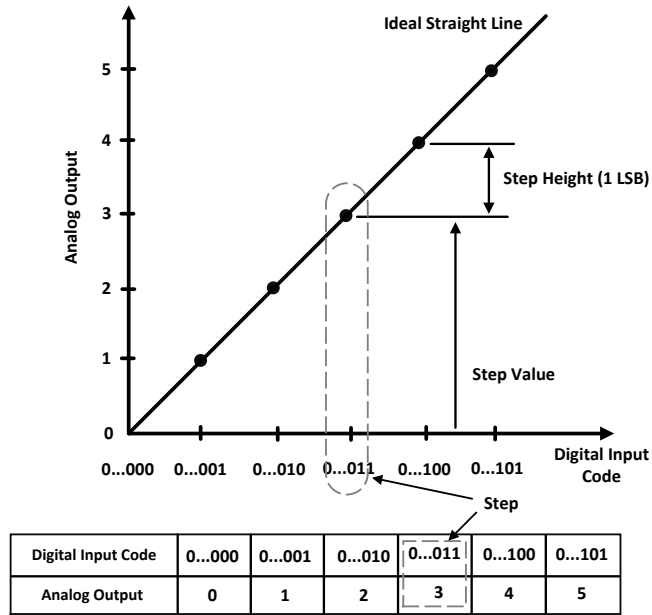


Fig. 3-1 The Ideal Transfer Function of a DAC [54]

In addition to the quantization noise, several other non-idealities exist in DACs due to process and random mismatches during fabrication, architectural limitations in DAC design, as well as designer's skill. Some of these non-ideal behaviors are described in the following sections.

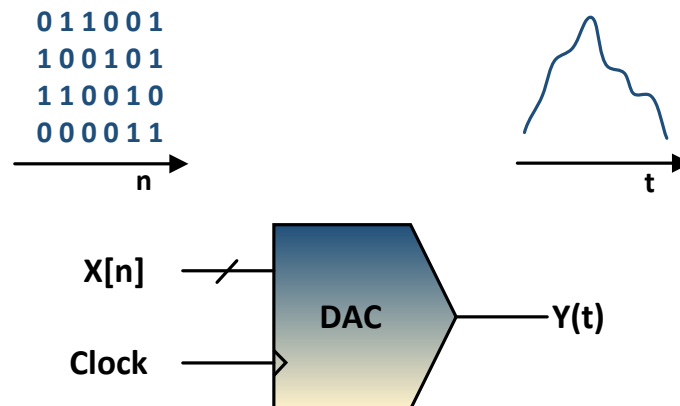


Fig. 3-2 Basic model of a DAC, with inputs for data and clock, and an analog output [56]

3.1. Static Behavior

A. DNL and INL

Differential nonlinearity (DNL) error is the difference between the actual and the ideal LSB values (example shown in Fig. 3-3). The DNL error is zero if the step height exactly equals to 1 LSB. DACs can become non-monotonic if the DNL errors exceed 1 LSB. In a monotonic DAC, an increase in the magnitude of the input always leads to an increase in the magnitude of the output [54].

The deviation of the actual transfer function from the ideal straight line (in Fig. 3-4) is called integral nonlinearity (INL) error. This straight line is usually the line that connects the lowest point of the DAC output to its highest point [54].

B. Full scale current and maximum output swing

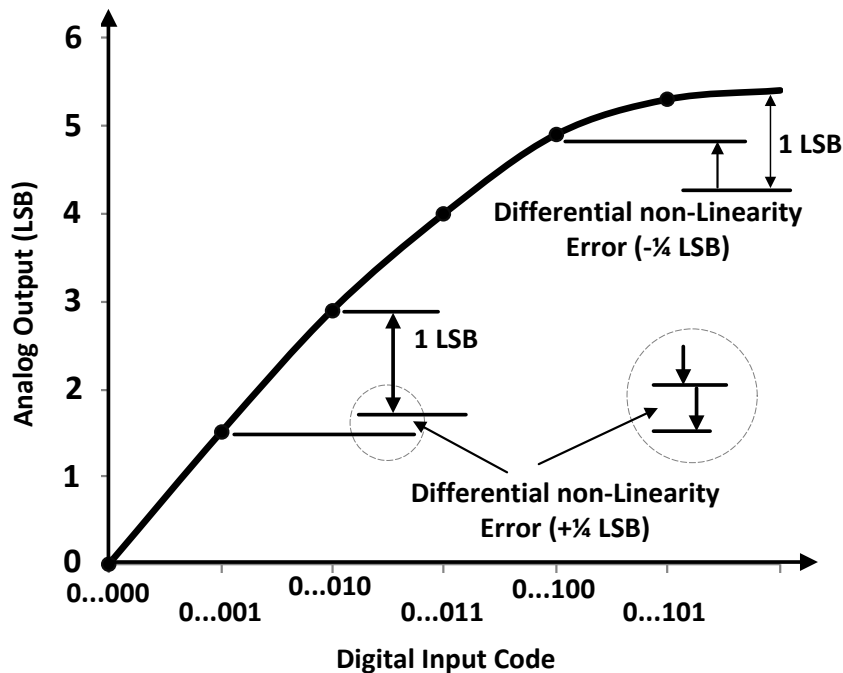


Fig. 3-3. Differential nonlinearity (DNL) in DACs [54]

The next static factor in characterizing DACs is the full-scale current and the maximum output swing. Most of the DACs deliver current to the output load, and maximum output current defines the maximum delivered power. The maximum current usually limits size of the load (R_L). If the R_L is too large, DAC cannot drive it properly [56].

3.2. Dynamic Behavior

In order to characterize dynamic performance of a DAC at high frequencies, output spectrum of the analog output should be studied. At high frequencies, frequency domain response provides a clearer picture [56].

A. Maximum sample rate

Maximum sample rate (F_s) is an important factor in characterizing a DAC. Maximum

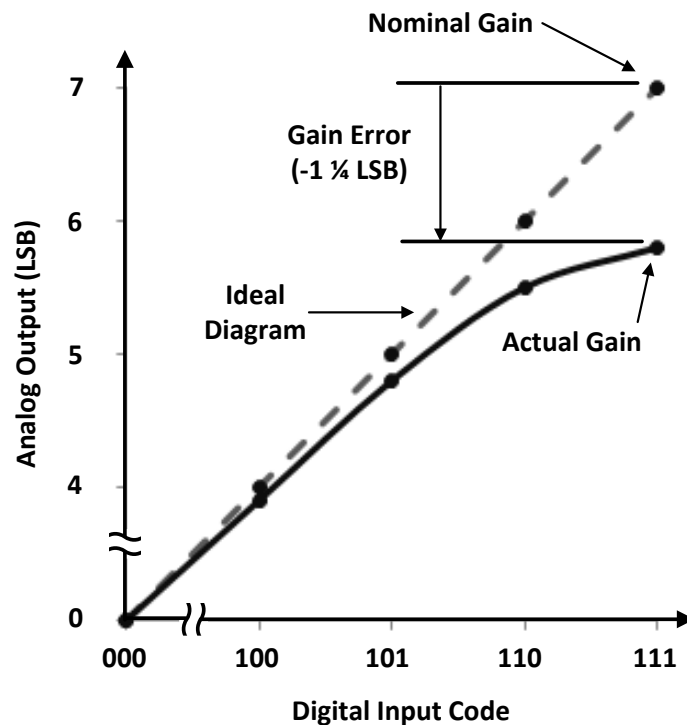


Fig. 3-4. Integral nonlinearity (INL) in DACs [54]

sample rate is the fastest clock rate that can be applied to a DAC before it fails meeting the desired specifications. In Nyquist rate DACs, output spectrum stretches from 0 to $F_s/2$.

B. Nonlinear distortion

INL and DNL are the nonlinearity factors considered at low frequencies; these parameters are more important when acquiring high resolution is more important. Nonlinear relation between input and output of a DAC can be studied in the frequency domain as well; especially when achieving high-speed sampling rate is desired. At higher frequencies dynamic nonlinearities resulted from nonlinear capacitances, are probably dominant.

Spurious free dynamic range (SFDR) is a popular specification for DACs, it is the power of the fundamental over the power of the largest in band spur; this spur could be originated as a harmonic or from external sources, or mixing products (Fig. 3-5).

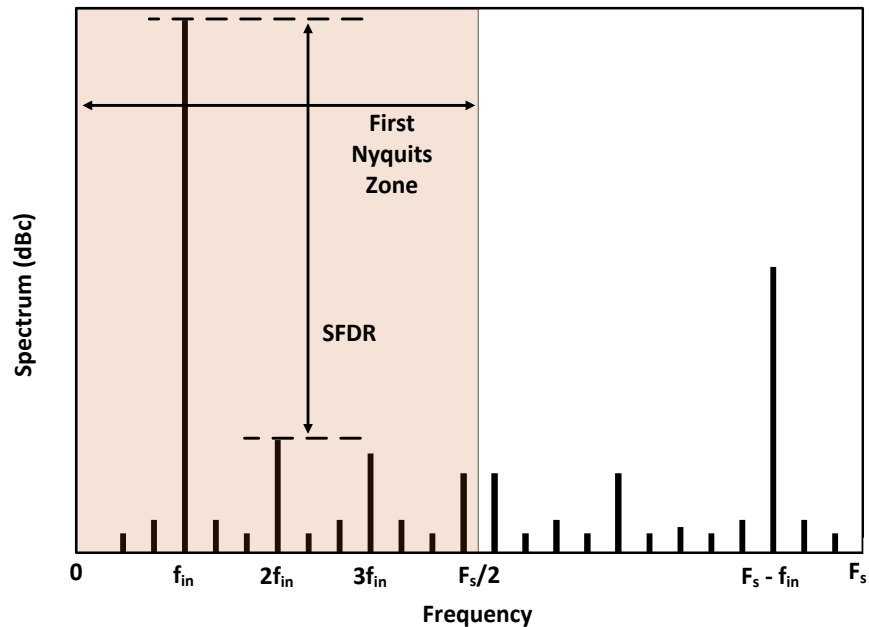


Fig. 3-5 Example output spectrum of a DAC [56]

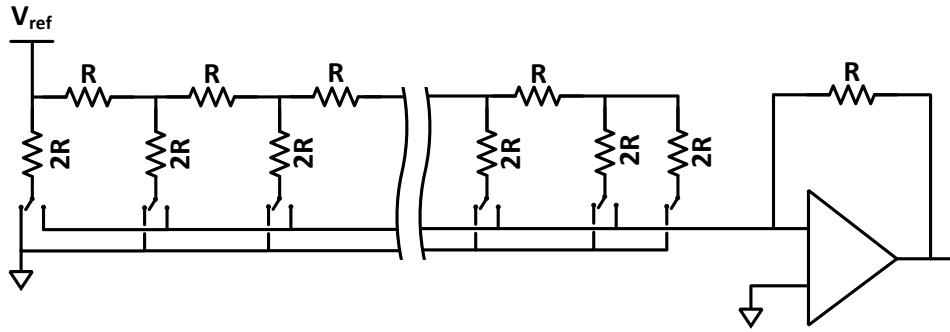


Fig. 3-6 R-2R ladder DAC [56]

3.3. Popular DAC Topologies

In this section, some of the popular DAC architectures are studied.

A. R-2R Ladder DACs

R-2R ladder DAC (shown in Fig. 3-6) consists of a resistor network that acts as a current divider. Switchable current sources, along with the resistor network sets the output value

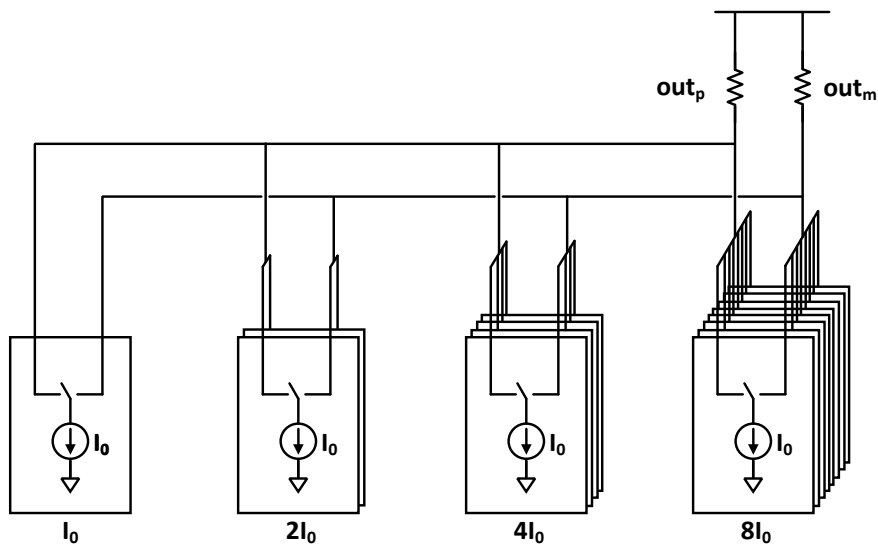


Fig. 3-7 A current-steering DAC architecture [56]

based on the digital input code. In this DAC architecture, by laser trimming the resistors after fabrication, high accuracy can be achieved. R-2R DAC was more popular years ago, however due to the design limitations, over the time this architecture has been replaced by some other DAC architectures. Some of the limitation factors are [56]:

- On-chip resistors are big.
- They do not match well after fabrication.
- R-2R network suffers from parasitic switch resistors.

There are only a few applications for this architecture these days.

B. Current-Steering DACs

Current steering DAC architecture is a popular architecture that can achieve high sampling rate, and does not require high-speed op-amps; thus are commonly used in generating high-frequency signals. Fig. 3-7 illustrates the current-steering DAC design; it consists of equally (and/or binary) weighted current cells. And it works by summing the current of each current cells at the output [56].

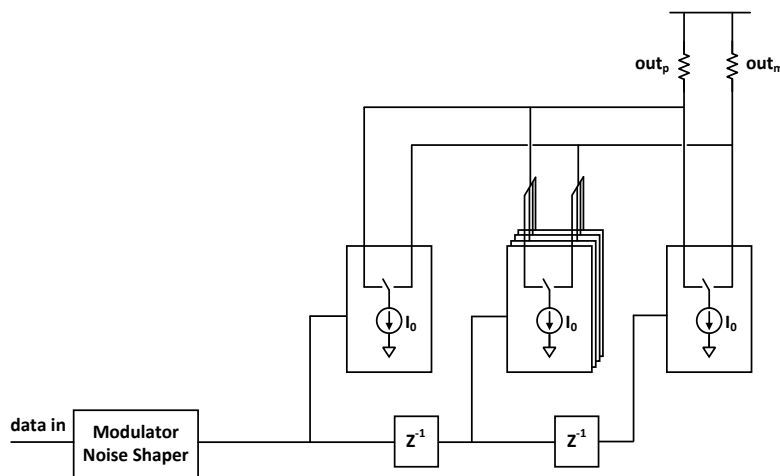


Fig. 3-8 Oversampling DAC with semi-digital reconstruction filtering [56]

C. Oversampling DACs

In oversampling DACs, by oversampling the digital input and leveraging the noise shaping, a high SNR, and as a result high resolution DAC is achieved. Noise shaping is possible by clocking and sampling the data at a much higher speed than the signal bandwidth [56]. Ratio of sampling frequency to twice the signal bandwidth is called oversampling ratio (OSR). In this DAC architecture (shown in Fig. 3-8), by sacrificing the signal bandwidth, high resolution DAC can be achieved without precisely matching the components.

D. Charge Redistribution DACs

Charge redistribution DACs architecture is so similar to the resistor based DAC design, except we have capacitors instead of resistors. Fig. 3-9 shows one example of this DAC design. Different clock phases are used in this topology for resting the nodes and redistributing the charge [56]. Capacitors are so bulky in this design and they occupy large areas; matching these capacitors from the largest to the smallest, in order to achieve high resolution, is not trivial either.

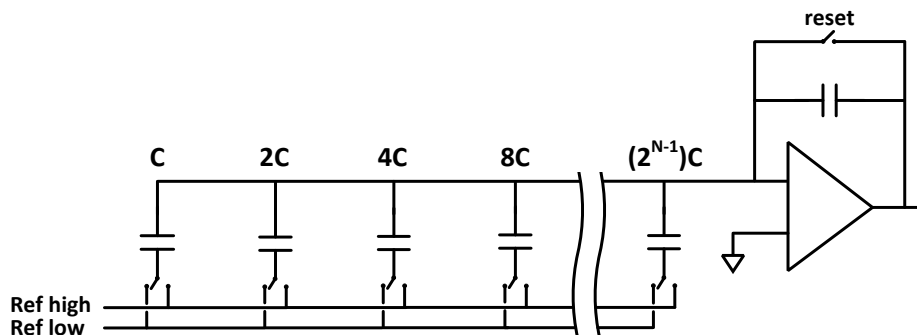


Fig. 3-9 Charge-redistribution DAC [56]

Chapter 4

VLSA DAC

This Chapter presents our VLSA DAC design and measured results of its fabricated chip.

4.1. High-Level Architecture

We employ a current-steering DAC architecture whose speed puts it among the most widely used [57]-[61]. Since it consists of many current cells, this architecture benefits from the VLSA design flow discussed in Chapter 2. The current cells are the only analog blocks, from the DAC core, that are designed on a standard cell grid and then automatically laid out with digital cells. Our design approach is to use digital circuitry as much as possible, and use analog blocks only when needed. This allows us to express the majority of the circuit at a behavioral level using Verilog. The analog parts consist of regular blocks that are expressed by structural Verilog to instantiate and wire many unit analog cells. As a result, our DAC design is completely synthesized and laid out via existing CAD tools.

Fig. 4-1 shows the high-level architecture of our proposed 12-bit cell-based current-steering DAC. Double data rate (DDR), low voltage differential swing (LVDS) inputs are converted into full swing digital signals through pseudo differential LVDS receivers followed by synchronous multiplexers. Multiple pipeline stages are placed along the signal path in order to increase the operating speed of the DAC, as well as synchronize the digital signals. Our design employs look-up tables (LUTs) to control each current cell individually. The LUTs control the order in which the cells are activated and play an important role in the calibration step. A digital controller manages the entire system, and performs important tasks such as reading/writing from/into on-chip LUT, defining the required number of pipeline stages, and setting the control bits that select different paths for the input signals and the clock.

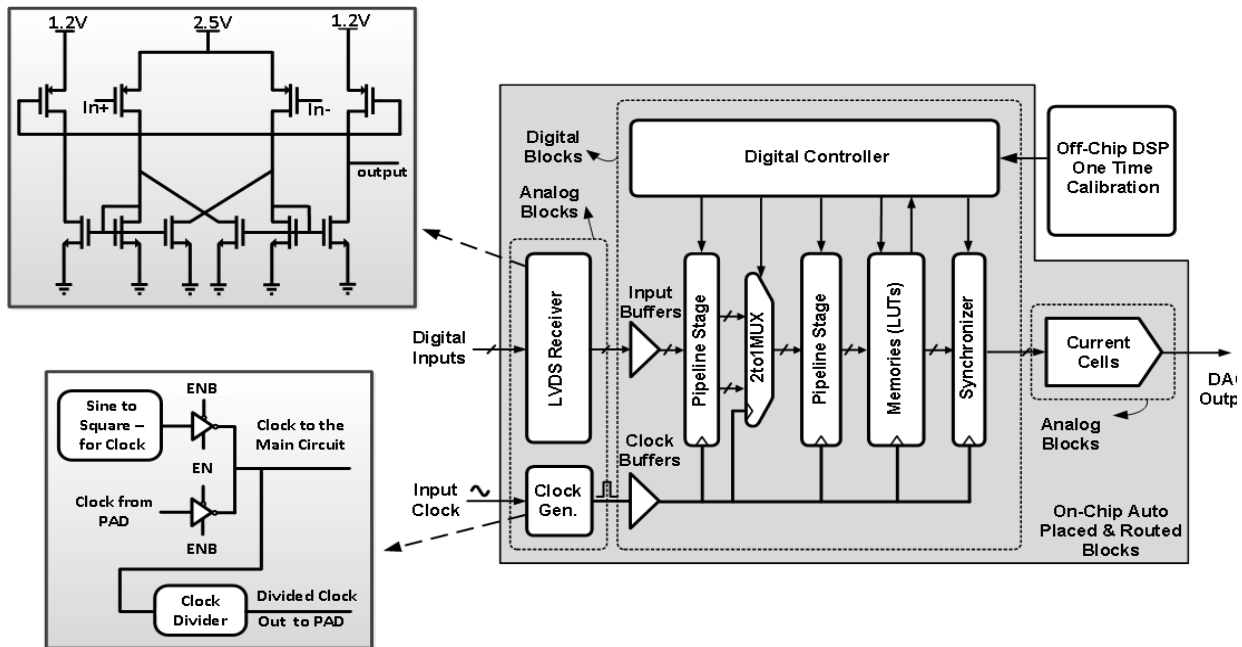


Fig. 4-1 High level architecture of the synthesized DAC

4.2. Current Cells

Main current cells are divided in three different banks with 15 cells each. The four most significant bits (MSBs) of the digital input control the bank with the largest cells, and the four least significant bits (LSBs) control the bank with the smallest cells. Assuming the current weight coefficient of the smallest bank is 1, then the weight of the middle bank and the largest bank is 16 and 256 respectively.

Fig. 4-2 shows the design of the current cells. In addition to the main DAC cells, tri-state spare DAC cells are also placed for calibration purposes. They can be switched off, having no effect on the circuit, or they can be switched on, in which case they act as a regular current steering cell. LUTs control when these calibration cells are used on a per-sample

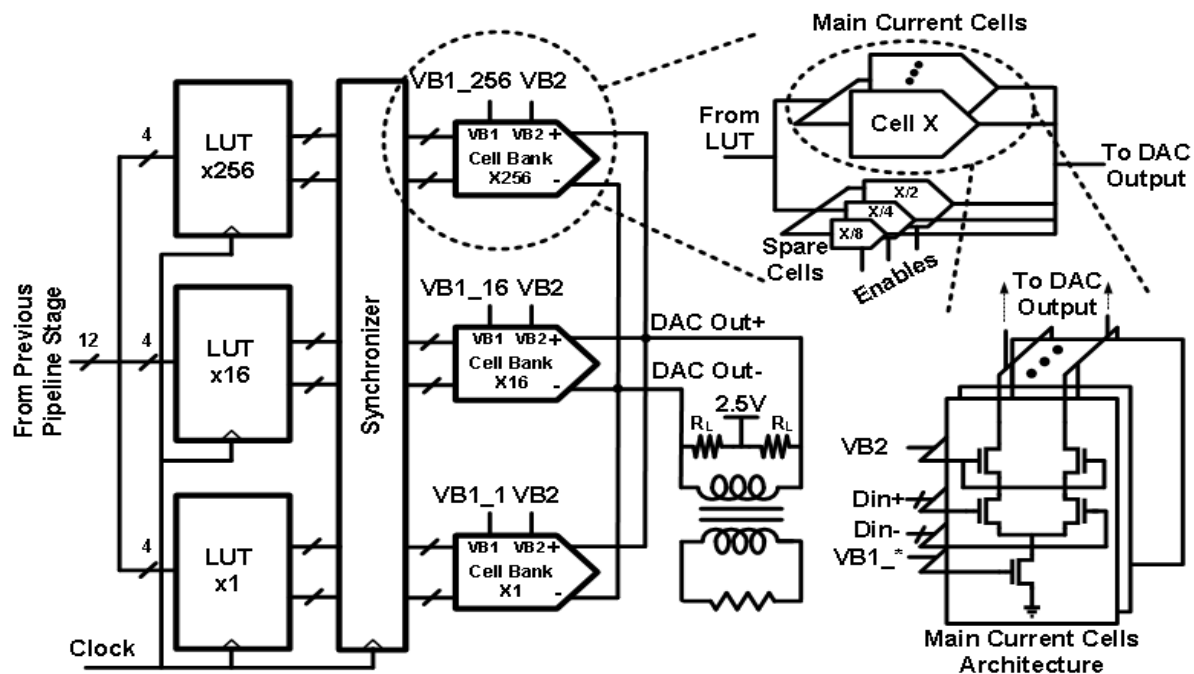


Fig. 4-2 DAC current cells and look up tables (LUTs).

basis. All the current cells, including the original and spare cells, are individually designed and laid out on the standard cell grid, and vary in size from that of a minimum-sized inverter to largest D flip-flops (in terms of length). For wire routing and pin spacing, we followed the standard set rules. Larger current cells (cells with larger current weight) that did not fit in the grid of standard cells (in terms of height) are broken down into several smaller cells with the same height of the standard cells, and during auto placement they are placed together. Local power routings for these cells are handled during the manual design phase and they are connected to the global power rails at the top level. After the careful manual design and characterization mentioned above, the cells are then abstracted and integrated into the EDA flow, via a structural Verilog description in which each cell is instantiated many times. The remaining parts of the circuit are expressed via behavioral Verilog and are synthesized along with the current cells.

Several scripts are used in this step for different purposes. First for placing the analog sub-blocks (of a large current cells that did fit in one height grid) in the right place relative to each other so they generate a true large current cells. Second, these scripts place all current cells close to each other, in order to reduce the effect of process variations on them. Third, these scripts handle the routing settings to/from the analog blocks, e.g., they set the wire width, length, and etc. considering the timing, loading and current requirement. These scripts are design-dependent not technology-dependent, hence they can be used in the design of such DAC in another technology. The scripts are the input resources for the APR tool, and all the placements and routings are performed automatically considering the constraints defined in the scripts.

The current cells are controlled by LUTs that can be programmed to activate different cells for each input pattern. The contents of the LUTs are changed during the calibration phase, in order to compensate for the non-linearities by activating better cell combinations and possibly utilizing spare cells. The one-time calibration process is described in section 4.4.

4.3. Design Trade-Offs, Number of Cells and Look-Up Tables

The main core of the DAC comprises many current steering cells. These cells can have different sizes and may have various architectures (see Fig. 4-2). The choice of architecture affects several aspects of the DAC, some of which cannot be easily quantified. Speed, accuracy, flexibility (for calibration), area, size of the on-chip memory (LUTs), and the number of the manually designed analog cells are the aspects that should be considered

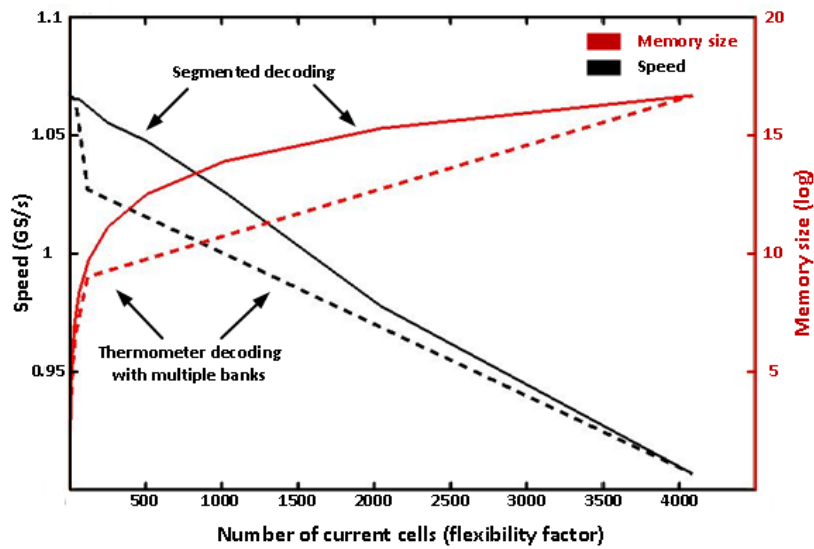


Fig. 4-3 Design trade-offs – Speed, and memory size vs. total number of current cells (calibration flexibility factor).

Table 4-1 DAC specifications with segmented decoding (binary+thermometer) of the input code, m represents the number of binary bits

<i>Number of binary codes (m)</i>	<i>Number of current cells</i>	<i>Maximum settling time (ps)</i>	<i>Memory (LUT) size</i>
12 (or 11)	12	936	12
10	13	938	19
9	16	938	58
8	23	938	233
7	38	939	968
6	69	939	3,975
5	132	942	16,134
4	259	948	65,029
3	514	955	261,124
2	1,025	975	1,046,531
1	2,048	1,023	4,190,210
0	4,095	1,103	16,769,025

when choosing a cell arrangement. We will explain these aspects through several examples.

Assuming that the target design is a 12-bit DAC, one example design is to employ 12 distinct binary-weighted current cells with weights $2^0, 2^1, \dots, 2^{11}$, and implement the whole DAC by using one of each. One benefit of this simple arrangement is that it is fast due to the small number of cells; hence lower device parasitic and wiring cap loading. The number of LUTs required is also low for the same reason. On the other hand, this approach requires 12 different cells that must be designed manually, and accurately matched. Since there is only one instance of each cell, it is not possible to use cells to cover each other; in other words, this arrangement provides no flexibility.

Another extreme arrangement for the 12-bit DAC is to employ 4095 current cells with 2^0 weight. This arrangement, also known as the thermometer coding, yields a slower DAC, but is very suitable for a VLSA design flow because (i) only one cell type needs to be designed manually and (ii) as many as 4095 cells needs to be laid out, so APR is the most efficient

Table 4-2 DAC specifications with dividing the DAC cells into different banks, n represents the number of banks

<i>Number of banks (n)</i>	<i>Number of current cells</i>	<i>Maximum settling time (ps)</i>	<i>Memory (LUT) size</i>
12	12	936	12
6	18	938	54
4	25	939	196
3	45	939	675
2	126	974	7,938
1	4,095	1,103	16,769,025

way of doing so. Furthermore, this arrangement provides maximum flexibility, since all the cells are identical and can cover each other. The main drawback of this arrangement is that it requires a high number of LUTs to control the 4095 DAC cells, assuming cells will be re-ordered based on some calibration mechanism.

An intermediate arrangement (between the two extremes mentioned above) would strike a balance between flexibility, speed, and the memory required. To find the optimum point, we considered several arrangements shown in Table 4-1 and Fig. 4-3. One approach is to have the m binary weighted cells for the lowest m bits, and use a thermometer coding for the high (12 - m) bits. Table 4-1 shows the properties of this approach for different values of m. Lower values of m provide better flexibility, but the memory required quickly increases as m decreases. We refer to this as Approach A. In this Approach the number of current cells and memory (LUT) size are defined using Eq. 1 and 2, respectively.

$$\text{Number of cells in Approach A} = m + (2^{(12-m)} - 1) \quad \text{Eq. 1}$$

$$\text{Size of memory in Approach A} = (m + (2^{(12-m)} - 1))^2 \quad \text{Eq. 2}$$

Maximum settling time defines the speed of the DAC, and is measured under the worst case (full-scale) output change, which is translated to change from all 0s to all 1s in the input digital code.

Another approach, called Approach B, is to divide the 12 input bits into n groups of size $12/n$ and use thermometer coding for each group. Table 4-2 shows the properties of Approach B for different values of n . In Approach B the number of current cells and memory (LUT) size are defined using Eq. 3 and 4, respectively.

$$\text{Number of cells in Approach B} = n \times (2^{\frac{12}{n}} - 1) \quad \text{Eq. 3}$$

$$\text{Size of memory in Approach B} = n \times (2^{\frac{12}{n}} - 1)^2 \quad \text{Eq. 4}$$

We observe that for $n = 3$, Approach B provides the same speed and flexibility as $m = 7$ in Approach A, but it requires less memory. Furthermore, only three different cells should be designed manually as opposed to the 6 manual cell designs in Approach A (with $m = 7$). In addition, the thermometer encoding is very suitable for the VLSA approach. So we chose Approach B with $n = 3$ because it has most of the benefits while meeting the memory requirements.

4.4. Calibration

As mentioned earlier, automatic calibration is a key step in the DAC's design process. In addition to the non-idealities introduced by cell mismatches (due to intra-die process variations and local random variations, as studied in [62]), the APRed layout also causes systematic mismatches in the interconnecting wires and further degrades the performance

of the DAC. These variations are estimated via Monte-Carlo simulations (for both pre- and post- layout designs). To compensate for these variations, we added flexibility and several degrees of freedom to the DAC, which enable us to experiment with several calibration techniques that are explained in Sections IV.A-D and compare their relative impacts on the performance. Section IV.E discusses how these techniques are used step by step. Note that the calibration techniques are not targeted at the schematic level; they address all the non-idealities introduced after APR and fabrication.

A. Gain Adjustment

Each DAC cell has an adjustable bias voltage that is applied to the tail transistors of the DAC cells. The bias circuitry is embedded in the design of each cell in order to reduce the coupled noise on the bias current. The bias voltage of the cells from the same bank are controlled together. This allows to adjust the relative weight between the cell banks. Even though the cells were designed to have the weights 20, 24, and 28, they may exhibit a different relative weight due to process variations. The bias voltage allows us to compensate the variations for each individual chip. The estimations for gain tuning range is based on the Monte-Carlo simulations.

B. Cell Reordering

The programmable on-chip LUTs provide several degrees of freedom, because they allow any combination of the current cells to be activated for any given input pattern within their bank. Due to process variations, as well as the non-idealities introduced by the VLSA approach, different cells may produce a slightly different current output. Thus, activating them in a fixed ordering can lead to a non-linear DAC behavior.

Each bank has 15 cells c_1, c_2, \dots, c_{15} , and their default activation ordering is to activate all the c_i 's with $i \leq k$, where k is the 4-bit input of the bank. However, given the non-identical behavior of each cell, there may exist a cell ordering that produces a more linear output. For example, we can activate all the c_i 's with $i > 15 - k$, for a given 4-bit input k . In the calibration phase, we search all the possible activation orderings, and choose the one with the most linear output for a ramp input.

C. Spare Cell Usage

As mentioned, spare current cells are employed in the DAC architecture for calibration purposes. These cells are also controlled through LUTs and can be activated for arbitrary input combinations. The spare cells are used to fine-tune the calibration step discussed in Section IV.B. The basic idea is to activate these cells in the non-ideal cases that cannot be fixed by the main cells. Another benefit of the spare cells is that they can completely replace a bad current cell. Spare cell architecture is shown in Fig. 4-4. The required number of

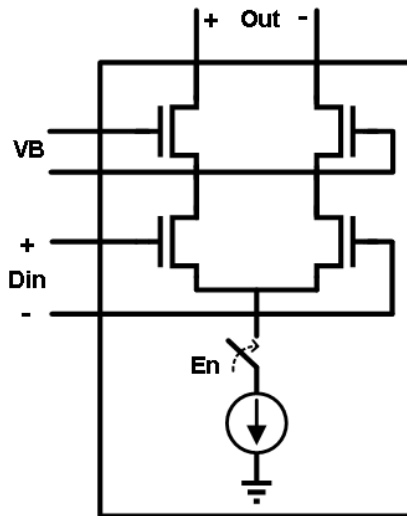


Fig. 4-4 Tri-state DAC current cell for calibration purposes.

spare cells is calculated based on post-layout Monte-Carlo simulation results of a DAC without any spare cells. This reveals the mismatch and process variations. Then we consider enough auxiliary (spare) cells to cover the variation range. This is done by overlapping the the spare cells with the existing DAC cells.

D. Code Swapping

The last calibration resource is code swapping, which allows the digital input combinations to be used as each other. The idea is very similar to the cell ordering technique discussed in Section IV.B, but it is applied to the digital input of the DAC. First, all the input combinations i.e., codes, and their corresponding output voltages are recorded in a table. Then, the table

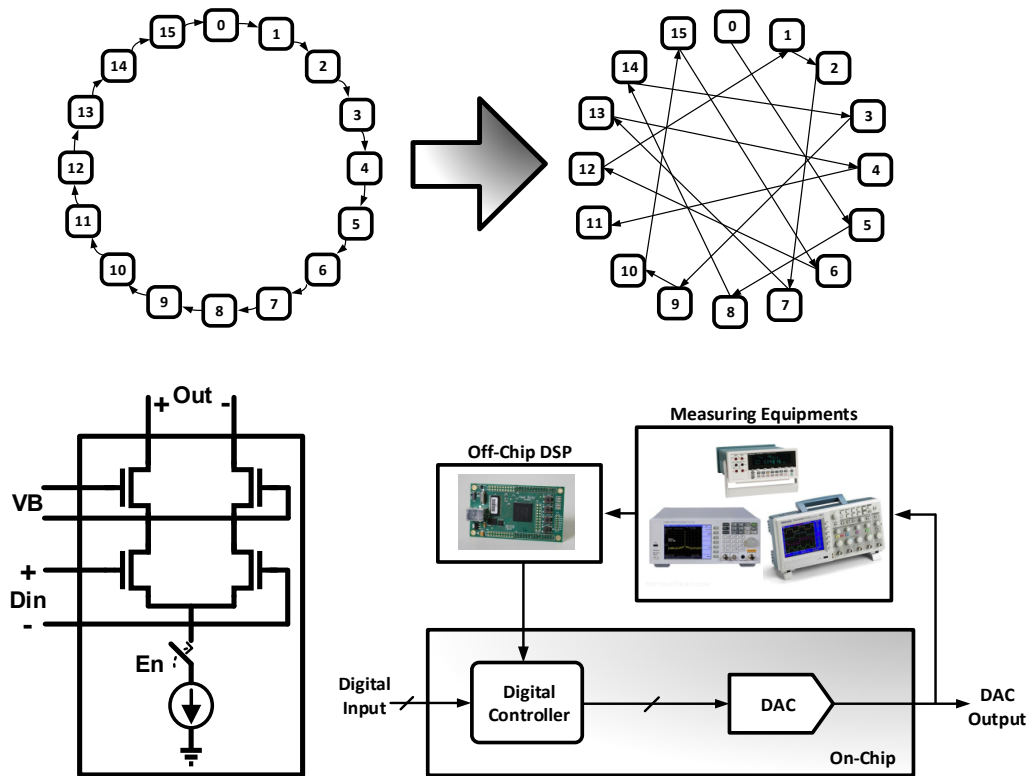


Fig. 4-5 Calibrations setup

is sorted according to the output voltages. Finally, a suitable ordering of the codes are selected that provide a linear and monotonically increasing output voltage. Once the best code ordering is decided, a digital circuit is used to swap the codes at run time.

E. Calibration Steps

The calibration process is done once per chip, at low frequency, and is also automated using a digital signal processor (DSP). Calibration setup is shown in Fig. 4-5. The first step in the calibration process is (fine-) tuning the current of each bank (gain adjustment); hence their values match with each other with the appropriate weights. Next, the DSP measures the output voltages of the DAC, compares with the target values, and adjusts the LUTs accordingly to improve its performance. The calibration algorithm first measures the DNL of the DAC at its default configuration, and then re-orders the DAC cells until the DNL can no longer be improved. It then targets the input combinations for which the maximum

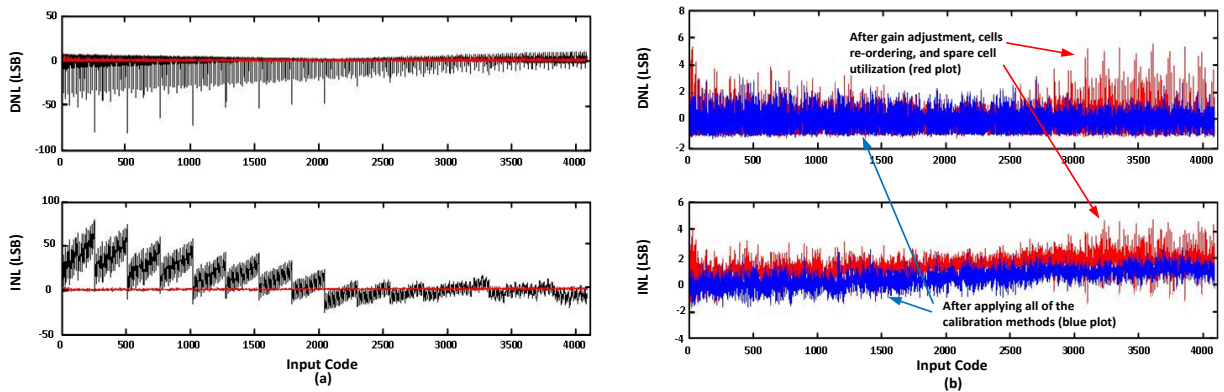


Fig. 4-6 DNL and INL measurement results before and after calibration – a) Before calibration, b) After calibration, first step: applying gain adjustment, cell re-ordering, and spare cells utilization. Second step: adding code swapping technique.

distortion has occurred and enables spare calibration cell(s) of the appropriate size to eliminate the distortion. The algorithm continues until no progress can be achieved in distortion improvement, or until it runs out of spare calibration cells. The final calibration step is input-code re-ordering (code swapping), which is handled in the DSP that provides the digital inputs for the DAC. In this technique, the input codes will be re-ordered in such a way that the DAC generates the most linear ramp output. Among all the calibration techniques, we found out that gain adjustment and code swapping were the most effective ones in improving the performance of the DAC.

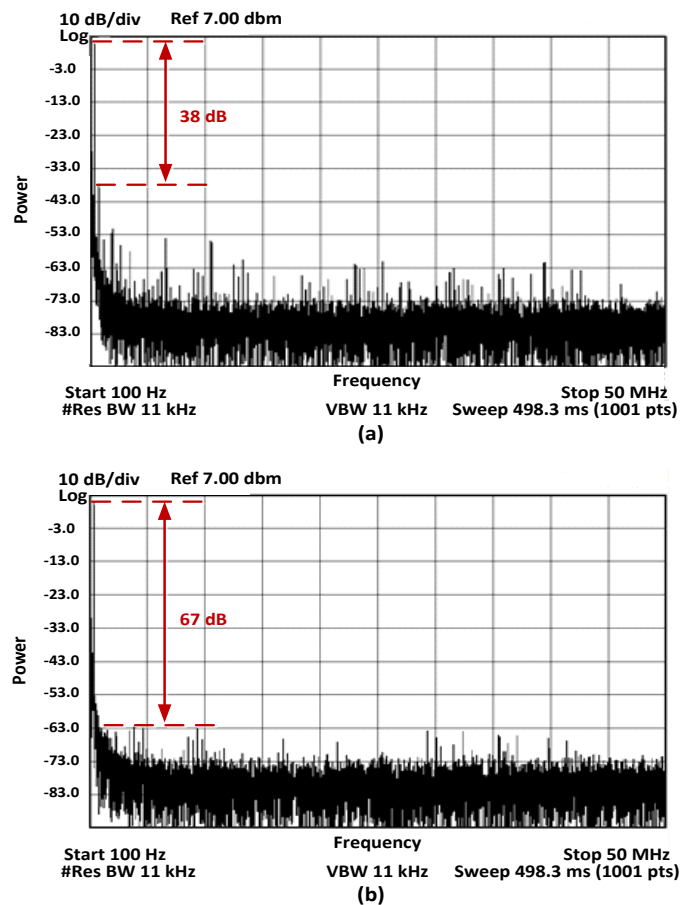


Fig. 4-7 DAC output spectrum for low input frequency before and after calibration.

4.5. Measurement Results

The DAC is fabricated in a 65nm CMOS technology, and operates at up to 250MS/s. Fig. 4-6 illustrates the INL and DNL measurement results before and after the calibration, respectively. The calibration results are seen in two steps in Fig. 4-6. The red lines show DNL and INL results after gain adjustments, cell re-ordering, and spare cell utilization. The blue lines are results after all the calibration steps. Before calibration, the DNL and the INL value ranges are $[-84.13, 9.79]$ and $[-24.4, 79.11]$ LSBs, respectively. After calibration, these ranges are reduced to $[-1.1, 2.4]$ and $[-1.97, 2.2]$ LSBs; the LSB size is $317\mu\text{V}$.

Fig. 4-7 shows the single tone measurement results of the DAC before and after calibration at low input frequency. This plot demonstrates a 29dB improvement in SFDR after the

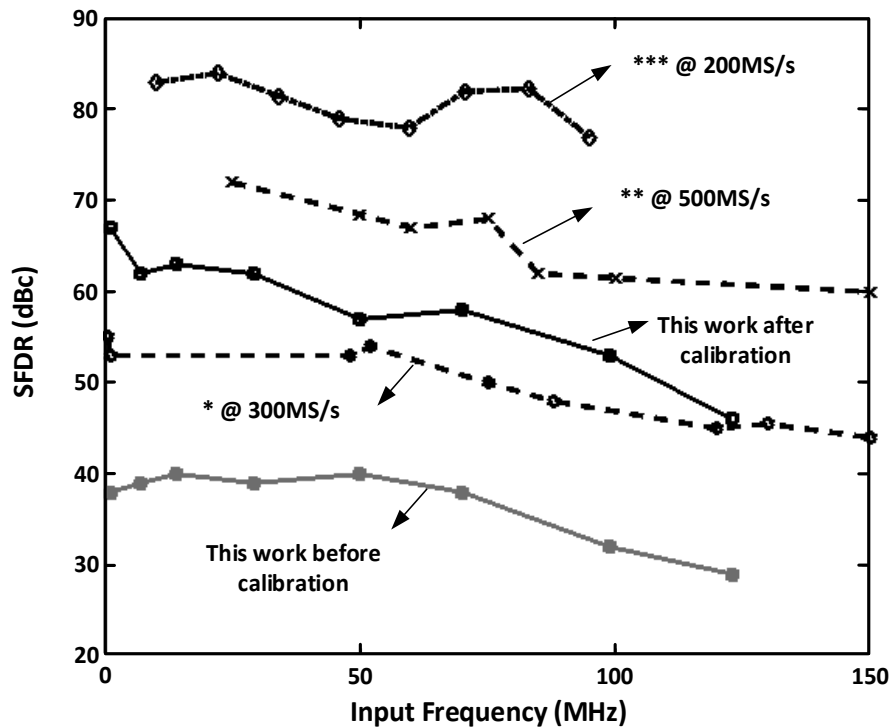


Fig. 4-8 SFDR measured results vs. input frequency and comparison with some recent works; *: [59], **: [60], ***: [61]

calibration. SFDR results for different input frequencies are shown in Fig. 4-8. In this plot, SFDR ranges from 67dBc to 46dBc across the Nyquist band, which is improved by an average of 22 dB after calibration.

The total synthesized area is 0.11mm², with only 32% occupied by active standard cells (0.035mm²) and 1.1% occupied by current cells (0.0012mm²); the rest is allocated to filling cells. Fig. 4-9 shows the die photo of the fabricated chip. By leveraging a cell-based design and APR, this is the smallest reported high-resolution DAC in this technology. The power consumption of the DAC is only 5mW from a 1V power supply, at 250MHz, excluding output current; 67% of this power is consumed in the memory blocks. Table 4-3 summarizes the specifications of this DAC, along with several recent DAC designs.

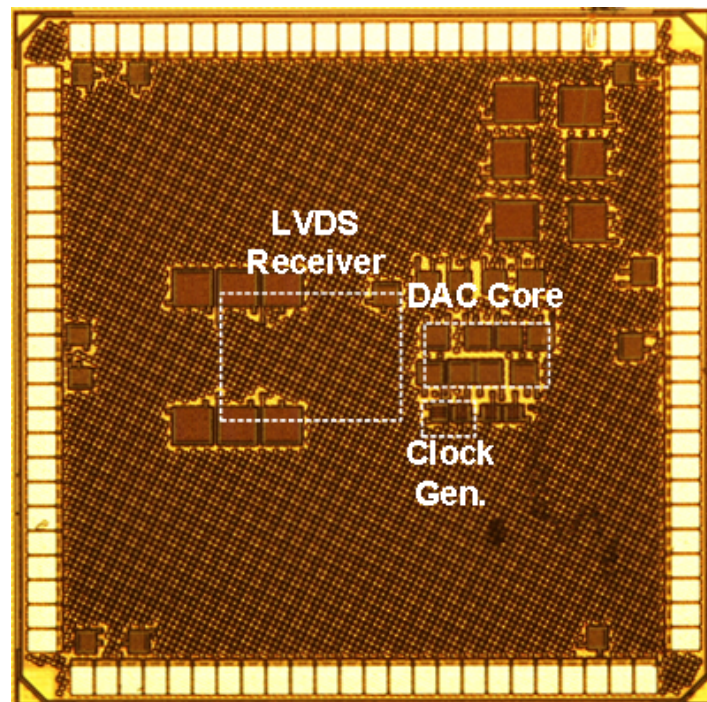


Fig. 4-9 Chip die photo – Fabricated in 65nm CMOS.

Table 4-3 Comparison of the proposed DAC with previous designs

<i>Specifications</i>	<i>This Work</i>	<i>[57]</i>	<i>[58]</i>	<i>[59]</i>	<i>[60]</i>	<i>[61]</i>
Resolution (N)	12	12	12	10	12	14
F_s (GS/s)	0.25	2.9	1.6	0.3	0.5	0.2
Max F_{in} (MHz)	120	300	800	150	240	95
R_L (Ω)	50	50	50	50	50	12.5
V_{p-p} (V)	1.3	2.5	0.8	6	1.5	0.5
SFDR (dBc)	>46	>66	>70.3	>44	>61	>78
Power (mW)	5.5	188	40	476	216	270
Area (mm²)	0.035	0.315	0.016	2.25	1.13	2.4
Process (nm)	65	65	40	45	180	140

Despite being the only DAC using a fully automated physical design flow, the linearity and SFDR place it on the lower end, but competitive with other state of the art DACs with full-

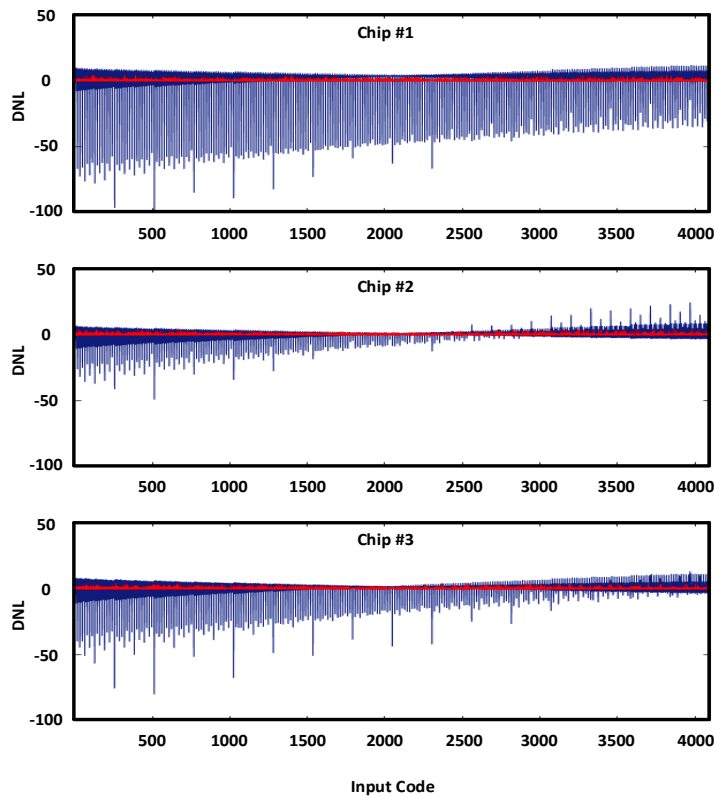


Fig. 4-10 DNL plots for 3 different chips, before and after calibration (before calibration: blue plots, after calibration: red plots)

custom design. The primary advantage being the design cycle time is significantly reduced by EDA tools. Secondly, the power and area of the DAC, even when scaled by sample rate and process node, is the best reported among these high-performance DACs. We attribute this entirely to the power digital EDA tools used to optimize the layout, compacting its size. Smaller area results in lower total switching capacitance, which significantly reduces the power.

Finally, to gauge the performance the chips under variation, we compared the before-calibration DNL results measured from another three different chips (see Fig. 4-10). The

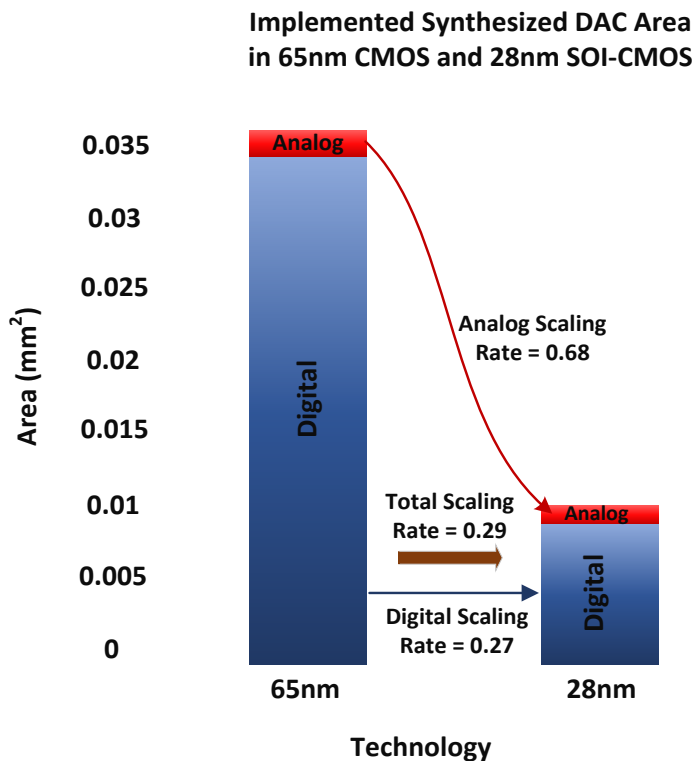


Fig. 4-11 Synthesized-DAC analog and digital area-scaling rate from 65nm CMOS to 28nm SOI-CMOS

significant variation that exists between the three chips is brought within the acceptable range ($|DNL| < 2.5$ LSBs) using our calibration techniques.

In addition to 65nm CMOS, we implemented the same DAC in 28nm SOI technology in order to investigate the scalability of the VLSA DAC in more advanced technologies. The digital and analog scaling rate is different from one technology node to another; analog designs are usually lagging in adapting the advanced node. As shown in Fig. 4-11, the area scaling rate for the digital portion of our synthesized DAC is 0.27, whereas the analog portion has scaled at a 0.68 rate. In our VLSA approach, the majority of the design is expressed as digital, therefore the overall scale rate will not be dominated by the analog scaling rate. In this example, the overall scaling rate is 0.29, which is slightly larger than that of the digital portion, but significantly lower than the analog scaling rate. It is worth noting that the theoretical density scaling rate from 65nm to 28nm is around 0.2 [63]. As seen in Fig. 4-11, the digital scaling rate is much closer to the theoretical rate.

4.6. Conclusions

This chapter presented the design of a fully synthesized low-power, low area, cell based DAC. The architecture of the proposed DAC is amenable to automatically place and route (APR) design flow. By applying calibration, the nonlinearities introduced by the APR tools is compensated. The calibration techniques are enabled by modifying the current steering architecture and employing tri-stated spare current cells, programmable LUT blocks, programmable gain adjustment units, and a programmable decoder for digital inputs.

With minimal design time and effort, this DAC achieves a performance comparable to conventional DACs. We predict that with the rapid growth in the number of design rules, a

cell-based design approach becomes a necessity, as it is much faster than full-custom design approaches. Semiconductor scaling only makes the problem worse, and it is only a matter of time before complex analog designs are forced in this direction; a path adopted decades ago by digital designers. Our cell-based approach also allows porting the design into other processes with negligible effort, as all of the code and scripts are reused with only minor adjustment.

Chapter 5

Baseband DSP for Ultra Wideband (UWB) Transceiver (TRX)

WSNs applications vary widely in different fields. However, it is apparent that the hardware design plays an important role for specific purposes. Compact hardware size along with long lifetime is generally desirable, nonetheless, quite challenging. Fig. 5-1 shows the block diagram of a typical WSN node. An integrated WSN node generally has several sensors, a digital signal processor (DSP) and controller, voltage regulation for power management, a radio frequency (RF) front-end and antennas for wireless communication, a battery as the energy source, and a crystal for frequency reference. Among these building blocks, the crystal reference and power source have been the most difficult to integrate into silicon [64], thus hindering the miniaturization towards cubic-mm scale WSNs.

A frequency reference provides a stable timing reference over process, voltage and temperature (PVT) variations for RF and clock synchronization of a communication system. The required timing accuracy depends on the system specifications [65] and can be achieved in different ways. A quartz crystal is the most common source of frequency reference. It provides excellent stability with PVT variations. However, their volume does not scale down with process or frequency, and they require a piezoelectric process, which is incompatible with monolithic integration. In order to get stable oscillation out of crystals, a certain amount of driving power is still necessary [66]. Therefore, the bulky size and cost of system integration become one of the bottlenecks for implementing crystals in mm-scale WSN nodes.

Micro-batteries are commercially available today with volumes approaching 0.2mm^3 .

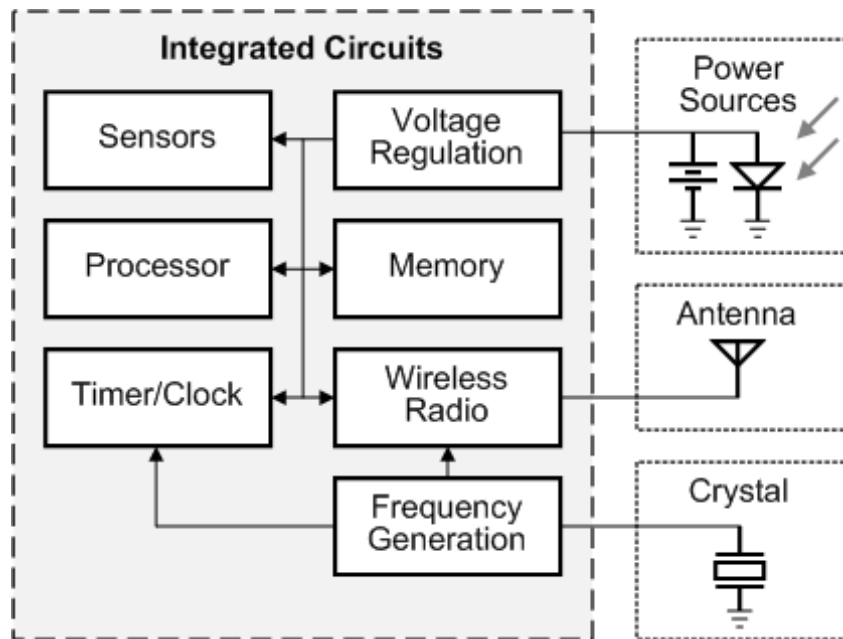


Fig. 5-1 Block diagram of a typical WSN node

However, they have limited capacity because of the small volume. Furthermore, peak current and capacity directly trade off in solid-state batteries, and because capacity is typically maximized, the peak current of micro-batteries is small (i.e. high output resistance) [67]-[71]. For example, in 1.38 x 0.85 x 0.15mm custom lithium-ion (Li-ion) battery from Cymbet Corporation, the capacity is 1 μ Ah and the maximum measured discharge current is 10 μ A [67]. The average power consumption must be <1nW for a one year node lifetime—therefore, leakage current is critical. These limitations present a direct challenge to the radio circuits, which typically consume >100 μ W when active. In order to function under this constraint, the node must be duty-cycled heavily; it must harvest energy from other sources; or the battery capacity must improve significantly. From a circuit design point-of-view, energy usage must also be reduced by clever circuit techniques.

In order to realize a fully-integrated wireless node at the mm-scale that operates off a micro-battery, this Chapter presents a 9.8GHz impulse-radio ultra-wideband (IR-UWB) radio in a 0.18 μ m BiCMOS technology. The use of a SiGe process provides higher breakdown voltages, higher transconductances, and higher current on/off ratios when compared to standard CMOS processes in general. These criteria are crucial for a radio design with a high supply voltage and low sleep power. This radio includes current-limiting at the battery supply to prevent it from exceeding the peak current of the battery, which would degrade capacity and lifespan. The charge coming from the battery is stored on a local storage capacitor, so that it can be discharged at higher currents for a short amount of time when the RF front-end is enabled and recharged between bits. IR-UWB communication is chosen because the pulse-based modulation scheme naturally provides

the smallest duty-cycling ratio [72]. The integrated modem of the radio duty-cycles the RF front-end at the bit-level, and is controlled through an I²C controller. The crystal reference is replaced with a temperature-compensated relaxation oscillator. The RF operation frequency is at 9.8GHz considering the tradeoff between the circuit power consumption and antenna size. Finally, this radio is designed to operate the RF blocks over the entire battery voltage range of 3.2~4.1V [67].

5.1. UWB Radio Architecture

The architecture for the IR-UWB radio is shown in Fig. 5-2. The receiver (RX) and

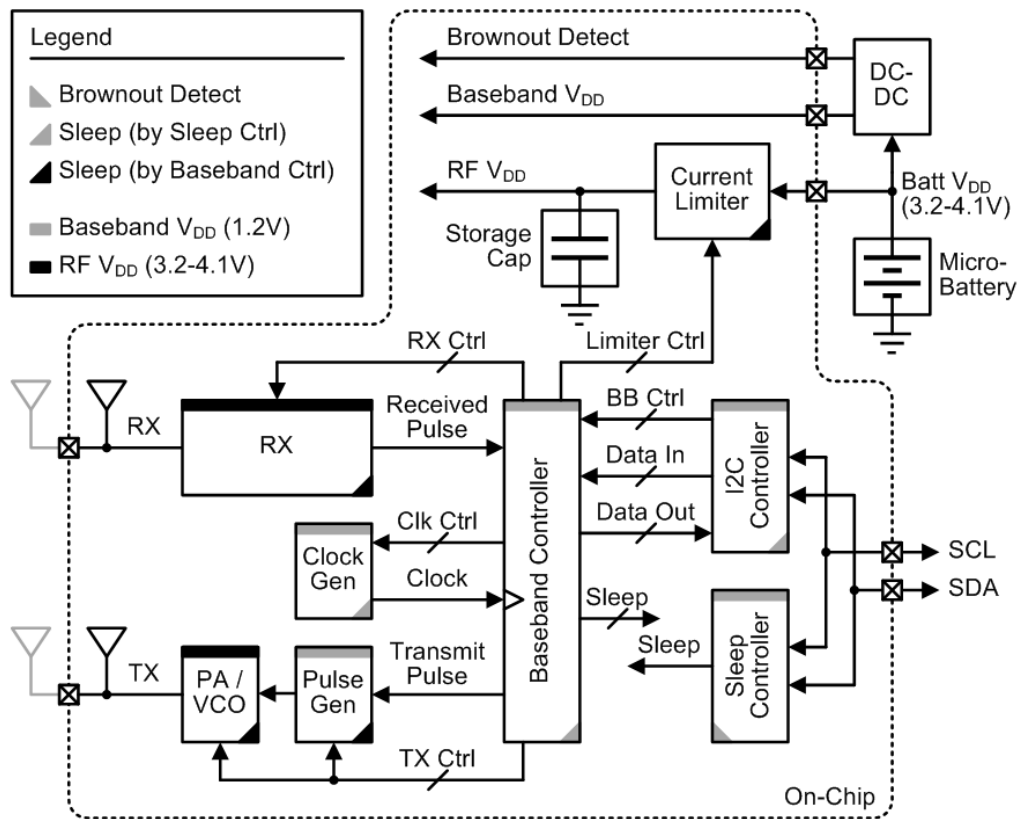


Fig. 5-2 System block diagram of the entire crystal-less UWB radio

transmitter (TX) operate at the battery voltage (3.2-4.1V), through a current limiter (CL) to protect the micro-battery from over-current and under-voltage conditions. An internal 3nF storage capacitor made of MIM layers allows higher current draws from the TX and RX during duty-cycled operation. Digital baseband blocks operate from a 1.2V VDD to reduce power consumption, regulated by a power management IC stacked above the radio (a separate die).

The baseband controller consists of a finite state machine (FSM) and memory for transmitting and receiving data. To survive on the limited resources of the micro-battery, all blocks on the radio have a low-power sleep state. RF and other analog blocks are duty-cycled at the bit level by the baseband controller, while baseband blocks are duty-cycled at the packet level by a separate sleep controller. The sleep controller remains on continuously unless an under-voltage condition occurs. The sleep controller begins and ends the wake-up procedure for each packet via I²C communication. The I²C controller has modified I/Os with keeper latches to eliminate pull-up resistors, and provides bidirectional communication with other stacked die in a mm-scale sensor node. Our main contributions are the baseband controller and the modulation/demodulation scheme, which will be explained more in the next section.

5.2. Baseband Controller

As previously mentioned, the RX and TX must be duty-cycled between incoming pulses and operate on a total average current of <100μA from the battery, which requires their active windows to be synchronized with each other over the channel. The baseband processor (BBP) is in charge of synchronizing the RX and the TX, as well as modulating and

demodulating the signals, tuning all the other RF/analog blocks, and communicating with the higher layers of the WSN node. Fig. 5-3 shows a high-level state diagram of the transmission and reception processes.

The transmission and reception processes begin with a signal from the I²C controller to wake up the BBP and start communication. In the case of transmission, a preamble is first sent, which is a sequence of pulses that the receiver will synchronize to. After the preamble, a header flag is transmitted to indicate the start of the payload, after which data transmission begins immediately, or the BBP goes through an optional handshaking with the RX before doing so. After the data payload is transmitted, another flag is sent to indicate

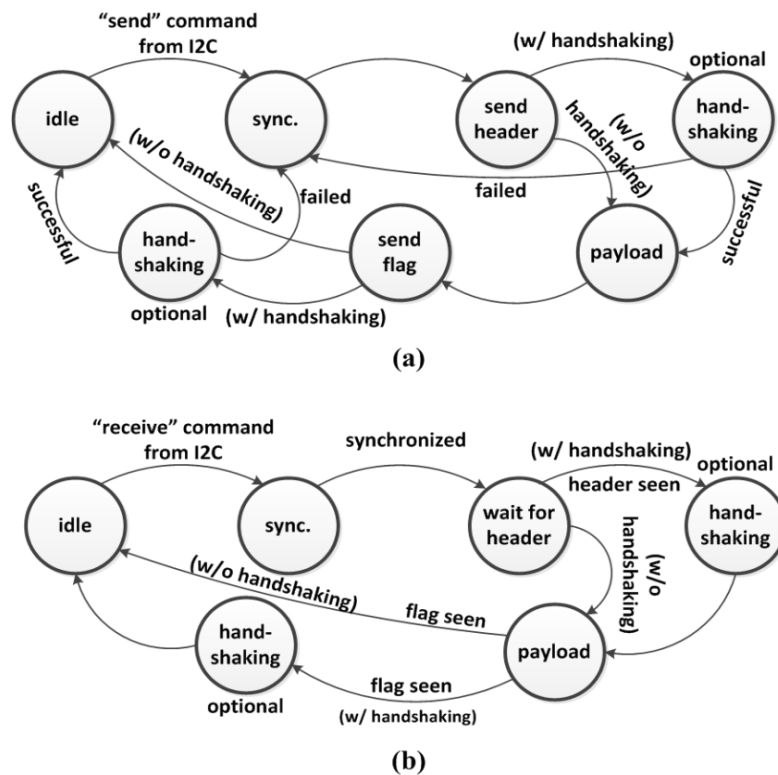


Fig. 5-3 High level state diagram for (a) transmission, (b) reception

the end of the packet, and another optional handshaking is done before returning to idle.

In the case of reception, the BBP first enters an acquisition mode where an active window is moved each bit cycle until continuous pulses from the TX are seen, after which, the RX locks its active window to that of the TX (coarse tracking). The receiver goes through the same steps as the transmitter, while keeping its active window locked on that of the TX throughout the whole transmission process, assuming a maximum of 1% clock drift.

The preamble is a long sequence of 1's. The RX has a searching window that is shifted until this window overlaps the synchronization pulses. Since the data is known to be a sequence

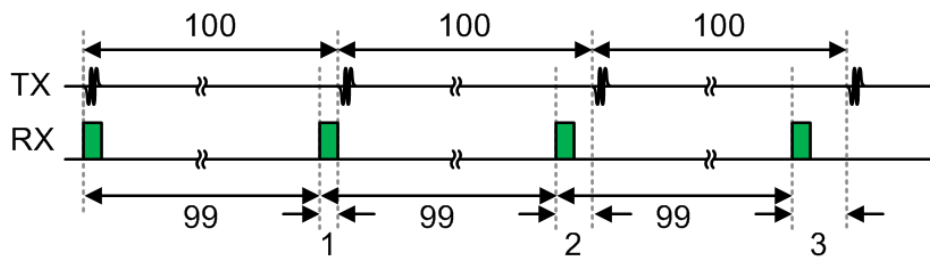


Fig. 5-4 Bit-level active window of the RX when its clock is 1% faster than the TX clock; the transmitted pulses are not being tracked by the RX.

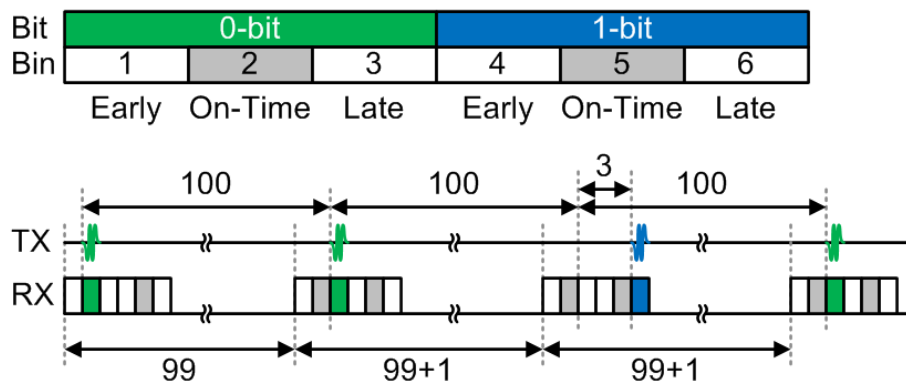


Fig. 5-5 Bit-level active window of the RX when its clock is 1% faster than the TX clock. The transmitted PPM pulses are being tracked by the receiver to maintain synchronization and a constant duty-cycling ratio of 6%

of 1s, the RX locks its window on the position corresponding to a bit 1. After initial synchronization, the clock drift can cause the RX to lose lock. Fig. 5-4 shows an example of how the receiver can lose synchronization with the TX due to a 1% clock drift. Thus, the RX must track phase to keep its window locked on the transmitted pulses throughout the entire transmission process.

The size of the active window of the RX is chosen based on the duty-cycle and the maximum difference between the TX/RX clocks. For a 1% duty-cycle and a maximum of 1% clock drift, a 6-clock-cycle window is sufficient to keep the RX locked throughout the process. This window has designated positions for the 0 and 1 data pulses. Once initially locked, the RX expects to receive a pulse corresponding to a 0bit during the second clock cycle of the window, and a pulse corresponding to a 1bit during the fifth cycle. These positions are highlighted (as grey) in the RX windows shown in Fig. 5-2. The 1% clock drift can cause the TX pulses to arrive one clock cycle earlier or later than their designated

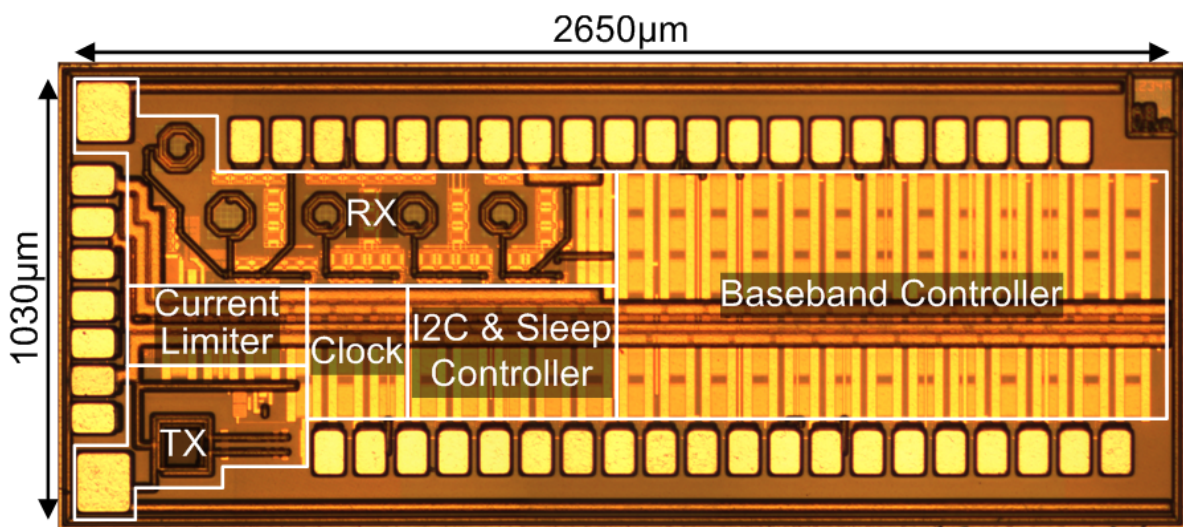


Fig. 5-6 Die photo of the radio

positions, in which case the RX adjusts its reception window accordingly. For instance, if a pulse arrives at the third cycle as shown in Fig. 5-5, the RX assumes that the data is a 0bit and concludes that it is one clock cycle ahead of the TX. The BBP then compensates for the difference in the next bit cycle. This process can continue throughout the whole data transmission.

5.3. Measurement Results

The radio was fabricated in $0.18\mu\text{m}$ BiCMOS technology with MIM capacitors. The RX and TX front-ends operate at the battery voltage of 3.2-4.1V. Digital processing and scan blocks operate at 1.2V to reduce dynamic power consumption.

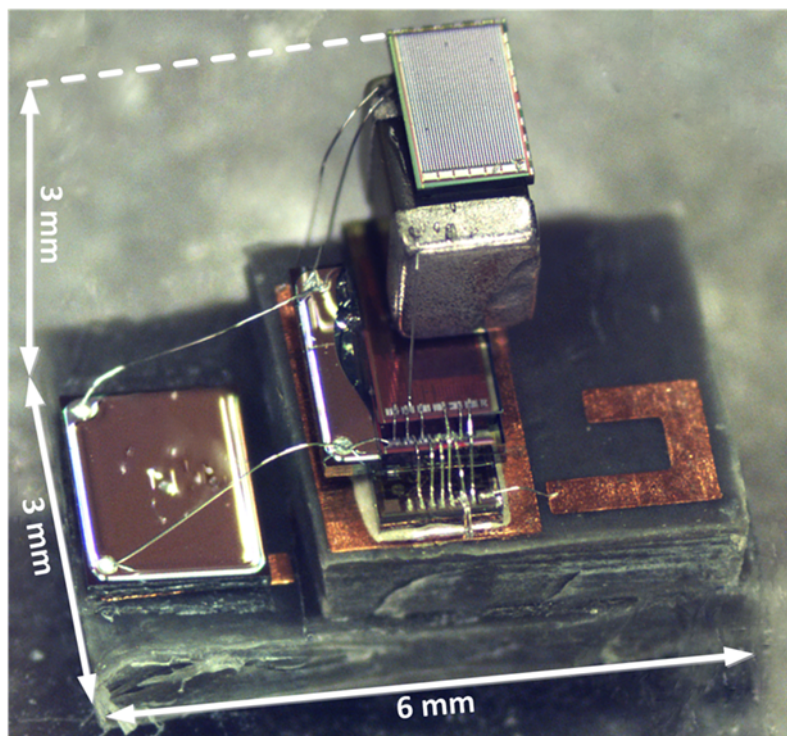


Fig. 5-7 Photo of the cubic-mm stacked WSN system

Each block in the radio consumes $<1nW$ while asleep by carefully including thick-oxide headers/footers on all blocks, making this system ideal for heavily duty-cycled cubic-mm sensor nodes. A performance summary is provided in Table 5-1. The die occupies approximately $2.73mm^2$, dominated by the baseband processor (Fig. 5-6). The entire radio is designed to operate from just the 7 pads on the left edge to enable die stacking; the remaining pads are for debugging and have internal pull-down resistors so they may be left open. We have compared the proposed radio with recently published UWB radios in 5.II. The proposed radio is the only one to operate at the battery voltage range and includes an entire baseband controller. In order to demonstrate that the radio can be integrated in a cubic-mm scale WSN node, a heterogeneous die-stacking system is implemented (Fig. 5-7).

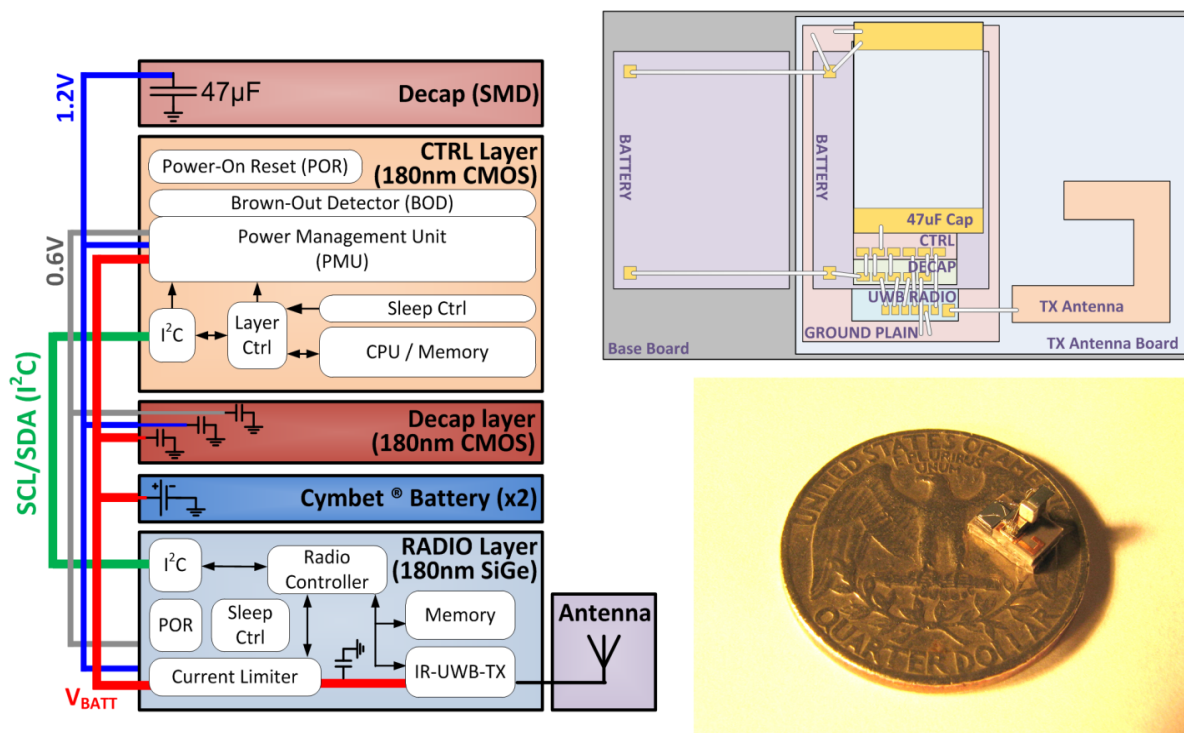


Fig. 5-8 Top level block diagram of the proposed wireless sensor node (left), wire-bond diagram, and size comparison of the autonomous node

Fig. 5-8 shows the high level block diagram and wire-bond structure of the stacked node. It contains the crystal-less IR-UWB radio, two Li-ion micro batteries (EnerChip™ CBC005), digital system control CPU, inductor-less power management unit, decoupling capacitor to supply large peak currents to the radio, and an on-board antenna.

By folding the monopole on-board antenna, it can be minimized to a total electric length of $0.08\lambda_0$. The off-chip antenna occupies an area of 1.95mm^2 on an RT/Duroid 5880 substrate. The standalone node with no external connections demonstrated node-to-base-station communication up to 2.5m, and sustained autonomous operation on the micro battery for 17 minutes, transmitting 255 packets during that time with no recharging

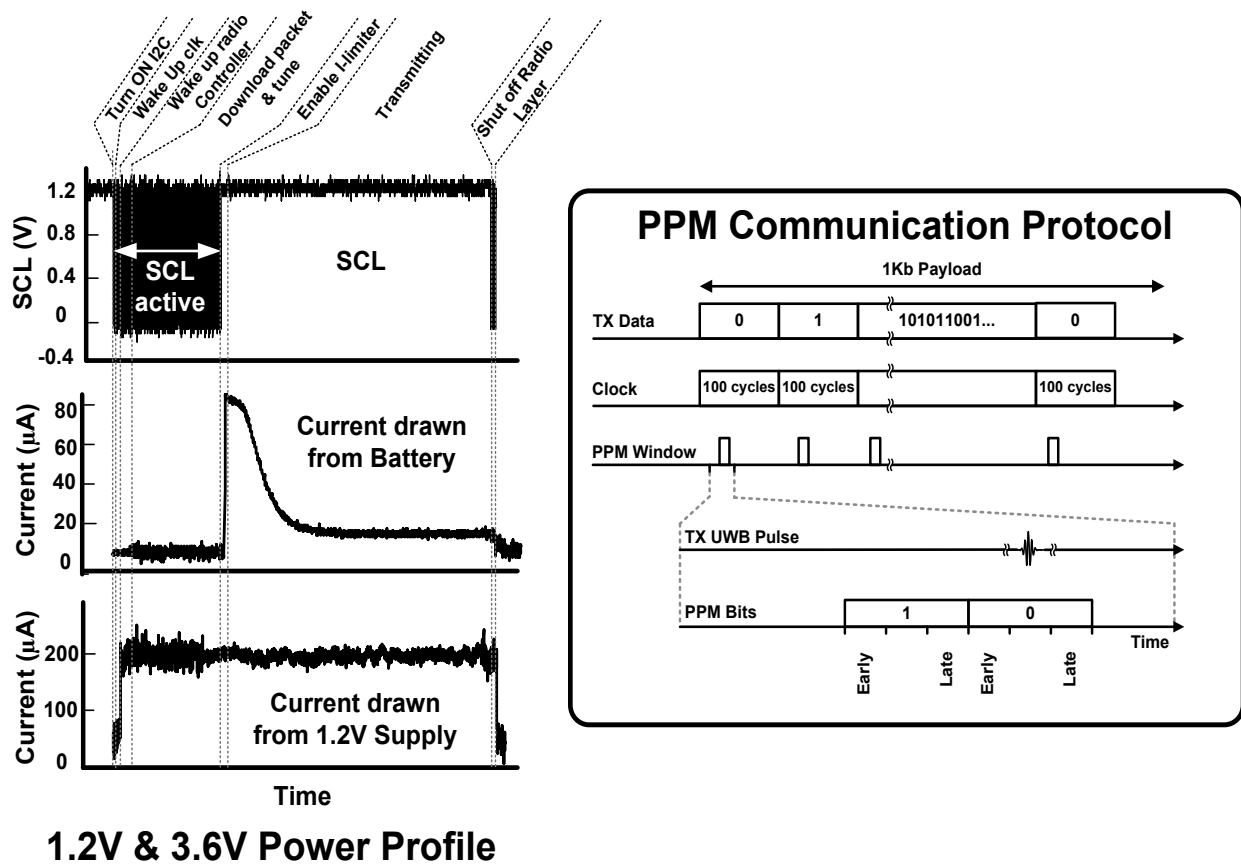
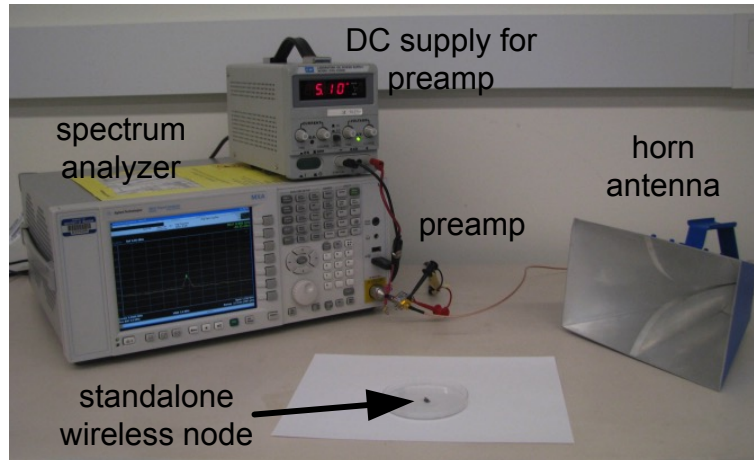


Fig. 5-9 Power consumption profile of the radio layer in a basic transmission cycle

between transmissions; Fig. 5-10 shows the stand-alone node test setup. Fig. 5-9 shows the measured power consumption profile of a basic transmission.

Measurement Setup



Node lifetime @ 30kbs	~17mins
Center frequency	8.9GHz
Horn antenna gain	10dBi @ 8.9GHz
Preamp gain	24dB @ 8.9GHz

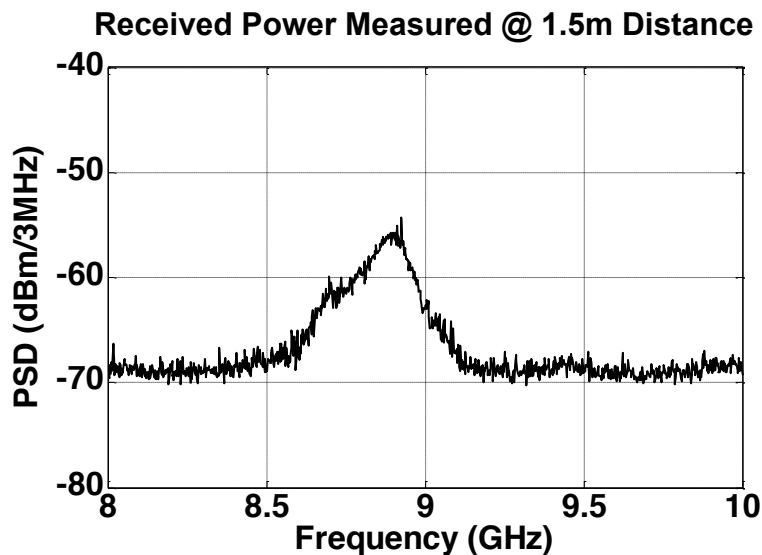


Fig. 5-10 Test setup, transmission characteristics, and received power at 1.5m. Node lifetime is measured while the sensor node transmits 1kb every 3 seconds at data rate of 30kpbs

Table 5-1 Summary of the Radio Performance

<i>Process</i>	0.18 μ m BiCMOS
<i>Modulation</i>	PPM
<i>Center Frequency</i>	9.8 GHz
<i>RF Voltage</i>	3.2-4.1V
<i>Baseband Voltage</i>	1.2V
<i>Clock Frequency</i>	3 MHz
<i>Active RF Power (Average)</i>	597 μ W @3.6V
<i>Active Baseband Power</i>	26 μ W @1.2V
<i>Sleep Power</i>	1.0nW @3.6V
	1.8nW @1.2V
<i>Data Rate</i>	30kb/s
<i>Total Area</i>	2.73mm ²

The entire radio layer is duty-cycled between transmission packets with an enable/disable sequence controlled by the CPU on the control layer. The startup sequence is initiated when the CPU wakes up the radio's I²C module, which handles all communications between the control layer and the radio layer. The CPU first activates the clock and initiates the radio controller, holding it in reset. After the clock stabilizes, the controller is released from reset; the CPU transmits configuration and packet data and sends the instruction to initiate transmission. The radio controller then enables the CL to charge the integrated storage capacitor on the RF supply and activates the TX. Immediately following transmission, the power down sequence begins. The CL is disabled which drops the current draw to 150pA

and deactivates the transmitter via power-gating. The radio controller and clock are put to sleep, and finally, the I²C module is deactivated except for a sleep controller which monitors the I²C lines for the next wakeup.

5.4. Conclusions

This chapter presented a fully integrated DSP for IR-UWB radio in 0.18 μ m BiCMOS technology. The DSP operates within the limits of mm-scale micro-battery miniaturized WSN nodes. We demonstrated the first full-system standalone mm-scale sensor node in a heterogeneous die-stacking system. The standalone node (with no external connections) performs a node-to-base-station 2.5m communication, and sustains autonomous operation (with the micro battery) for 17 minutes, transmitting a total of 255 packets with no recharging between transmissions.

Chapter 6

Beyond CMOS

This chapter presents our design automation techniques for superconducting circuits. First, an overview of the current state of superconducting circuits is given. Then the design flow and its challenges are introduced. Finally, our automation techniques and the experimental results are provided.

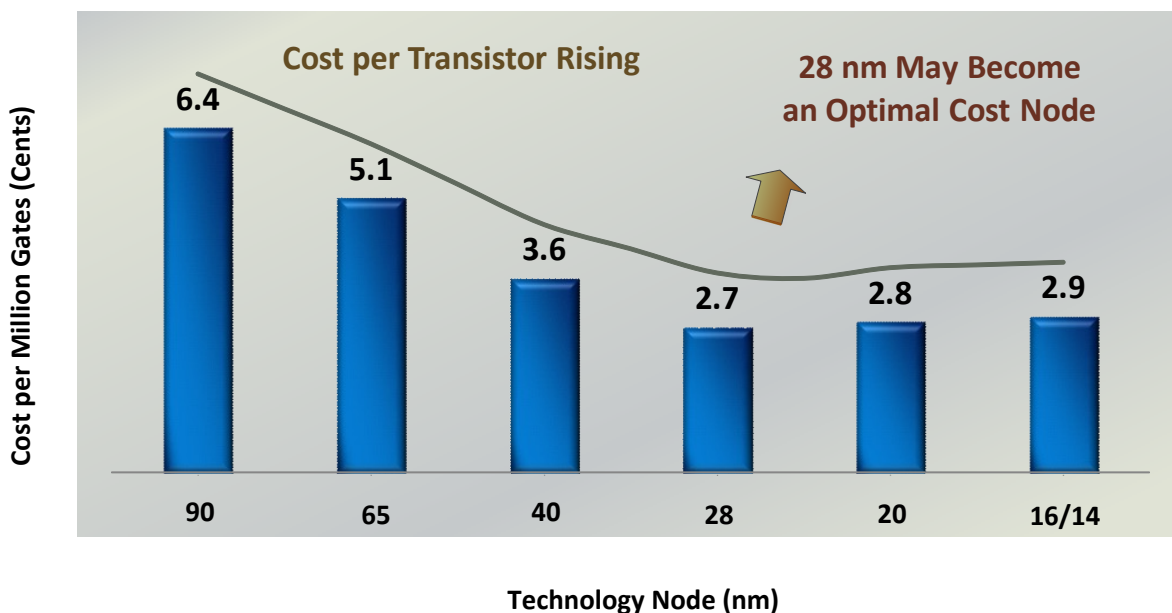
6.1. Motivation

As discussed in the earlier chapters, smaller technology nodes usually provide higher speed and increases the level of integration. Scaling CMOS devices has slowed down in the recent years and as a result, the cost per transistor of the circuits has increased (Fig. 6-1), along with the cost of operations [74]. In addition, scaling increases wire resistances, leading to higher delays, as well as heating problems [83].

One of biggest challenges of the current high efficiency digital systems is their power consumption. The estimated power required for big data and internet-related systems [85] (i.e., servers and data centers) is about 12GW in the Unites States, which is equal to the output of 25 power plants [86]. The energy consumption of such systems is expected to

reach ~15% of the total energy consumption of the world soon. Thus, reducing their power consumption is a desirable goal that benefits humanity and helps in slowing down global warming [85].

The rapidly growing demands for wireless communications, computing units, and data centers, and etc. exceed the limitations of the existing technology nodes and systems [88]. Consequently, significant changes must be applied to the way the circuits are designed (both at the device level as well as the system level). There are three main approaches to address the limitations: (i) creating new devices, (ii) building new architectures, and (iii) developing new computational paradigms (see Fig. 6-2). New architectures and computational paradigms (e.g., neuromorphic computing, dataflow computing, architectures for dark silicon, accelerator-rich architectures, etc.) are being developed to replace the old ones that are no longer efficient and cannot keep up with the new



**Fig. 6-1 Cost per transistor for deep nanometer CMOS technology nodes [74]
(copy of Fig. 1-9)**

technology demands. Similarly, new devices and materials are being evaluated as a replacement for analog and digital CMOS devices [88]. For instance, superconducting circuits, which are the main topic of this chapter, have become attractive because of their extreme power efficiency.

A short-term solution to address the demands is to fabricate CMOS-based devices that grow in the third (vertical) dimension. The 3D fabrication development alongside with the new packaging and architectures enhances the performances, but they are only temporary solutions. As discussed, new classes of devices and novel computing systems are required to meet the exponential growing needs for communications and IT systems today and in the future. These systems must also be scalable and should be economically manufactured [88].

6.2. Introduction to superconducting circuits

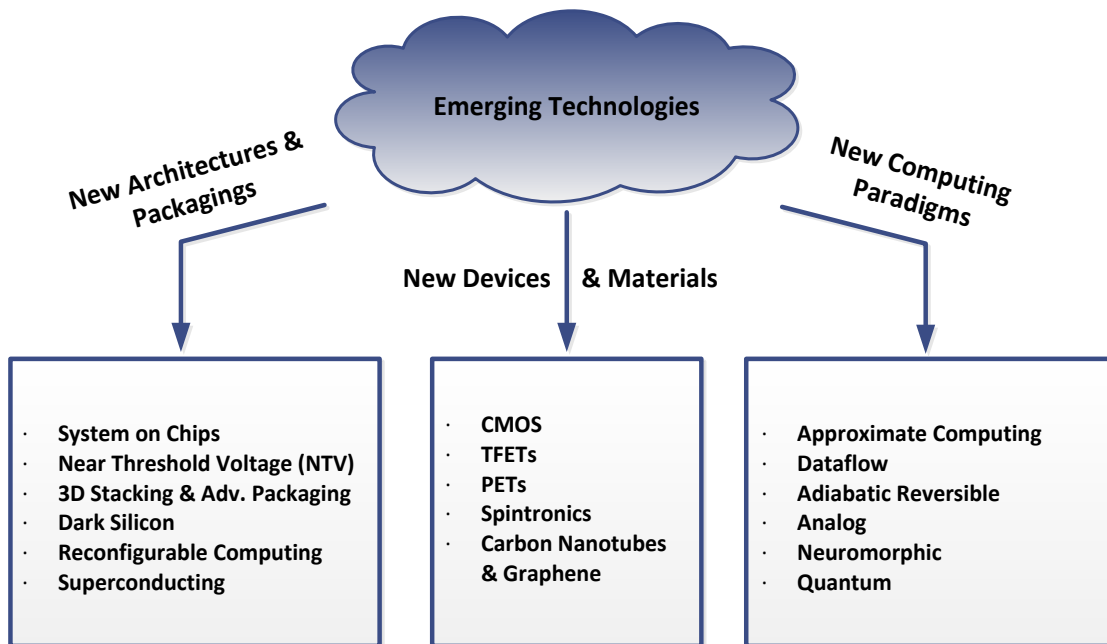


Fig. 6-2 Emerging technologies [88] (copy of Fig. 1-10)

Superconducting circuits are among the new technologies that are being evaluated for future computing systems, mainly due to their high speed and low power characteristics. In this section we provide a brief introduction to superconductor devices and integrated circuits.

Superconductors are materials (inter-metallic alloys or compounds) that conduct electricity with zero resistance, once they are below a certain temperature (T_c). Since they effectively become a perfect conductor below T_c , electrical current flows through superconductors without any energy loss. This makes them very suitable for low-power processing circuits [75]-[76].

In conventional CMOS circuits, data is encoded into different voltage levels in the transistors. Due to the charging and discharging of the loads (i.e., the interconnect and gate capacitors) and also the leakage of devices, energy is lost (dissipates as heat). With the exponential growth in the number of transistors on chip, the power consumption of large systems grows exponentially as well. This limits the clock rate of the billion-transistor systems to only a few gigahertz. It also limits the portion of the chip that can operate safely without violating the thermal limit; a phenomenon known as “dark silicon” [85].

Superconducting circuits with Josephson junctions (JJ) (active superconductor components) have been recently re-visited as an energy efficient alternative to CMOS for high performance computing systems, mainly because of their potential to run at extremely high clock rates and very low energy dissipation. Recent studies have focused on the possibility of building large computing systems that employ JJ-based superconductor devices [85].

Unlike CMOS, where the power consumption scales with the size of the devices, the power of JJ devices depends on the Fermi level of the materials and it does not scale. The highest reported bandwidth of the superconducting circuits is as high as 770GHz [86]. This limit is caused by the fact that the devices lose their superconductivity when operating beyond certain frequency (critical frequency). In contrast, interconnects can significantly reduce the performance of CMOS systems, especially when the system is dense. The power per operation of superconducting logic circuits is significantly lower than that of CMOS, and interconnects dissipate zero power.

Josephson junction devices employ single-flux quantum (SFQ) logic as the means of carrying information. This allows the devices to operate in the order of picoseconds (ps), with a power consumption that is three orders of magnitude lower than that of the advanced (deep nanoscale) CMOS logics. The amount of the magnetic field of a single quanta is $\hbar/2e = 2.06 \times 10^{-15}$ Wb, which is equivalent to 2mVps = 2mApH. This shows that with a persistent current in an inductive loop, we can get a 2mv pulse for the period of one picosecond. The pulse energy of the SFQ is around 1×10^{-19} J, which is only three orders of magnitude larger than the thermal Boltzmann limit ($K_B T$) [86]. It is important to note that superconducting circuits operate below a certain temperature, so the power used for their refrigeration must be included in the total power consumption of the system.

As mentioned, superconducting circuits have recently gained attention and different groups have started employing them in various applications. For example, in [81] the authors present a high-speed superconductor based D-Flip-Flop that operates up to 750GHz. The authors of [82] show that superconductor IC technology has the potential to merge the digital and RF domains for software defined radio applications. They also show

that by rapid single flux quantum (RSFQ) logic, in which a flux quantum is used as an information carrier, clock frequencies beyond 100GHz can be achieved. The authors of [83] discuss the development of SFQ logics for switches for high-end routers and microprocessors used in high-end computers. They also show that with the advancements in the design tools, large scale circuits with several thousand junctions can be designed. Such systems operate in the order of 10-100GHz. Mukhanov et al. [84] present several ADC architectures, including a superconductor Nyquist ADC (Flash ADC). This ADC utilizes SQUID comparators and has a low design complexity, while allowing fast sampling (20GHz).

Herr et al. [86] introduce a new logic family called reciprocal quantum logic (RQL). RQL maintains the good properties of CMOS logic (low static power consumption, efficient (fast) combinational logic, etc.) while using high speed and low power superconductor devices. Overall, RQL's power consumption (including the power required for refrigeration) is around 300 times lower than that of nano-scale CMOS systems. The remainder of this chapter focuses on RQL circuits.

An existing limitation of the superconductor technologies is their lack of scalability. In order to successfully implement contemporary computing tasks, RQL circuits reach the complexity of existing VLSI systems that are easily implemented in CMOS technologies [86]. Thus, despite recent advancements, we are yet to see VLSI-grade superconducting circuits. Next, we will discuss some the RQL technology features.

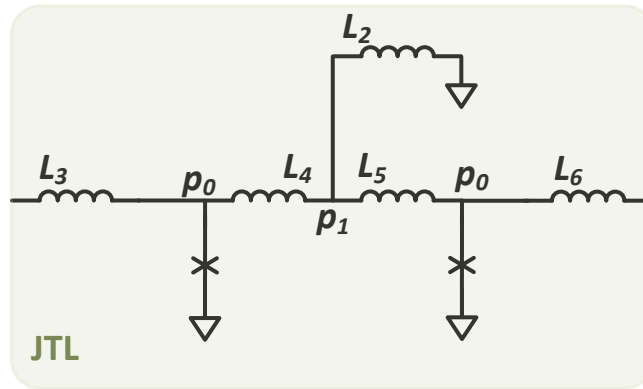


Fig. 6-3 Design of an active Josephson transmission line (JTL) [86]

A. Active Interconnect

Josephson transmission lines (JTLs) (shown in Fig. 6-3) are active interconnects in the RQL technology. They provide isolation for the gates they connect and they synchronize the signals with different clock phases. JTLs are powered by AC clock signals. An example design is shown in Fig. 6-4. This is a shift register design with JTLs that are driven by four-phase clock signals. Two clock lines (I and Q) provide four clock phases (0, 90, 180, and 270 degrees).

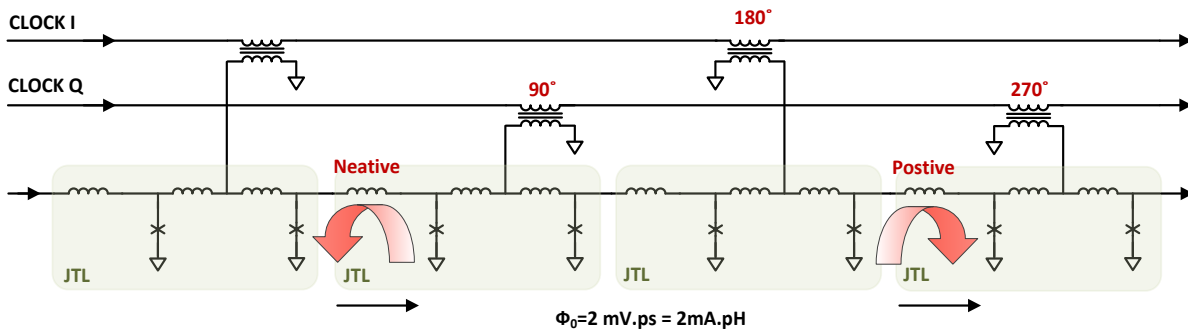


Fig. 6-4 A shift register design, with JTLs that are driven by four-phase clock signals [86]

degrees) for the designs [86]. In the RQL technology, data is represented by pulses or lack thereof; a positive SFQ pulse followed by a negative pulse (half a clock cycle later) represents a binary 1 and a nonappearance of any pulses represents a binary 0 (Fig. 6-5).

Fig. 6-6 (a-d) shows four propagation phases of a positive input pulse in a JTL. First, the two JJs in the JTL are biased through inductive coupling (a), then a positive input pulse arrives (b) and the current of the first JJ exceeds the critical current and its flux is changed to the opposite direction (c). As a result, the current of the second JJ also exceeds the critical current and results in a flux phase change and the propagation of the flux to the second loop in JTL. Fig. 6-7 (a-c) shows the propagation of the negative input pulse in a JTL. A negative input signal clears out the excessive flux in the JTL loops, and sets back the state of the JJ biases to the original state and prepares it for the next operation [86].

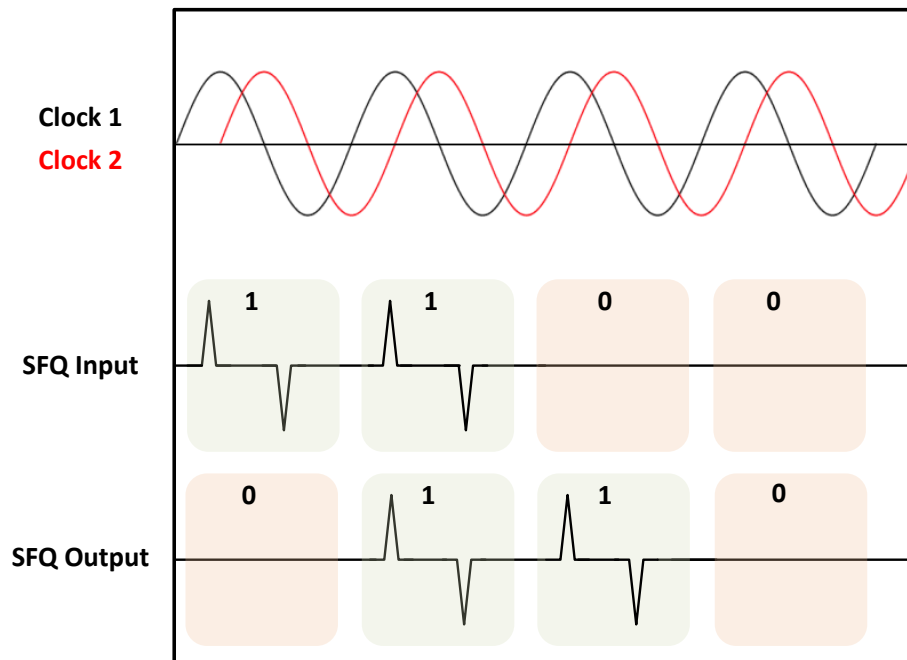


Fig. 6-5 Representation of the 1's and 0's in reciprocal quantum logic (RQL) [86]

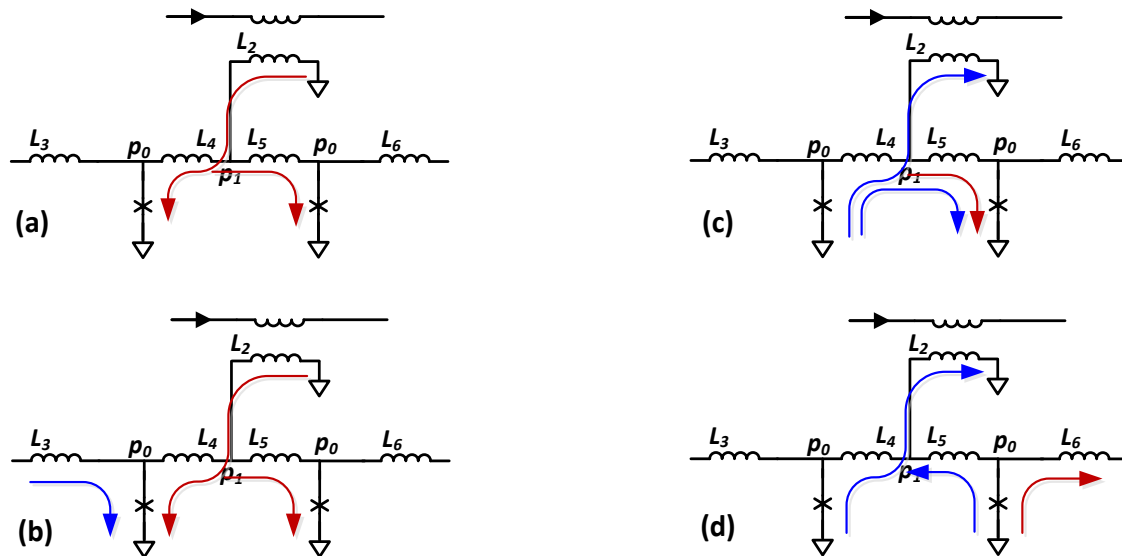


Fig. 6-6 Timing diagram for a positive input pulse in a JTL

B. Power and Timing

Driving all the JJ (Josephson Junction) devices simultaneously requires a significant amount of power, especially because the JJ devices have low impedances. Therefore, the devices are

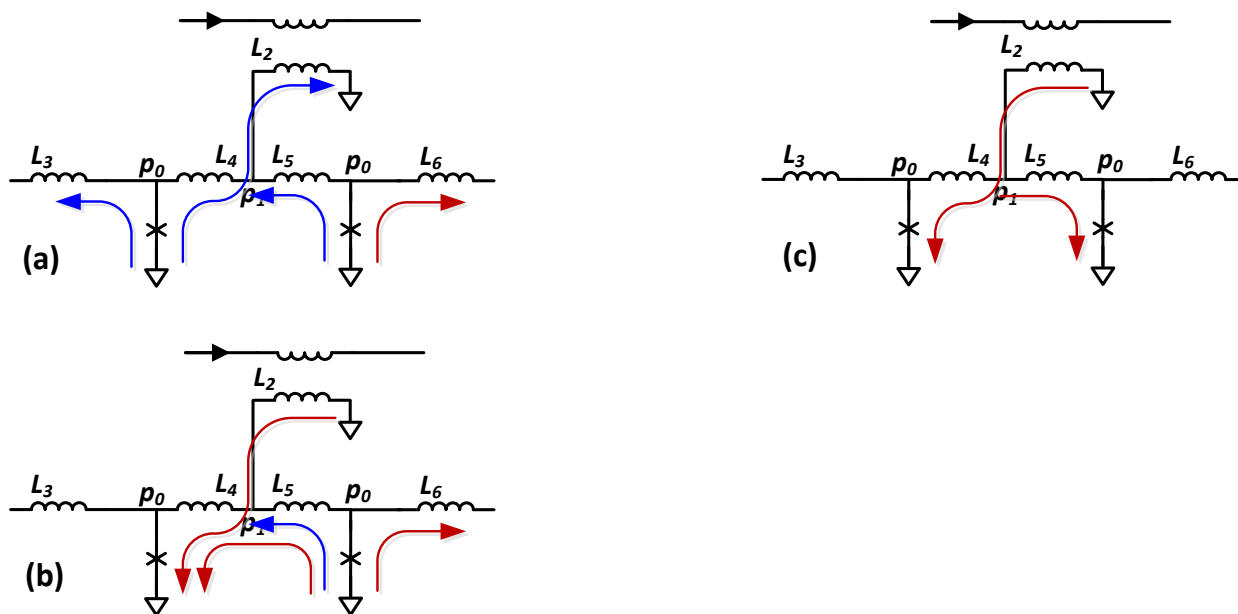


Fig. 6-7 Timing diagram for a negative input pulse in a JTL

powered in series by an inductively coupled AC power source. The static power of these devices is zero (because non switching JJ devices consume no power). Pulses are transmitted freely within a clock phase before it reaches the next phase. For timing purposes, a highly stable AC clock is utilized, which is essential because it makes all the signal timings stable [86].

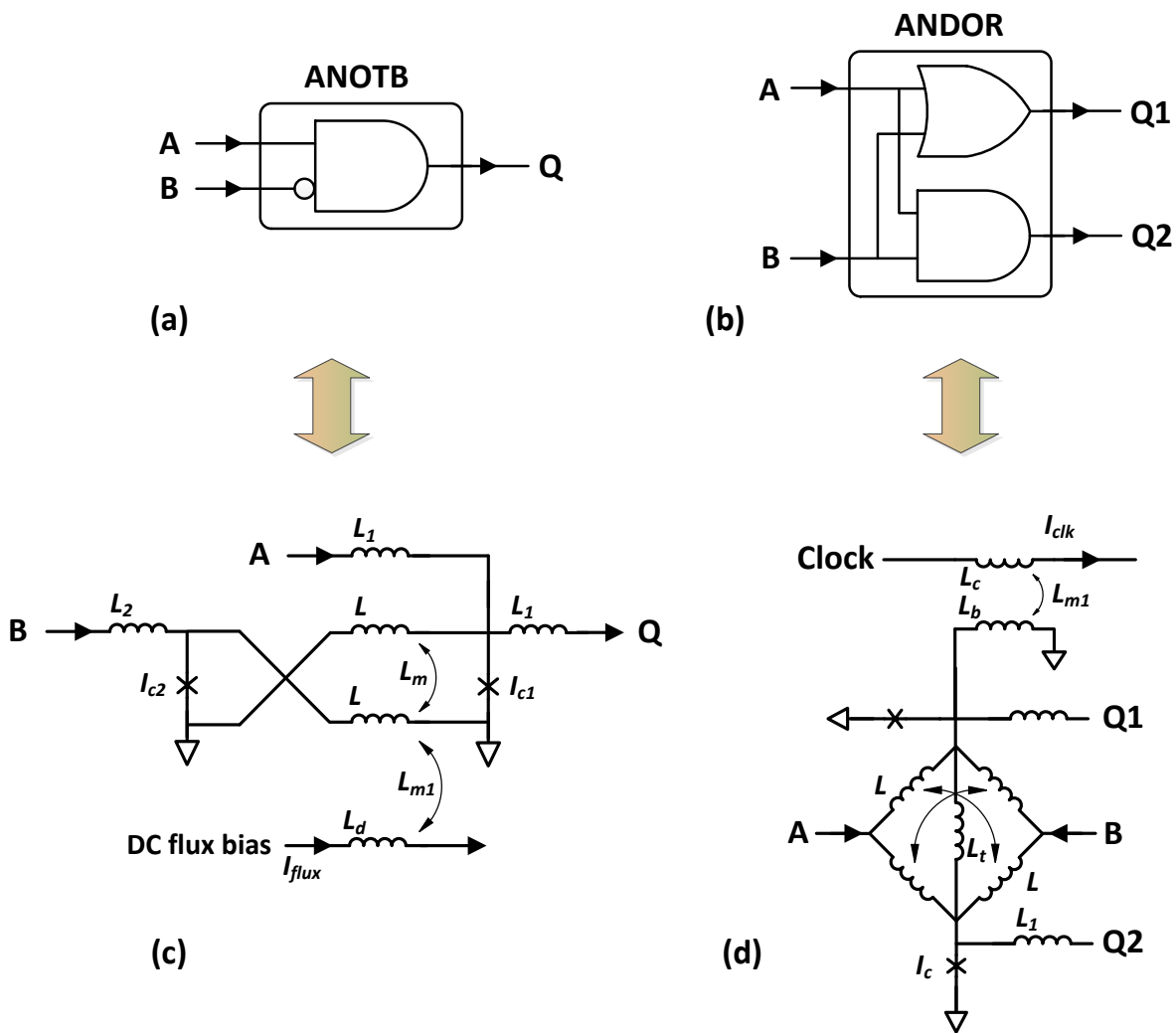


Fig. 6-8 RQL logic gate examples [86]

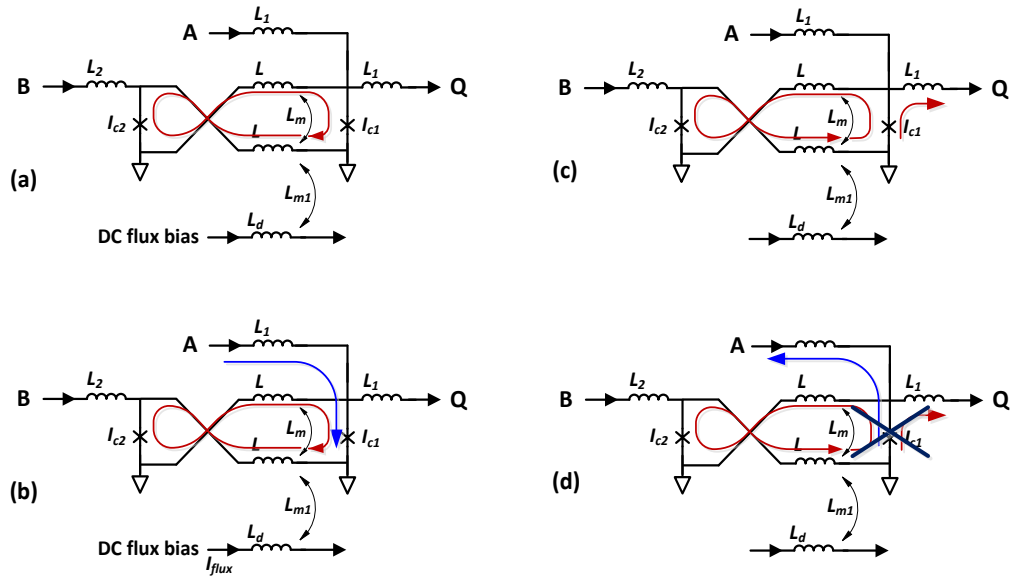


Fig. 6-9 Timing diagram for a positive followed by a negative input A pulse in the absence of input B

C. Logic Gates

Similar to CMOS, RQL circuits are made out of a set of pre-designed standard cells (or gates). Fig. 6-8 shows two basic standard gates that together form a universal set of gates,

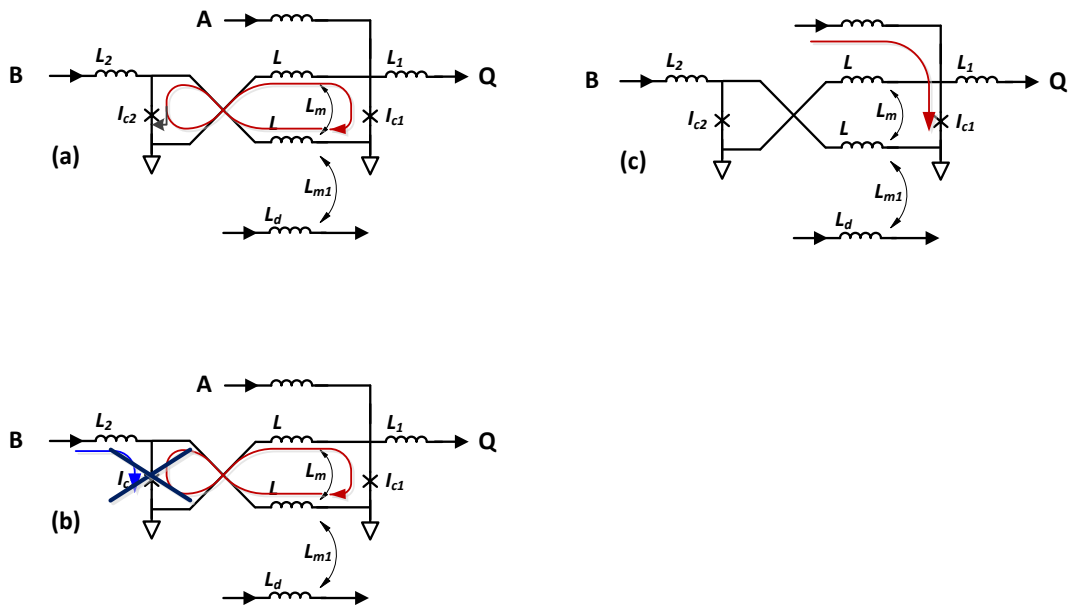


Fig. 6-10 Timing diagram for a positive input A pulse in after a positive input pulse at input B

meaning that they are capable implementing any combinational function. Fig. 6-8 (a) shows the ANOTB gate, in which the input A will pass to the output Q only if the input B is zero. Timing is important in this gate; the B pulse (if any) must arrive before the A pulse for proper gate operation. Fig. 6-9 (a-d) shows signal propagation in an ANOTB gate when a positive pulse followed by a negative pulse arrives at input A, and input B is absent. Fig. 6-10 also shows the state of the ANOTB gate when a positive pulse at B arrives before the positive pulse at A. In this scenario no input signal propagates to the output.

The gate shown in Fig. 6-8 (b) is the ANDOR gate, in which gate the first input pulse propagates to the OR output (Q1) and the second input pulse goes to the AND output (Q2). An RQL logic cell cannot drive another cells because the maximum fanout of each logic gate is less than one. Therefore, JTL interconnects are used to connect the gates together. In addition, one JTL is capable of driving up to two other JTLs, essentially acting as a 1-to-2 fanout module. Hence, a cascade of JTLs can be used to increase the fanout of a gate and route its output to multiple locations[86].

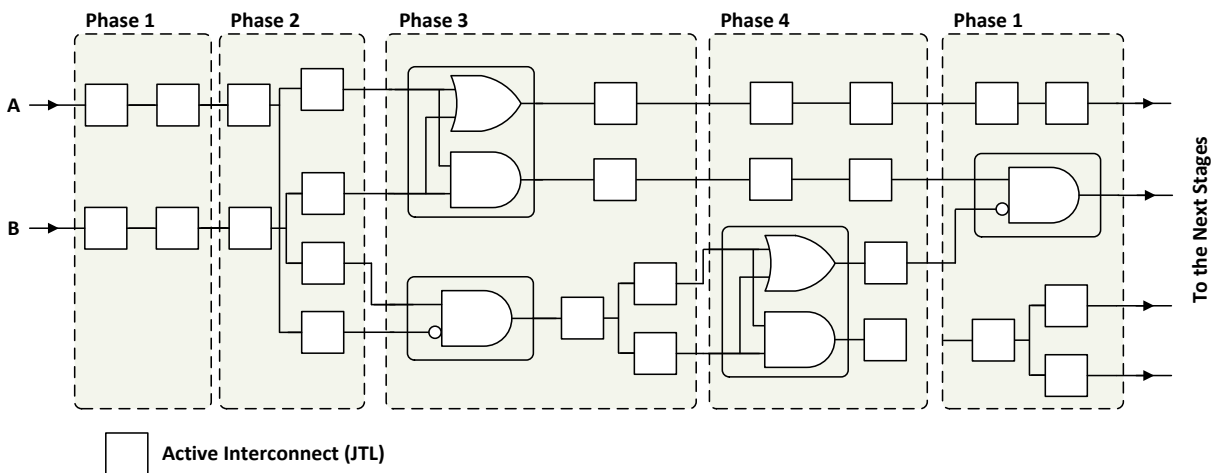


Fig. 6-11 An example of a synthesized and timed design in RQL technology

The driving strength of RQL gates is different from that of CMOS gates. Consequently, the design rules used for synthesis are different from CMOS (Fig. 6-11). For instance, RQL signals are synchronized with AC clocks (they are effectively wave-pipelined), whereas in CMOS, combinational circuits operate with no clock and only get synchronized through registers. In a JTL, a low-to-high transition occurs only during the positive half of the clock and a high-to-low transition occurs during the negative half of the clock (similar to Domino logic). Therefore, timings and synchronizations are important in the RQL technology [86].

6.3. Top-down design steps: RTL Verilog to Josephson junctions

Superconducting circuits operate at a high speed and consume low power. However, to compete with CMOS circuits, they must scale up to large integrated systems. Thanks to the recent advances in superconductor integrated circuit fabrication, scaling in superconducting circuits is now possible. The integration could further be improved by automating the design and layout process of the superconducting circuits. In this section we describe our contributions in automating the RQL design process.

The RQL Datapath Compiler (DPC) is a design tool developed at Microsoft Inc. [87] that allows designers to map Verilog register-transfer level (RTL) descriptions into different representations that are used in the design-flow steps that lead to a chip-level design. A flowchart of the DPC design process is shown in Fig. 6-12. This section describes the current design process, as well as our contributions to the flow and the DPC tool. These contributions affect the green blocks of the design flow shown in Fig. 6-12.

As mentioned earlier, by applying the cell-based concept and breaking the circuits into smaller blocks, we can generate a set of library cells (e.g., superconductor standard cells).

The superconductor standard cells have to be custom designed and optimized manually. However, higher-level designs can benefit from abstraction and re-use of the cells, allowing them to be expressed via behavioral or structural Verilog. Automatic placement of the cells, as well as their routing, is managed via DPC and existing CAD tools that are used in CMOS VLSI designs (e.g. Encounter).

A. Synthesis Netlist

The synthesis step of the RQL technology is similar to CMOS; the RTL Verilog description is translated into a gate level Verilog netlist. The only difference is the use of certain library sets that include functionality and timing information related to the RQL gates (standard

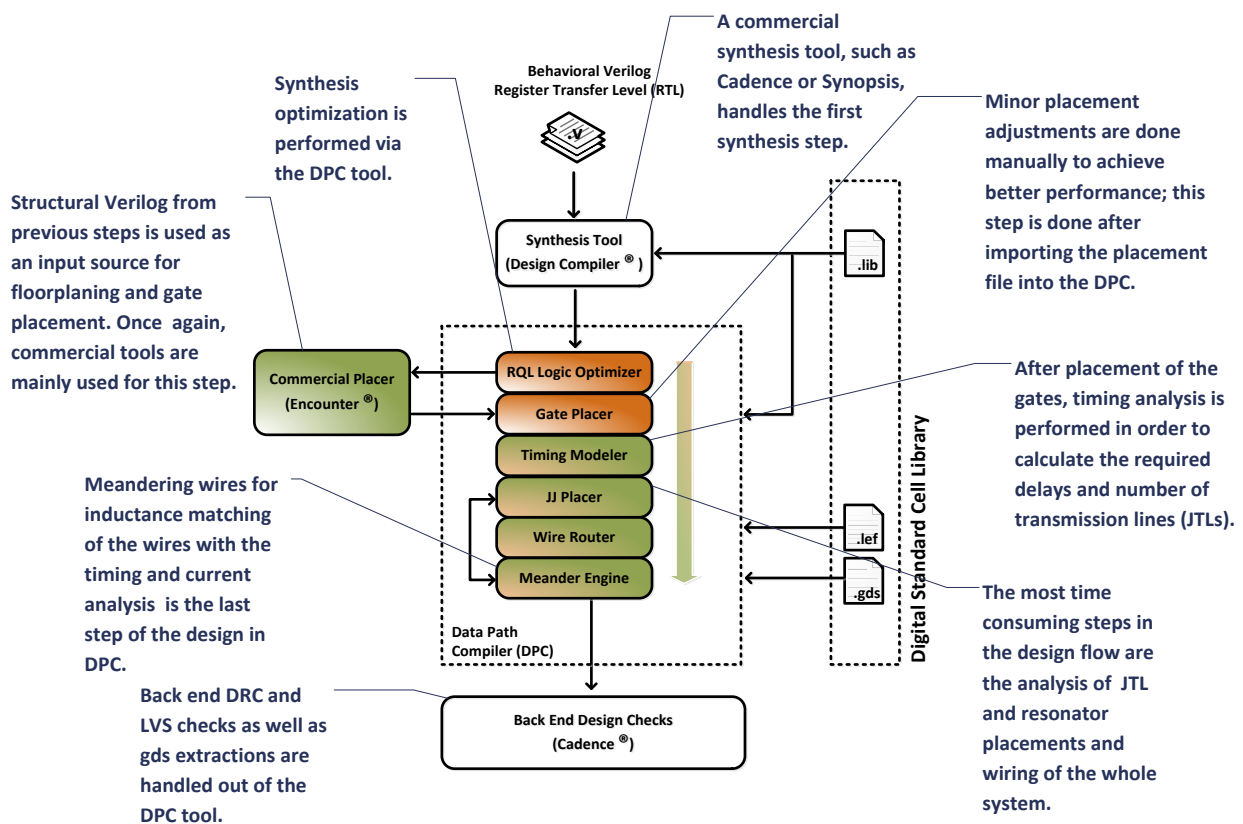


Fig. 6-12 Automated RQL design flow

cells). In the existing design flow, a combination of commercial tools and the DPC tool is used to generate netlists that are then used for placement and routing.

B. Gate Placement File

The DPC tool does not support automatic placement of the gates. While it is possible to use DPC for manual gate placement, it is not an efficient process, especially for designs with hundreds or thousands of gates. Therefore, commercial and industrial placement tools are currently used in the RQL design. These commercial tools, however, are optimized for CMOS technologies, and they need to be adjusted before being employed for superconducting circuits. The adjustments are performed through a set of scripts and constraint files that are fed into the tools. Once the placement is complete, the generated files are imported into DPC for minor placement alterations, timing calculations and routing.

C. Timing and closure

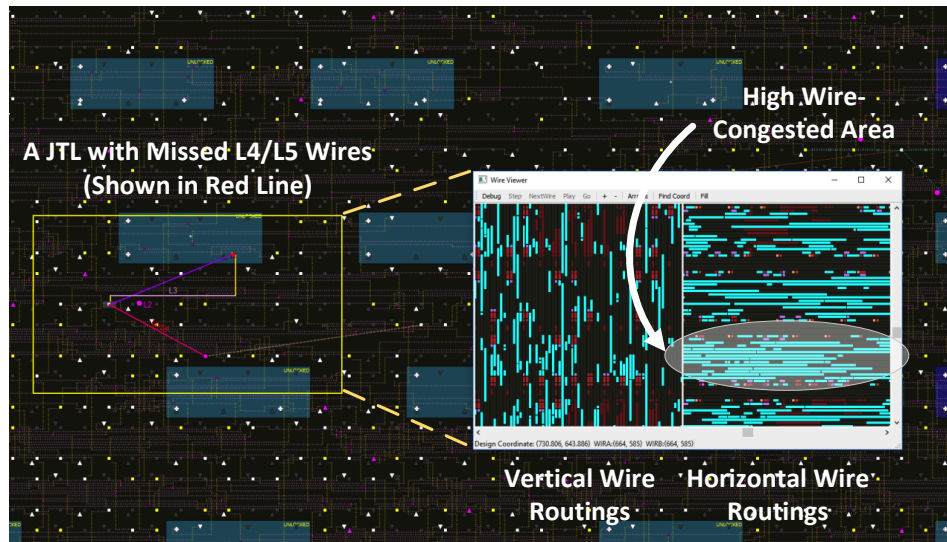
The immediate step after the placement is timing calculation. This step is conceptually similar to the clock tree calculation of the CMOS design flow. First the operating phase of each node is calculated and the number of JTLs required for each path is obtained.

In the final step, the design is finalized in DPC and is prepared for the fabrication. After timing calculations, JTL and resonators are placed, wires are routed, and the wire inductances (lengths) are adjusted (via meandering wires) using various DPC subtools.

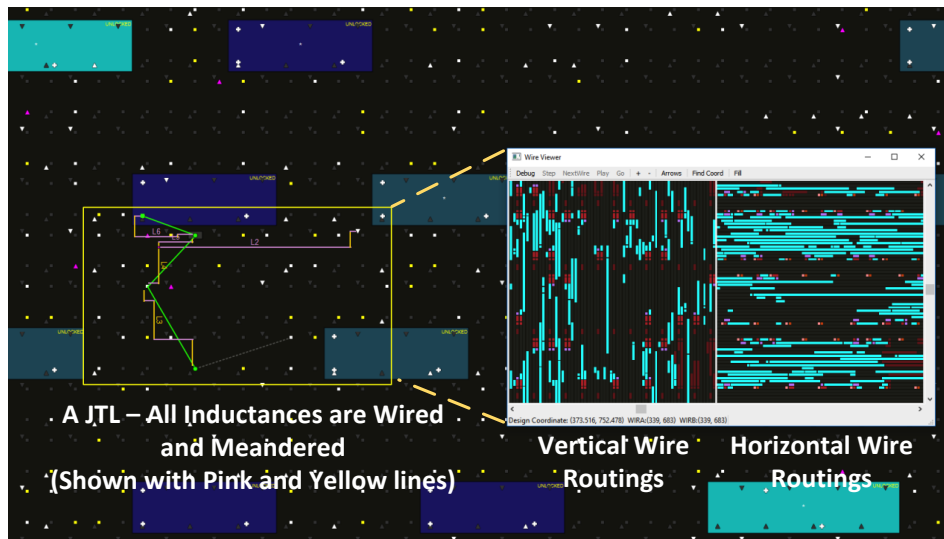
It is important to note that the DPC tool is under development, and is not mature yet. There are several challenges that exist in the design steps using DPC, some of which are nontrivial. For example, DPC may produce a layout in which a wire cannot be routed or a JTL cannot be placed due to congestions or non-optimized gate placements. Fig. 6-13 shows

an example JTL with missing wires in a congested routing area along with a successfully placed and routed JTL inside the DPC tool environment.

6.4. RQL design challenges and solutions



(a)



(b)

Fig. 6-13 (a) a JTL with missing wires in a congested routing area vs. (b) a successfully placed and wired JTL inside the DPC tool.

Similar to the CMOS design process, there are several important objectives that must be targeted during the RQL design process (for instance, we want to reduce the size of the design, in order to reduce the per chip cost). In this section we first discuss these objectives, and then we discuss the challenges that exist in the design process, as well as our solutions for them.

As mentioned, it is desirable to reduce the per chip cost of RQL circuits by reducing their area. One contributing factor of our interest is the density (or gate utilization), which refers to the portion of the area that is utilized by gates. This is usually determined after the place and route step. In the RQL technology, like in any circuit design, some portion of the area should be dedicated to interconnects, i.e., wires and the active transmission lines (JTLs). Since the number of gates is fixed after the synthesis step, an increase in the density leads to a lower area and cost.

The other important parameter is the latency or the timing delay between the inputs and outputs. As the delay reduces the operating speed of the design improves. However, increasing the speed (or decreasing the latency) could be challenging beyond some point, due the limitations imposed by the RQL timing methodology and the design architecture.

An important goal of this project is to automate the design process, and minimize the amount of manual work required for closing a design within the DPC. In general,

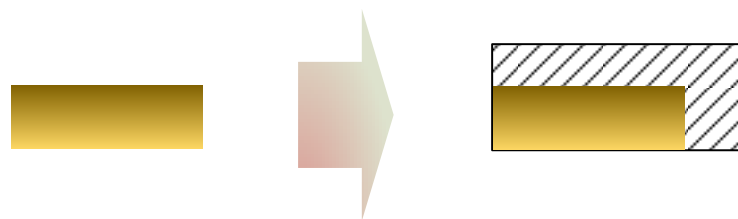


Fig. 6-14 The size of standard cells are virtually increased for optimized placement.

minimizing the manual work is an essential milestone in automating the design process of new technologies.

A. Placement and routing challenges and solutions

The current version of the DPC tool is not optimized for the initial placement of the gates. Instead, we use existing commercial tools, e.g., Cadence’s Encounter, for automatic placement and routing. By employing this approach, we were able to significantly reduce the required manual work. We note that the existing tools are optimized for CMOS technologies and cannot be readily used in the RQL design process. For instance, during the placement step, the main objectives of the existing tools are to minimize the routing distance between the gates and reduce the wiring capacitance. While these objectives hold,

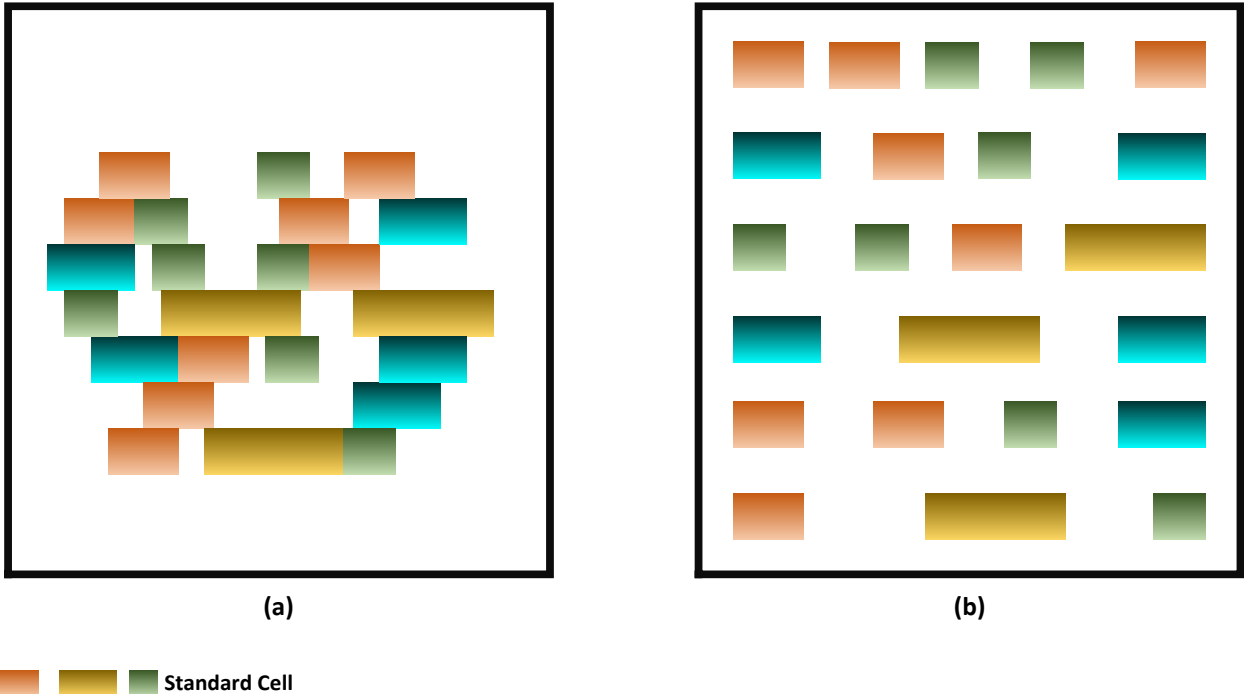


Fig. 6-15 Placed RQL standard gates (a) before optimization and (b) after optimization.

more restrictions exist in the RQL technology (e.g., it is not desirable to have RQL gates placed adjacent to each other, because some room is required for the JTLs that connect to the inputs and outputs of the gates). Due to the fanout limitations of the standard RQL gates, they cannot be directly connected to each other. Instead, the connections must pass through one or more JTL(s).

To overcome this challenge, we modified the standard CMOS placement process that existed in commercial tools and virtually increased the size of standard cells, so that after placement, some empty room becomes available. This modification is illustrated in Fig. 6-14. The height of each cell is doubled, because it fits within the standard cell grid. The cell's length, on the other hand, is only increased by 30%. In addition, we modified the

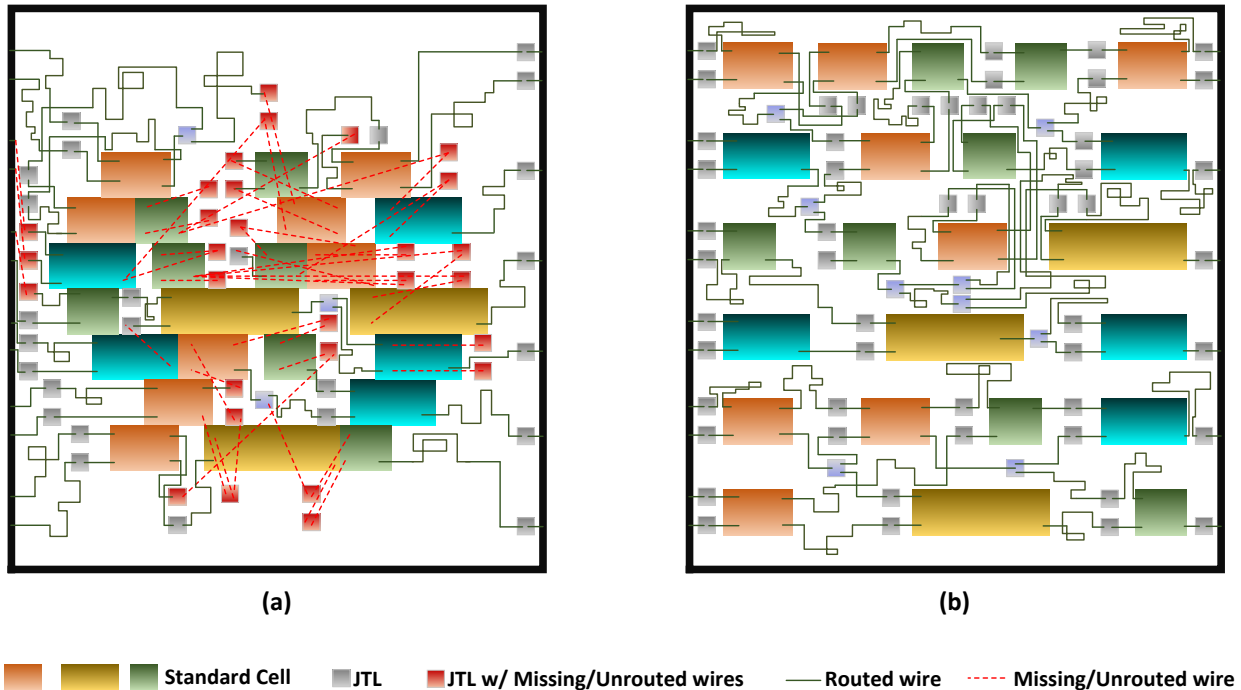


Fig. 6-16 Placement and routing of an example design (a) before and (b) after optimization.

gate placement flow to distribute the gates uniformly across the chip area. Fig. 6-15 shows two gate placement scenarios using the unoptimized and optimized flows. In the unoptimized flow, the gates are placed close to each other, which in turn leads to wire congestion. As a result, routing and interconnect circuitry placement becomes unsuccessful (as seen in Fig. 6-16). On the other hand, the optimized flow leaves enough room between the gates, and succeeds in placing the interconnects.

Fig. 6-17-Fig. 6-19 show how these simple optimization steps improve the design closure time of some ARM Cortex M0+ blocks. We report the number of missing (un-routed) wires and JTLs, as well as failures in meandering wires. The errors cannot be automatically fixed, and they each require extra manual effort, leading to an increase in the time required to close the design. A linear increase in number of missing (un-routed) wires, JTLs, and etc. leads to exponential growth in design closing time. We show that the optimizations can

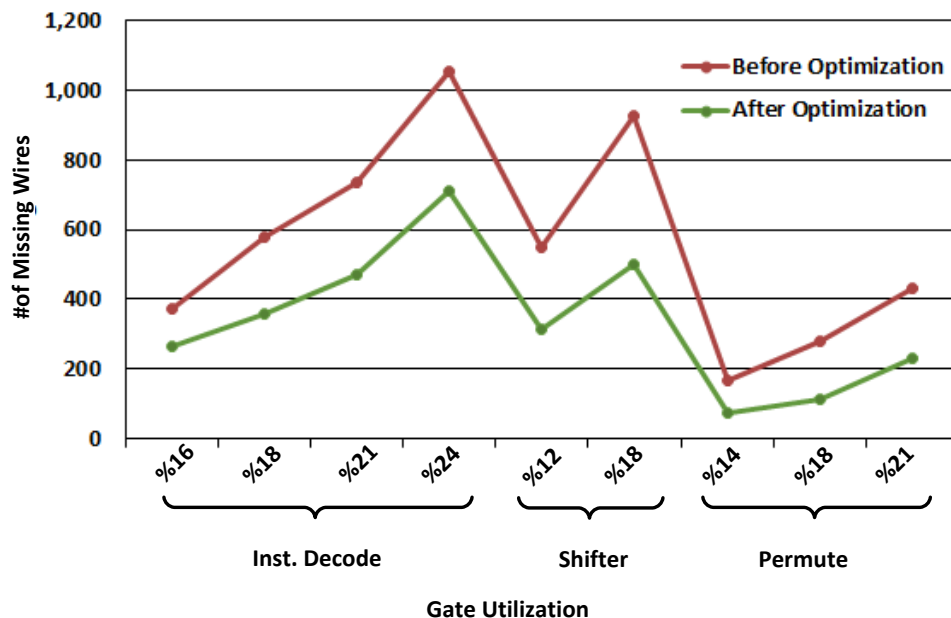


Fig. 6-17 Number of missing/unrouted wires for different designs with different gate utilizations before and after gate placement optimization

reduce the number interconnect failures (i.e., the number of wires that failed to be routed/meandered and the number of JTLs that failed automatic placement) by up to %50. To illustrate the effect of our optimized flow, we note that designing a representative block (decoder) using the old flow requires several days, while the new flow closes a design in less than one hour. Fig. 6-19 compares the design before and after the optimization; in Fig. 6-19 (a) we show how our algorithm to even distribution of the gates in the design area, and Fig. 6-19 (b-c) depicts the heat map of the wiring and JTL congestions before and after the optimization.

B. Fanout problems

Another challenge of the RQL design process is the fanout problem that causes difficulties in routing circuits with many inverters. An RQL inverter requires an extra input in addition

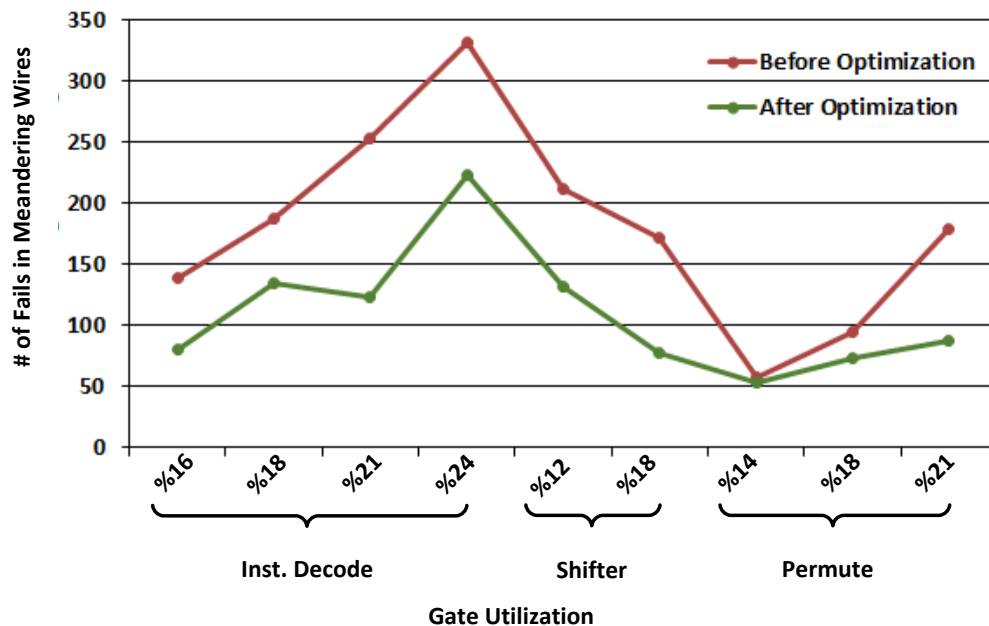


Fig. 6-18 Number of failures in meandering wires for different designs with different gate utilizations before and after gate placement optimization

to its regular input. This input comes from a pulse generator that is shared among all the inverters of the design. We improved this design flow by changing the synthesis library, and making the synthesis tool to employ special inverters that have local pulse generators. This approach leads to an increase in the circuit area, but eliminates many routing and fanout issues for the pulsegen signal.

C. Miscellaneous improvements

The DPC tool is still under development, and a goal of this project is to improve the efficacy of the tool by providing the developers with useful information and insights. These insights are obtained through various design iterations, and they usually lead to new features being added to DPC. Their main goal is to automate the design flow of superconducting circuits.

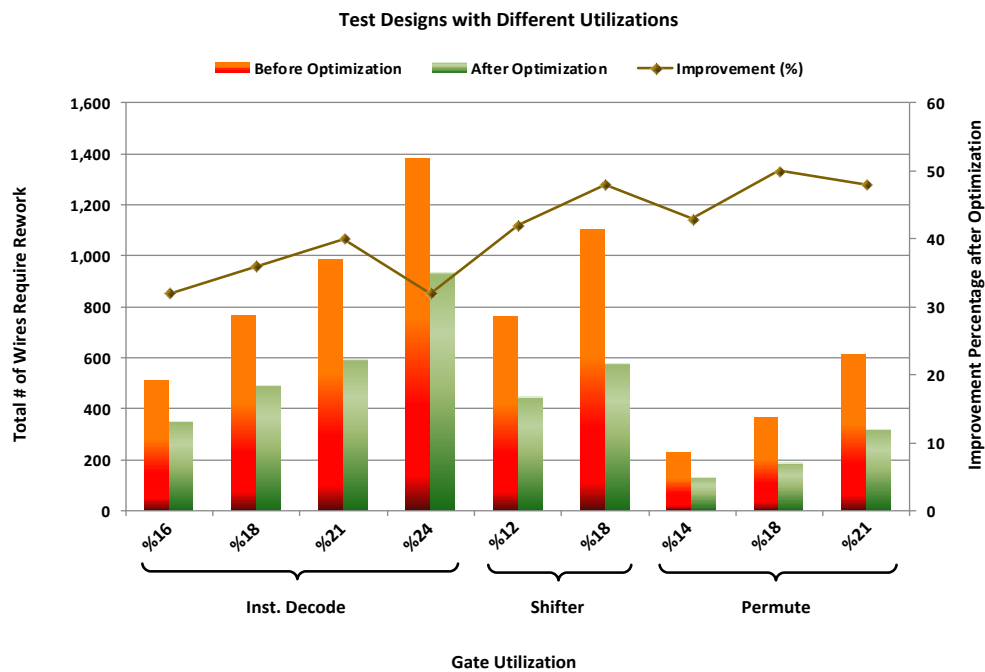


Fig. 6-19 Total number of wires (missing wires + missing JTLs + failure in meandering) require rework for different designs with different gate utilizations before and after gate placement optimization.

Some examples of the added features are explained next.

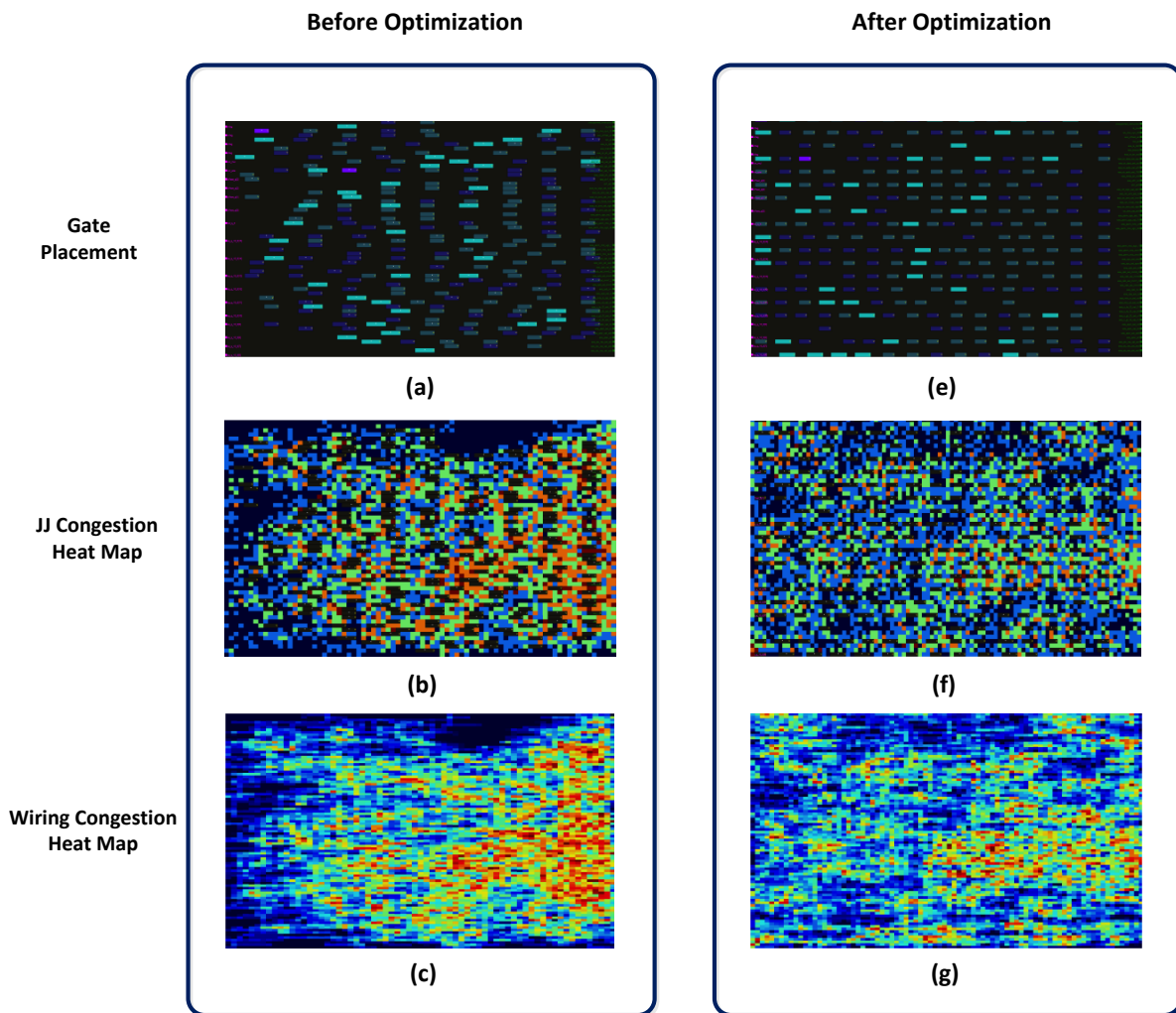
The ability to move (drag and drop) the standard cells (gate tiles) within the DPC user interface, is one the features that was added to alleviate the placement problem. Although placement using commercial tools improves the design time, the ability to drag and drop gates within the DPC tool enables further optimization of the design.

Another added feature is the ability to modify the IO pin placements inside the DPC tool as well as the capability of having the IO pins on all sides of the blocks. Primarily, the only acceptable places for pins were the left and the right sides of the design block. In addition, DPC was modified to route with wires that are no longer than a certain limit. This leaves room for adjusting the inductance of the wires without violating any design rules. These added features allow the designers to further optimize design parameters such as density, latency, throughput, and etc.

6.5. Design prototype

To demonstrate the automated RQL design process, several blocks of the ARM Cortex M0+ processor (instruction decoder, shifter, and permutation block) were implemented using the DPC tool. Fig. 6-20 shows three main blocks that are designed in our current test chip. These blocks operate on a 3GHz resonator and the biggest block, i.e., the shifter, takes less than 0.5mm². The power consumption of the shifter in simulation is only ~7.8μW. All the designs we implemented using our automated RQL design flow.

The test chip for the designed blocks is severely IO limited. For this reason, input and output test wrappers were designed to multiplex and de-multiplex the main block signals. The number of allocated pins for each block is three (two inputs and one output). The input test wrapper is a gateless block that uses the timing characteristic of the RQL technology to



(a)-(c) :Gate Placement, JJ Congestion Heat Map, and Wiring Congestion Heat Map (Before Optimization)
(d)-(f) : Gate Placement, JJ Congestion Heat Map, and Wiring Congestion Heat Map (After Optimization)

Fig. 6-20 Gate placement and heat map of the wiring and JTL congestions before and after the optimization step.

Table 6-1 Summary of the implemented blocks in the RQL technology

<i>Parameters \ Design Name</i>	<i>Inst. Decoder</i>	<i>Shifter</i>	<i>Permute</i>
<i># of JTL</i>	2565	6467	3537
<i># of Allocated JJ</i>	5130	12934	7074
<i># of Allocated Resonators</i>	2565	6464	3537
<i># of Gates</i>	259	608	306
<i>Gate Utilization</i>	15	14	11
<i>Resonator Frequency (GHz)</i>	3	3	3
<i>Maximum Operating Phase (Latency)</i>	20	40	56
<i>Block Size (mm²)</i>	0.198	0.452	0.223
<i>Power (μW)*</i>	0.31	0.76	0.41
<i># of Inputs/Outputs</i>	21/50	32/62	30/35

**Power = (0.1(a) × (# of JJ) × f) × α ; α = activity factor = %20*

de-multiplex a single (serial) input signal into multiple inputs for the design under test (DUT). Output test wrapper, on the other hand, consists of gates that multiplex the parallel outputs into one serial output signal. The output test wrapper also incorporates the timing and delay feature of the RQL technology. In addition to the test wrappers, a cyclic redundancy check (CRC) circuit is implanted to further compress the number of output pins and to minimize the wiring interface between the DUT and the output test-wrappers.

To compare the power consumption of the prototype blocks with that of a contemporary technology, we synthesized and APRed the same blocks at 3GHz clock frequency using 65nm CMOS technology. The reported power of each block appears in Table 6-2. The total power consumption of the RQL circuits is four orders of magnitude smaller than that of the

Table 6-2 Power comparison for blocks implemented in RQL and 65nm CMOS technologies

<i>Parameters \ Design Name</i>	<i>Inst. Decoder</i>	<i>Shifter</i>	<i>Permute</i>
<i>Power of design implemented in RQL @3GHz</i>	0.31 μ W	0.76 μ W	0.41 μ W
<i>Power of design implemented in 65nm CMOS @3GHz</i>	1.70 mW	7.84 mW	2.42 mW
<i>Energy/Gate of design implemented in RQL @3GHz</i>	7.9×10^{-17} J/gate	8.5×10^{-17} J/gate	9.2×10^{-17} J/gate
<i>Energy/Gate of design implemented in 65nm CMOS @3GHz</i>	2.6×10^{-15} J/gate	3.15×10^{-15} J/gate	2.63×10^{-15} J/gate

CMOS circuits. After considering the cooling factor for superconducting devices at 4K, energy consumption per gate is still two orders of magnitude smaller in RQL designs. We should note that RQL designs are still under development and not as optimized as the CMOS counterparts.

Similar to any digital CMOS design, the functionality of the digital RQL designs is first verified through behavioral Verilog simulations. In addition, extra functionality and timing analysis is performed on the designed blocks using VHDL functional and timing simulations. The VHDL model of the design is extracted within the DPC tool. It contains all the components of the design including gates and JTLs, as well as the timing information of the components.

6.6. Conclusions

Several beyond CMOS technologies, including the RQL technology discussed in this chapter, are promising alternatives to CMOS that can enable future high speed and low power applications. In this chapter we proposed design automation methods that reduce the

design time and increase the scalability level of RQL circuits. We also discussed our contributions in the development of the DPC tool (an under-development RQL design tool). We showed that our proposed method reduces the manual placement and routing adjustments by up to almost %50, which leads to significantly shortened design times (from several days down to less than an hour). Finally, we implemented several blocks of ARM Cortex M0+ Processor.

Chapter 7

Concluding Remarks

Today's SoCs usually consist of billions of transistors accounting for both digital and analog blocks. But integrating such massive blocks on a single chip involves several challenges. For instance, transferring analog blocks from an older technology to newer ones (to benefit from scaling) incurs a significant design cost. Automating the design processes can reduce the design time, and time to market of the complex SoCs with both digital and analog blocks.

Furthermore, the exponential growth for IoT devices necessitates small and low power circuits; otherwise, there will not be enough energy to fuel the devices of the trillion sensors era. Hence, new devices and architectures must be investigated to meet the power and area constraints for wireless sensor networks (WSNs).

In this dissertation we have addressed the aforementioned challenges, by focusing on automating the design process of analog designs in advanced CMOS technology nodes, as well as RQL superconducting circuits. In addition, we have developed a communication protocol suitable for low power mm³-scale WSNs and designed several baseband DSPs for them. This chapter presents the conclusions and the future directions of this dissertation.

7.1. Conclusions

A. VLSA Methodology

The analog design automation technique (called VLSA) introduced in this work employs digital automatic placement and routing tools to synthesize and lay out analog blocks along with digital blocks in a cell-based design approach. In the VLSA design approach, analog complexity is relaxed in favor of automation, and as a result, the (re-)design cycle is dramatically shortened. Furthermore, APR enables higher levels of integration and better scaling of analog designs, and thus benefits from Moore's law without being hindered by the increasingly complex design rules. We used this technique to design a digital-to-analog converter (DAC) and with minimal design time and effort, the VLSA DAC achieved a performance comparable to conventional DACs. We predict that with the rapid growth in the number of design rules, a cell-based design approach becomes a necessity, as it is much faster than full-custom design approaches. Semiconductor scaling only makes the problem worse, and it is only a matter of time before complex analog designs are forced in this direction; a path adopted decades ago by digital designers. Our cell-based approach also allows porting the design into other processes with negligible effort, as all of the code and scripts are reused with only minor adjustment.

B. Large Scale Superconducting Circuits

We investigated a low power high frequency technology node called reciprocal quantum logic (RQL), which is based on superconductor materials and is a promising alternative for CMOS. We introduced several techniques to automate the design process of RQL circuits, and enable the design of VLSI-scale systems in this technology. We improved the RQL

design flow by adding features to an (under development) superconducting circuit design tool called datapath compiler (DPC), and also presented several placement optimization techniques. We showed that our techniques significantly reduce the closing time of several representative circuits and enable integration of large-scale designs in this technology.

C. WSNs for IoT Applications

In the area of low-power mm³-scale wireless sensor nodes, we presented several baseband DSPs, modulators and demodulators, and digital feedback gain controllers that are used in digitally assisted transceivers. These design techniques allow high frequency transceivers to operate with the power constraints of standalone IoT nodes. We demonstrated the first full-system standalone mm-scale sensor node in a heterogeneous die-stacking system. The standalone node (with no external connections) performs a node-to-base-station 2.5m communication, and sustains autonomous operation (with the micro battery) for 17 minutes.

7.2. Future Directions

The line of work presented in this dissertation can be followed up in several directions. In this section we briefly present some of these directions.

The VLSA methodology has proven to be successful in cell-based analog circuits, but can be extended to be employed in any analog design, specifically those that cannot be divided into smaller blocks. This is the natural extension of the VLSA and can be beneficial in the light of analog design challenges.

Another interesting extension of the VLSA methodology is the study of novel calibration techniques for the automatically designed analog circuits. Since automatic placement and

routing of analog circuits produces unpredictable non-linearities in the analog blocks, they heavily rely on calibration. But are the existing calibration techniques sufficient? Or do we need techniques that are specific for VLSA circuits? The answers to these questions can be an interesting future work.

Design automation of the RQL circuits is an ongoing work and it still needs a lot of attention in order to become a mature technology. A powerful placement tool that is aware of RQL constraints can significantly reduce the manual work required in the RQL design. This can be achieved by incorporating the RQL constraints into commercial gate-placer tools, in order to make them RQL-aware. Another approach is to incorporate standard gate placement into the DPC tool, which is still under development. Finally, a natural extension of the work presented in this dissertation is to apply design automation techniques to the RQL analog/mixed signal.

APPENDIX

As mentioned, ubiquitous sensing is projected to reach volumes of 1000 sensors per person by 2025 [24], a number that will dwarf the current cell phone market. Considering that today we are surrounded by ~100 sensors at work, in the car, and at home; and there is growing demand for “smarter” devices in many applications, this target may not be far off. With low-power computing as the initial spark, sensor nodes have reduced in volume by 100× over the past decade, with cubic-mm sensors now a reality. The primary challenge of mm-scale sensors is power management due to severely limited harvested and stored energy. Furthermore, mm-scale battery technology requires efficient, integrated power conversion, meticulous duty cycling of high-power modules, ultra-low leakage power, and under-voltage/over-current battery-protection circuitry. Heterogeneous die stacking alleviates these challenges, enabling components to be designed in their optimal processes. However, seamless interaction across dies is essential, in particular at the interface circuits, for sustained operation with long lifetimes.

In addition to the UWB transceiver, discussed in Chapter 5, another RF system for the modular die stacked mm³ sensor node is studied and designed. The new SoC is a low power duty cycled GPS receiver. The GPS receiver chip includes the analog front-end (AFE), PLL, ADC, and on-chip baseband digital signal processor (DSP).

At the chips input input, the GPS signal is received, down-converted, filtered and amplified through LNA, mixer, GmC filter, and variable gain amplifier respectively. The high level architecture is shown in Fig 8-1 The amplified analog signals are received with an ADC (I and Q channel) and converted to digital domain for more processing and analysis. The on chip PLL provides a clock signal for the rest of the blocks as well as the LO signal for mixer. The AFE supports two frequency modes: single band mode (L1) as well as tri-band mode (L1, L2, and L5). The power of received GPS signal is around -110dBm. The signal is buried under the noise floor in AFE, but the output is coming out after the correlation with PN code. Our main contributions are the DSP and the analog to digital converter.

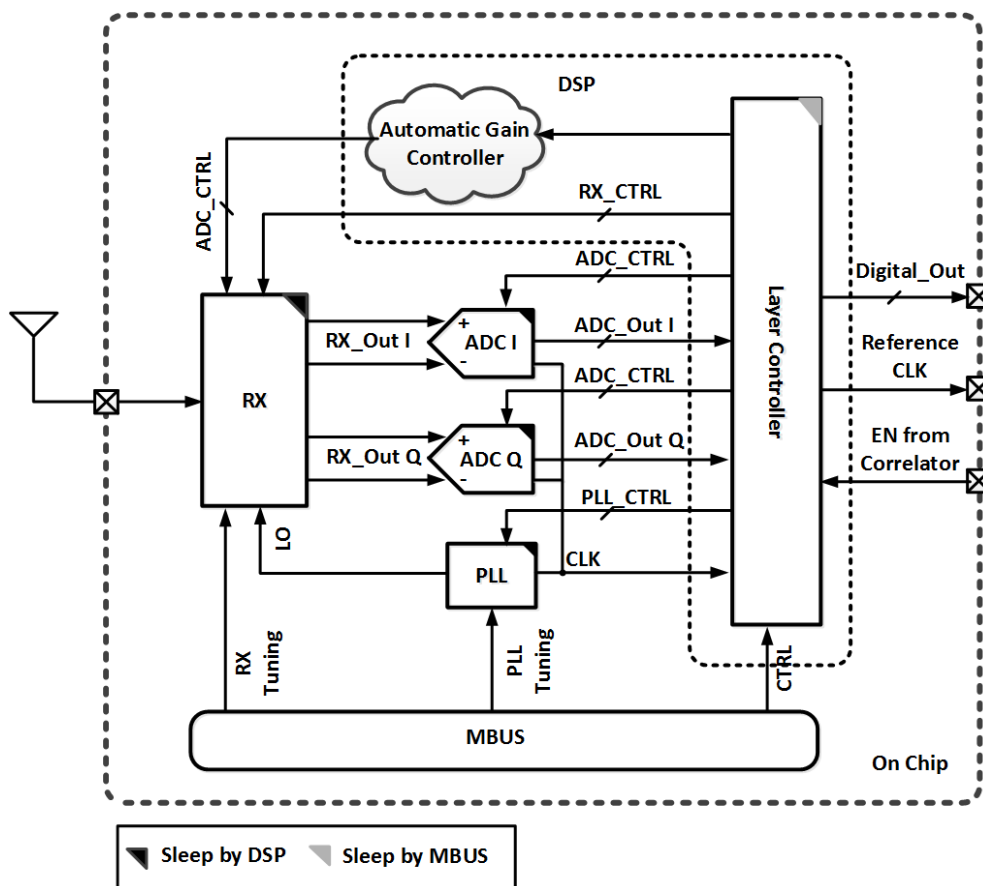


Fig. 8-1 High level schematic of the GPS receiver

For chip-to-chip communication, we use MBus, a specially designed interconnect that meets the unique constraints of millimeter-scale sensor nodes [73]. MBus is clockless, low power, robust, and fully synthesizable, and it supports multi master communication. This module replaces the I²C chip-to-chip communication unit (explained in Chapter 5) that was used in earlier mm³ nodes. Our main contribution in this project is the design of the DSP module, the feedback loop gain controller and the programmable analog to digital converter.

ADC and DSP

Figure 8-2 shows the programmable comparators architecture used for the differential flash ADC. The embedded programmability allows us to set the thresholds of the comparators based on the analog signal levels. This comparator is used in a two bit flash ADC, with I and Q channels. Since the ADC is only two bits, comparators should be precise

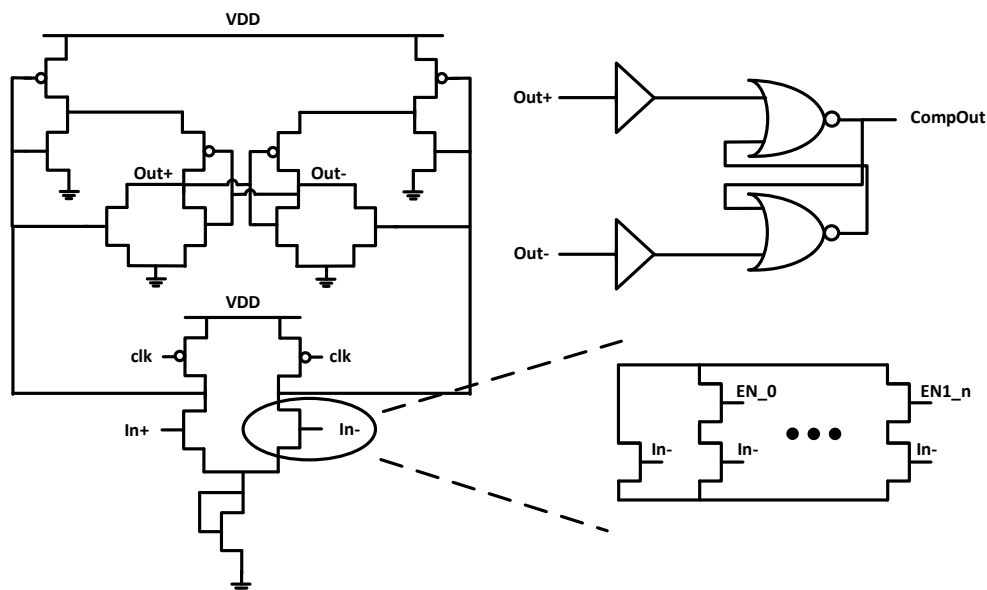


Fig. 8-2 Comparator block diagram

in order to convert the input data to the accurate digital outputs. ADC digital outputs are later processed in a digital signal processor (DSP).

The DSP unit monitors the received signals for a period of time and generates a histogram for $|I|+|Q|$ and, through the feedback control loop, adjusts the gain of the variable gain amplifier accordingly. The automatic gain adjustment is performed dynamically in the background while the receiver is turned on. DSP also includes a decoder in order to generate output signals with the appropriate signal formata for the correlator layer. Corellator layer is another chip layer in the mm^3 stack, which post processes the GPS received signal. All the blocks in AFE layer are power gated and DSP handles the duty cycling and power on/off sequence of the layer in the mm^3 stack in order to achieve the power requirements (a few μW for active power and a few nW sleep power). The GPS receiver is fabricated in 65nm technology, Fig. 8-3 shows the die photo of the mm^3 GPS receiver.

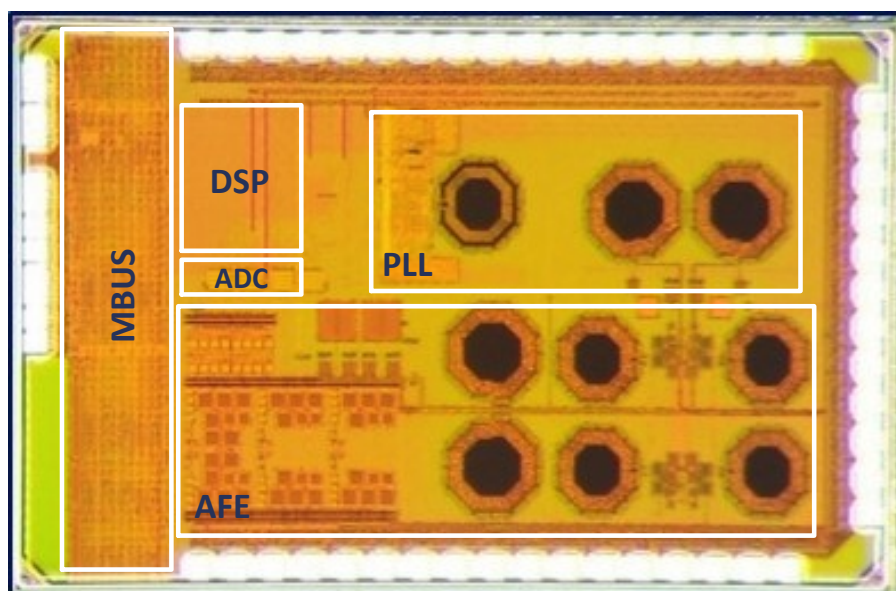


Fig 8-3 GPS receiver die photo

REFERENCES

- [1] "The Multiple Lives of Moore's Law", <http://spectrum.ieee.org>, retrieved Dec. 2015
- [2] "What is Moore's Law?", <http://www.extremetech.com> , retrieved Dec. 2015
- [3] "Look at Moore's Law in Action", <http://spectrum.ieee.org>, retrieved Dec. 2015
- [4] "Moore's Law and Analog/Digital Integration", <http://www.embedded.com>, retrieved Dec. 2015
- [5] G.F. Taylor, "The Challenges of Analog Circuits on Nanoscale Technologies", *Custom Integrated Circuits Conference (CICC), 2014 IEEE Proceedings of*, pp.1-6, 15-17 Sep. 2014
- [6] J. Xicheng, "Digitally-Assisted Analog and Analog-Assisted Digital IC Design", *Cambridge University Press*, Jul. 23, 2015
- [7] "Analog Circuits Benefit from Scaling Trends", <http://chipdesignmag.com>, retrieved Dec. 2015
- [8] "2012 Update Overview", *International Technology Roadmap for Semiconductors*, p. 1-5, 2013
- [9] W. Arden, et. al, "Towards a More-than-Moore Roadmap", *International Technology Roadmap for Semiconductors* , p. 8-9, Apr. 2011
- [10] "What is 'More than Moore'?", Mixed-Signal Foundry Experts, www.more-than-moore.com, retrieved Dec. 2015
- [11] "More-than-Moore Technology", <http://www.tsmc.com> , retrieved Dec. 2015
- [12] P. Harpe, A. Baschiroto, K.A.A. M., "High-Performance AD and DA Converters, IC Design in Scaled Technologies, and Time-Domain Signal Processing", *Springer*, Jul. 2014

- [13] B. Murmann, "Digitally Assisted Data Converter Design", *European Solid-State Circuits Conference (ESSCIRC)*, 2013
- [14] B. Murmann, "Digitally Assisted Analog Circuits", *Micro, IEEE*, vol.26, no.2, pp.38-47, 2006
- [15] K. Okada, S. Kousai, "Digitally-Assisted Analog and RF CMOS Circuit Design for Software-Defined Radio", *Springer*, 2011
- [16] G. Bell, "Bell's Law for the Birth and Death of Computer Classes", *Commun. ACM*, vol. 51, no. 1, pp. 86-94, Jan. 2008
- [17] T. Nakagawa, et. al, "1-cc Computer: Cross-Layer integration with UWB-IR Communication and Locationing", *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 964-973, Apr. 2008
- [18] J. Bryzek, "Emergence of a \$Trillion MEMs Sensor Market", *sensorscon*, Mar. 2012
- [19] "The World's Smallest Computer", <http://www.computerhistory.org>, retrieved Dec. 2015
- [20] J. Bryzek, "Emergence of Trillion Sensors Movement", *IEEE MEMS*, Jan. 2014
- [21] "An Introduction to the Internet of Things (IoT)", *Lopez Research*, Nov 2013
- [22] "Rise of the Embedded Internet", *Intel Embedded Processors, Intel Corporation*, 2010
- [23] D. Evans, "The Internet of Things: How the Next Evolution of the Internet is Changing Everything", *Cisco Internet Business Solutions Group*, Apr. 2011
- [24] J. Bryze, "Emergence of a \$Trillion MEMs Sensor Market", *Sensorcon*, 2012
- [25] K. Karimi, G. Atkinson, "What the Internet of Things (IoT) Needs to Become a Reality", Jun. 1013
- [26] M. White, "Pattern Matching Might Solve World Hunger", <http://blogs.mentor.com>, Jan 2010, retrieved Dec. 2015
- [27] "Does Double Patterning Mean the End of the World?", <http://Electronicdesign.com/Fpgas/Does-Double-Patterning-Mean-End-World>, retrieved Dec. 2015
- [28] R. Todd, et. al, "Design Rule Checking", in *EDA for IC Implementation, Circuit Design, and Process Technology*, CRC Press, 2006

- [29] D. L. Gonzalez, et. al, "EDA for RF and Analog Front-Ends in the 4G Era: Challenges and Solutions", European Conference on Circuit Theory and Design, vol., no., pp.24-27, Aug. 2007
- [30] D. Macmillen, et. al, "An Industrial View of Electronic Design Automation", *Computer-Aided Design (TCAD) of Integrated Circuits and Systems, IEEE Transactions On*, vol. 19, no. 12, Dec. 2000
- [31] "Analog EDA Finally Automated", http://www.Eetimes.com/Document.Asp?Doc_Id=1326192, retrieved Dec. 2015
- [32] G. Gielen, W. Sansen, "Symbolic Analysis for Automated Design of Analog Integrated Circuits", *Springer*, 1991
- [33] S. Balkir, G. Dündar, A. S. Ögrenci, "Analog VLSI Design Automation", *CRC Press*, Jun. 2003
- [34] G.V.D Plas, G. Georges, S. Willy, "A Computer-Aided Design and Synthesis Environment for Analog Integrated Circuits", *Springer*, 2002
- [35] M.F.M. Barros, J.M.C. Guilherme, N.C.G. Horta, "State-of-the-Art On Analog Design Automation", *Springer*, 2010
- [36] R.A. Rutenbar, "Design Automation for Analog: the Next Generation of Tool Challenges", *IEEE/ACM International Conference on Computer-Aided Design ICCAD*, vol., no., pp.458-460, Nov. 2006
- [37] R. Phelps, M. Krasnicki, R.A. Rutenbar, L.R. Carley, J.R. Hellums, J.R., "Anaconda: Simulation-Based Synthesis of Analog Circuits via Stochastic Pattern Search", *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol.19, no.6, pp.703-717, Jun. 2000
- [38] R.A. Rutenbar, "Analog Design Automation: Where Are We? Where Are We Going?", *Custom Integrated Circuits Conference, Proceedings of the IEEE*, pp.13.1.1-13.1.7, 9-May 1993
- [39] B. Preas and P. Karger, "Automatic Placement: A Review of Current Techniques", *Proceedings of DAC*, Jun. 1986
- [40] R.A. Rutenbar, "CAD Techniques To Automate Analog Cell Design", *ACM/DAC*, Jun. 2001

- [41] D. Gajski, R. Kuhn, "Guest Editors' Introduction: New VLSI Tools", in *Computer*, vol. 16, no. 12, pp.11-14, Dec. 1983
- [42] D. Gajski, "Silicon Compilers", *Addison-Wesley*, 1987
- [43] R. A. Rutenbar, G.G.E. Gielen; B.A. Antao, "A Case Study of Synthesis for Industrial-Scale Analog IP: Redesign of the Equalizer/Filter Frontend for an ADSL CODEC", *Computer-Aided Design of Analog Integrated Circuits and Systems*, Wiley-IEEE Press, pp.211-216, 2002
- [44] R.A. Rutenbar, "Analog Synthesis (and Verification) Revisited: What's Missing", *SMACD*, Sep. 2012
- [45] G.G.E. Gielen, R.A. Rutenbar, "Computer-Aided Design of Analog and Mixed-Signal Integrated Circuits", *Proceedings of the IEEE*, vol.88, no.12, pp.1825-1854, Dec. 2000
- [46] L. Hongzhou, A. Singhee, R.A. Rutenbar, L.R. Carley, "Remembrance of Circuits Past: Macromodeling by Data Mining in Large Analog Design Spaces", *Design Automation Conference, Proceedings of*, pp.437-442, 2002
- [47] H. Chang, A. Sangiovanli-Vincentelli, F. Balarin, E. Charbon, U. Choudhury, G. Jusuf, E. Liu, E. Malavasi, R. Neff, P.R. Gray, "A Top-Down, Constraint-Driven Design Methodology for Analog Integrated Circuits", *Custom Integrated Circuits Conference, IEEE Proceedings of*, pp.8.4.1-8.4.6, May 1992
- [48] Y. Park, D. D. Wentzloff, "An All-Digital PLL Synthesized from a Digital Standard Cell Library in 65nm CMOS", *Custom Integrated Circuits Conference (CICC)*, 2011
- [49] S. Weaver, B. Hershberg, U-K. Moon, "Digitally Synthesized Stochastic Flash ADC Using Only Standard Digital Cells", *IEEE Transactions On Circuits and Systems I*, vol. 61, no. 1, Jan. 2014
- [50] W. Deng, D. Yang, T. Ueno, T. Siriburanon, S. Kondo, K. Okada, A. Matsuzawa, "15.1 A 0.0066mm² 780μw Fully Synthesizable PLL with A Current-Output DAC and an Interpolative Phase-Coupled Oscillator Using Edge-Injection Technique", *International Solid-State Circuits Conference (ISSCC) Dig. Tech. Papers*, 2014
- [51] W. Deng, D. Yang, A. T. Narayanan, K. Nakata, T. Siriburanon, K. Okada, A. Matsuzawa, "A 0.048mm² 3mw Synthesizable Fractional-N PLL with a Soft

- Injection-Locking Technique”, *International Solid- State Circuits Conference (ISSCC) Dig. Tech. Papers*, 2015
- [52] M. Faisal, D. D. Wentzloff, “An Automatically Placed-and-Routed ADPLL for the Medradio Band Using PWM to Enhance DCO Resolution”, *Radio Frequency Integrated Circuit Symposium (RFIC)*, 2013
- [53] S. Ryu, J. Kim, “Cell-Based Construction of Mixed-Signal Systems Using Co-Design Flow of IC Compiler and Custom Designer: a Digital PLL Example”, *Synopsys User Groups (Snug)*, 2014
- [54] “Understanding Data Converters Data Converter”, *Texas Instruments*, 1999
- [55] W. Kester, “The Data Conversion Handbook”, *Chapter 5, Analog Devices*, 2005
- [56] C.H. Daigle, “Switched-Capacitor DACs Using Open-Loop Output Drivers and Digital Predistortion”, *PhD Dissertation*, Aug. 2010
- [57] C-H. Lin, F. Van Der Goes, J. Westra, J. Mulder, Y. Lin, E. Arslan, E. Ayranci, X. Liu, K. Bult, “A 12b 2.9Gs/S DAC with $IM_3 \ll -60dB_c$ Beyond 1GHz in 65nm CMOS”, *International Solid-State Circuits Conference (ISSCC) Dig. Tech. Papers*, 2009
- [58] W-T. Lin, T-H. Kuo, “A 12b 1.6Gs/S 40mw DAC in 40nm CMOS with $>70dB$ SFDR over Entire Nyquist Bandwidth”, *International Solid- State Circuits Conference (ISSCC) Dig. Tech. Papers*, 2013
- [59] M. S. Mehrjoo, J. F. Buckwalter, “A 10-b, 300-MS/s Power DAC with 6-Vpp Differential Swing”, *Radio Frequency Integrated Circuit Symposium (RFIC)*, 2013
- [60] K. Doris, J. Briaire, D. Leenaerts, M. Vertreg, A. Van Roermund, “A 12b 500MS/s DAC with $>70db$ SFDR Up To 120MHz in 0.18 μm CMOS”, *International Solid-State Circuits Conference (ISSCC) Dig. Tech. Papers*, 2005
- [61] Y. Tang, J. Briaire, K. Doris, R. Van Veldhoven, P. Van Beek, H. Hegt, A. Van Roermund, “A 14b 200MS/s DAC with $SFDR > 78dB_c$, $IM_3 < -83dB_c$ and $NSD < -163dB_m/Hz$ Across the Whole Nyquist Band Enabled By Dynamic-Mismatch Mapping”, *IEEE Symposium On VLSI Circuits (VLSIC)*, 2010
- [62] F. Gong, S. Basir-Kazeruni, L. He, H. Yu, “Stochastic Behavioral Modeling and Analysis for Analog/Mixed-Signal Circuits”, *IEEE Transactions On Computer-Aided Design (TCAD) of Integrated Circuits and Systems*, vol. 23, no. 1, Jan. 2013

- [63] L. Capodiecici, "Beyond 28nm: New Frontiers and Innovations in Design For Manufacturability at the Limits of the Scaling Roadmap", Global Foundries
- [64] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks", *IEEE Commun. Mag.*, pp. 102-114, Aug. 2002
- [65] S. Hanson, M. Seok, Y-S. Lin, Z. Foo, D. Kim, Y. Lee, N. Liu, D. Sylvester, and D. Blaauw, "A Low-voltage Processor for Sensing Applications with Picowatt Standby Mode", *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1145-1155, Apr. 2009
- [66] K. Sundaresan, G. K. Ho, S. Pourkamali, F. Ayazi, "Electronically Temperature Compensated Silicon Bulk Acoustic Resonator Reference Oscillators", *IEEE J. Solid-State Circuits*, vol. 42, no. 6, pp. 1425-1434, Jun. 2007
- [67] Cymbet Corp., "Enerchip Smart Solid State Batteries", <http://www.Cymbet.com/Products/Enerchip-Solid-State-Batteries.Php>, 2012, retrieved Nov. 2013
- [68] Samsung SDI, "Prismatic Rechargeable Battery", <http://Samsungsdi.com/Battery/Prismatic-Rechargeable-Battery.Jsp>, retrieved Nov. 2013
- [69] Y. Nishi, "Lithium ION Secondary Batteries; Past 10 Years and the Future", *J. Power Sources*, vol. 100, no. 1-2, pp. 101-106, Nov. 2001
- [70] A.H. Zimmermann, M.V. Quinzio, "Performance of Sony 18650-HC Lithium-ION Cells for Various Cycling Rates", *Aerospace Corp., Tech. Rep.*, Jan. 2010
- [71] S. Hossain, A. Tipton, S. Mayer, and M. Anderman, "Lithium-ION Cells for Aerospace Applications", *Proc. 32nd Intersociety Energy Conversion Eng. Conf.*, vol. 1, pp. 35-38, Aug. 1997
- [72] X. Wang, Y. Yu, B. Busze, H. Pflug, A. Young, X. Huang, C. Zhou, M. Konijnenburg, K. Philips, and H. D. Groot, "A Meter-Range UWB Transceiver Chipset for Around-the-Head Audio Streaming", *ISSCC Dig. Tech. Papers*, pp. 450-451, Feb. 2012
- [73] Y-S Kuo; P. Pannuto, G. Kim; Z. Foo; I. Lee; B. Kempke, P. Dutta, D. Blaauw, Y. Lee, "MBus: A 17.5 pJ/bit/chip Portable Interconnect Bus for Millimeter-Scale Sensor Systems with 8nW Standby Power", *Custom Integrated Circuits Conference (CICC), 2014 IEEE Proceedings of*, vol., no., pp.1-4, Sep. 2014
- [74] "Intel Calls for 3D IC", <http://www.Monolithic3d.com>, retrieved Dec. 2015
- [75] "Superconductors", <http://www.superconductors.org/index.htm>, retrieved Dec. 2015

- [76] "Superconductors", http://wps.prenhall.com/wps/media/objects/4680/4793217/ch21_06.htm, retrieved Dec. 2015
- [77] K-K. Huang, D.D. Wentzloff, "A 1.2MHz 5.8 μ W Temperature-Compensated Relaxation Oscillator in 130nm CMOS", *IEEE Transactions On Circuits and Systems – II*, vol. 61, no. 5, pp. 334-338, May 2014
- [78] J.K. Brown, K-K. Huang, E. Ansari, R. R. Rogel, Y. Lee, D. D. Wentzloff, "An Ultra-Low-Power 9.8GHz Crystal-Less UWB Transceiver with Digital Baseband Integrated in 0.18 μ m BiCMOS", *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 442-443, Feb. 2013
- [79] E. Ansari, D.D. Wentzloff, "A 5mW 250MS/s 12-Bit Synthesized Digital to Analog Converter", *IEEE Custom Integrated Circuits Conference (CICC)*, pp. 1-4, Sep. 2014
- [80] M. Faisal, "Towards Very Large Scale Analog (VLSA): Synthesizable Frequency Generation Circuits", *PhD Dissertation*, 2014
- [81] W. Chen, A.V. Rylyakov, V. Patel, J.E. Lukens, K.K. Likharev, "Rapid Single Flux Quantum T-FlipFlop Operating Up To 770 GHz", *Applied Superconductivity, IEEE Transactions on*, vol.9, no.2, pp.3212-3215, Jun. 1999
- [82] D.K. Brock, O.A. Mukhanov, J. Rosa, "Superconductor Digital RF Development for Software Radio", *Communications Magazine, IEEE*, vol.39, no.2, Feb 2001
- [83] H. Hayakawa, N. Yoshikawa, S. Yorozu, A. Fujimaki, "Superconducting Digital Electronics", *Proceedings of the IEEE*, vol.92, no.10, pp.1549-1563, Oct. 2004
- [84] O.A. Mukhanov, D. Gupta, A.M. Kadin, V.K. Semenov, "Superconductor Analog-To-Digital Converters", *Proceedings of the IEEE*, vol.92, no.10, pp.1564-1584, Oct. 2004
- [85] S. Tolpygo, "Superconductor Digital Electronics: Scalability and Energy Efficiency Issues", *arXiv*
- [86] Q.P. Herr, A.Y. Herr, O.T. Oberg, A.G. Ioannidis, "Ultra-Low-Power Superconductor Logic", *Journal of Applied Physics*, vol. 109, pp. 103903-103910, May 2011
- [87] K. Reneris, B. Smith, D. Carmean, "Datapath Compiler", *Microsoft Internal Publication*, 2016
- [88] J.M. Shalf, R. Leland, "Computing Beyond Moore's Law" in *Computer*, vol. 48, no. 12, p. 14-23, Dec. 2015