Low-Power Design of Digital VLSI Circuits around the Point of First Failure

Thèse N° 9180

Présentée le 25 janvier 2019

à la Faculté des sciences et techniques de l’ingénieur
Laboratoire de circuits pour télécommunications
Programme doctoral en microsystèmes et microélectronique

pour l’obtention du grade de Docteur ès Sciences

par

ANDREA BONETTI

Acceptée sur proposition du jury

Prof. C. Enz, président du jury
Prof. A. P. Burg, Dr A. S. Teman, directeurs de thèse
Prof. M. Alioto, rapporteur
Prof. D. Bol, rapporteur
Prof. Y. Leblebici, rapporteur

2019

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Acknowledgements

First of all, I would like to thank my thesis advisor, Prof. Andreas Burg, for giving me the opportunity to join his laboratory and for his continuous support and guidance throughout my doctoral studies. I am very grateful for every technical discussion we had, which always pushed me to look at research from different perspectives, helping me to continuously discover new aspects in digital VLSI design. I also would like to thank my thesis coadvisor, Dr. Adam Teman, especially for his invaluable support during the steep learning curve over the first years of my doctoral studies.

I would like to thank Prof. Christian Enz (Integrated Circuits Laboratory, EPFL) for acting as president of my PhD jury, Prof. Yusuf Leblebici (Microelectronic Systems Laboratory, EPFL), Prof. Massimo Alioto (Green IC Group, National University of Singapore) and Prof. David Bol (Electronic Circuits and Systems Group, Université Catholique de Louvain) for serving as examiners for my thesis defense.

I feel very lucky for having conducted my doctoral studies at the Telecommunications Circuits Laboratory, EPFL. There, I had the chance to meet and work with incredibly talented people who made the laboratory a very supportive and stimulating environment. For the great time we had, I would like to thank: Alexios Balatsoukas-Stimming, Andrew Austin, Christian Senning, Christoph Müller, Ester Vicario Bravo, Georgios Karakonstantis, Ioanna Paniara, Jeremy Constantin, Lorenz Schmid, Marco Widmer, Matthieu Cotting, Nicholas Preyss, Nicolas Frigerio, Orion Afisiadis, Pascal Giard, Pascal Meinerzhagen, Pavle Belanovic, Reza Ghanaatian, Ricardo Gómez Gómez, Robert Giterman, Shrikanth Ganapathy and Yann Kurzo.

Finally, I would like to thank my family: Marisa, Marco and Dirce for their continuous support and for having always believed in me.

Lausanne, January 1st 2019 Andrea Bonetti

Abstract

As an increase of intelligent and self-powered devices is forecasted for our future everyday life, the implementation of energy-autonomous devices that can wirelessly communicate data from sensors is crucial. Even though techniques such as voltage scaling proved to effectively reduce the energy consumption of digital circuits, additional energy savings are still required for a longer battery life. One of the main limitations of essentially any low-energy technique is the potential degradation of the quality of service (QoS). Thus, a thorough understanding of how circuits behave when operated around the point of first failure (PoFF) is key for the effective application of conventional energy-efficient methods as well as for the development of future low-energy techniques. In this thesis, a variety of circuits, techniques, and tools is described to reduce the energy consumption in digital systems when operated either in the safe and conservative exact region, close to the PoFF, or even inside the inexact region.

A straightforward approach to reduce the power consumed by clock distribution while safely operating in the exact region is dual-edge-triggered (DET) clocking. However, the DET approach is rarely taken, primarily due to the perceived complexity of its integration. In this thesis, a fully automated design flow is introduced for applying DET clocking to a conventional single-edge-triggered (SET) design. In addition, the first static true-single-phase-clock DET flip-flop (DET-FF) that completely avoids clock-overlap hazards of DET registers is proposed.

Even though the correct timing of synchronous circuits is ensured in worst-case conditions, the critical path might not always be excited. Thus, dynamic clock adjustment (DCA) has been proposed to trim any available dynamic timing margin by changing the operating clock frequency at runtime. This thesis describes a dynamically-adjustable clock generator (DCG) capable of modifying the period of the produced clock signal on a cycle-by-cycle basis that enables the DCA technique. In addition, a timing-monitoring sequential (TMS) that detects input transitions on either one of the clock phases to enable the selection of the best timing-monitoring strategy at runtime is proposed.

Energy-quality scaling techniques aim at trading lower energy consumption for a small degradation on the QoS whenever approximations can be tolerated. In this thesis, a low-power methodology for the perturbation of baseline coefficients in reconfigurable finite impulse response (FIR) filters is proposed. The baseline coefficients are optimized to reduce the switching activity of the multipliers in the FIR filter, enabling the possibility of scaling the power consumption of the filter at runtime.

The area as well as the leakage power of many system-on-chips is often dominated by embedded memories. Gain-cell embedded DRAM (GC-eDRAM) is a compact, low-power and

CMOS-compatible alternative to the conventional static random-access memory (SRAM) when a higher memory density is desired. However, due to GC-eDRAMs relying on many interdependent variables, the adaptation of existing memories and the design of future GC-eDRAMs prove to be highly complex tasks. Thus, the first modeling tool that estimates timing, memory availability, bandwidth, and area of GC-eDRAMs for a fast exploration of their design space is proposed in this thesis.

Key Words: Digital VLSI Circuits, Low-Power Design, Nanometer Nodes, Clock Distribution, Dual-Edge-Triggered Clocking, Dual-Edge-Triggered Registers, Timing Monitoring, Dynamic-Timing Margins, Dynamic Clock Adjustment, Multipliers, FIR Filters, Approximate Computing, Gain-Cell Embedded DRAMs, Memory Design, Architecture Optimization.

Sommario

Dato che un incremento di dispositivi intelligenti e autoalimentati è previsto in futuro, la progettazione di dispositivi energeticamente autonomi che telecomunichino i dati raccolti da sensori è cruciale. Sebbene tecniche come l'abbassamento della tensione di alimentazione riducano efficientemente l'energia consumata dai circuiti digitali, ulteriori risparmi energetici sono richiesti per aumentare la durata della batteria. Uno dei limiti di essenzialmente ogni tecnica a basso consumo di energia è la potenziale degradazione della qualità del servizio. Perciò, la comprensione di come i circuiti si comportino quando operati attorno al primo punto di malfunzionamento è essenziale per ridurre il consumo energetico. In questa tesi, circuiti, tecniche e strumenti sono descritti per ridurre l'energia consumata in sistemi digitali quando operati nella conservativa regione libera da imprecisioni, vicino al primo punto di malfunzionamento o in presenza di errori.

Considerando la regiona libera da imprecisioni, un approccio ben conosciuto per ridurre la potenza consumata nella distribuzione del segnale di clock è l'uso del dual-edge-triggered (DET) clocking. Tuttavia, l'approccio DET è raramente scelto, primariamente a causa della percepita complessità della sua integrazione. In questa tesi, un flusso di progettazione automatico è introdotto per l'applicazione del DET clocking ad un circuito inizialmente progettato con il convenzionale single-edge-triggered (SET) clocking. In aggiunta, il primo flip-flop DET statico con true-single-phase-clock che rimuove il rischio di clock overlap presente nei convenzionali registri DET è proposto in questa tesi.

Sebbene il corretto timing di circuiti sincroni è verificato nelle peggiori condizioni, il percorso critico può non sempre essere attivato. Per questo motivo, la regolazione dinamica del segnale di clock può rimuovere la presenza di qualsiasi margine dinamico di timing con la modifica della frequenza di clock durante il tempo di esecuzione. Per abilitare questa tecnica, un generatore di clock capace di modificare la frequenza del segnale prodotto ad ogni ciclo è descritto in questa tesi. In aggiunta, un flip-flop capace di avvertire transizioni di segnale al suo ingresso durante una delle due fasi di clock è proposto.

Le tecniche di computazione approssimale puntano a ridurre l'energia consumata al costo di una degradazione della qualitá del servizio qualora queste approssimazioni possano essere tollerate. In questa tesi, la perturbazione dei coefficienti di base di un filtro finite impulse response (FIR) programmabile è usata per ridurre la potenza consumata dal filtro. I coefficienti di base sono ottimizzati per ridurre l'attività di commutazione dei moltiplicatori del filtro FIR, permettendo la possibilità di ridurre la potenza consumata del filtro durante il tempo di esecuzione.

Sommario

L’area così come la potenza di leakage di diversi system-on-chip è spesso dominata dalle memorie integrate. Le gain-cell embedded DRAM (GC-eDRAM) sono valide alternative alle convenzionali static random-access memory (SRAM) grazie alla loro compattezza, basso consumo di potenza e compatibilità con la tecnologia CMOS qualora un’alta densità di memoria è desiderata. Tuttavia, l’integrazione di GC-eDRAM già esistenti e la progettazione di GC-eDRAM future sono processi complessi a causa della dipendenza di queste memorie su numerose variabili interdipendenti. In questa tesi, il primo modello di timing, disponibilità, banda e area delle GC-eDRAM è proposto per una rapida esplorazione del loro spazio di progetto.

Parole Chiave: Circuiti Digitali, Progettazione a Basso Consumo di Potenza, Tecnologie Nanometriche, Distribuzione del Segnale di Clock. Dual-Edge-Triggered Clocking, Timing Monitoring, Margini Margini Dinamici di Timing, Regolazione Dinamica del Segnale di Clock, Moltiplicatori, Filtri FIR, Computazione Approssimale, Gain-Cell Embedded DRAMs, Progettazione di Memorie, Ottimizzazione delle Architetture.

Contents

Acknowledgements **i**

Abstract (English/Italiano) **iii**

1 Introduction **1**

1.1 Contributions 3
1.2 Thesis Outline 7
1.3 Selected Publications 8
1.4 Third-Party Contributions 9

2 Dual-Edge-Triggered Clocking for Low-Power Operation **11**

2.1 Automated Integration of Dual-Edge Triggered Clocking 13
  2.1.1 Motivation and Tradeoffs in DET Clocking 14
  2.1.2 Design and Implementation of DET Library Cells 17
  2.1.3 Proposed DET Digital Design Flow 20
    2.1.3.1 Initial SET Synthesis 20
    2.1.3.2 Post-Synthesis Conversion to DET 21
    2.1.3.3 DET Automatic Place and Route Flow 22
    2.1.3.4 Methodology for Mixed SET/DET Integration 23
  2.1.4 Implementation and Results 24
    2.1.4.1 Test Cases and Implementation 25
    2.1.4.2 Power Savings 28
    2.1.4.3 Area Overhead 30
  2.1.5 Conclusion 31
2.2 An Overlap-Contention Free True-Single-Phase Clock DET Flip-Flop 32
  2.2.1 Clock-Overlap Failure Risk in DET-TGLM Cells 32
    2.2.1.1 Overview of the DET-TGLM 33
    2.2.1.2 Clock-Overlap Failure Risk 34
  2.2.2 Proposed SDET-TSPCFF 36
  2.2.3 Simulations and Results 38
  2.2.4 Conclusion 40

3 Circuits and Techniques to Monitor and Trim Dynamic Timing Margins 41
3.1 A Clock Generator for Cycle-by-Cycle Dynamic Clock Adjustment 43
3.1.1 Concept of Dynamic Clock Adjustment 44
3.1.2 Dynamically-Adjustable Clock Generator 45
3.1.3 Test Chip and Measurements Results 46
3.1.3.1 Characterization of the DCG 47
3.1.3.2 Application of the DCG to a DCA-Enabled Microprocessor . . . 48
3.1.4 Conclusion 49
3.2 A Timing-Monitor Sequential for Forward and Backward Error-Detection 50
3.2.1 Proposed Timing-Monitoring Sequential 52
3.2.2 Timing Analyses Enabled by the TMS 54
3.2.3 Simulations Results and Measurements 55
3.2.3.1 Post-Layout Simulations of the TMS 55
3.2.3.2 Measurements of a Multiplier Using TMS Cells 57
3.2.4 Conclusion 58

4 Exploiting Hardware Properties at the Algorithm Level for Energy-Quality Scaling 59
4.1 Digital Multipliers 62
4.1.1 Radix-2 Bough-Wooley Multiplier 64
4.1.2 Radix-4 Booth-Recoded Multiplier 64
4.2 Voltage Over-Scaling 65
4.2.1 Overclocking a Multiplier 65
4.2.2 Limitations of Voltage Over-Scaling 68
4.3 Switching Activity in Multipliers 69
4.3.1 Radix-2 Bough-Wooley Multiplier 69
4.3.2 Radix-4 Booth-Recoded Multiplier 71
4.4 Gate-Level Characterization of Multipliers 71
4.4.1 Power Characterization 73
4.4.2 Timing Analysis 77
4.5 Application to an FIR Filter 78
4.5.1 Optimization Algorithm 80
4.5.2 Simulations Results 81
4.5.3 Test Chip and Measurements Results 84
4.6 Conclusion 87

5 Gain-Cell Embedded DRAMs: Modeling and Design-Space Exploration 89
5.1 Related Work 91
5.2 Background on GC-eDRAM 92
5.2.1 Gain Cell 93
5.2.2 Memory Organization in GC-eDRAMs 95
5.2.3 Architectural Transformations in GC-eDRAMs 97
5.2.4 Features of GC-eDRAMs 97
5.2.5 Computing with GC-eDRAMs: Application to a 32-bit Microprocessor . 98

5.2.5.1 System Architecture 98
5.2.5.2 GC-eDRAM Subsystem 99
5.3 GEMTOO: GC-eDRAM Modeling Tool 101
5.3.1 Timing 103
5.3.2 Memory Availability and Bandwidth 105
5.3.3 Area and Utilization 107
5.4 Validation 107
5.5 Design-Space Exploration 109
5.5.1 Optimization for Operating Frequency 109
5.5.2 Optimization for Memory Availability and Bandwidth 111
5.5.3 Optimization for Memory Density 116
5.6 Conclusion 117

6 Conclusions and Outlook 119

A Power Tradeoffs in Dual-Edge-Triggered Clocking 125

Bibliography 138

Acronyms 141

List of figures 145

List of tables 149

Curriculum Vitae 151

List of Publications 153

1 Introduction

Energy efficiency is one of the main drivers for the rapid proliferation of portable and battery-powered devices. The increasing interest in embedded applications, such as the Internet of Things (IoT), is expected to boost this growth even further as the production of a trillion new IoT nodes is forecasted between now and 2035, according to ARM [1]. Beside the constant need for long-battery life, the design requirements on IoT nodes are driven by an increasing demand for flexibility to support different applications and intelligence to perform complex tasks. Thus, improvements in energy efficiency without any limitation on the performance are crucial for a widespread use of smart IoT nodes.

IoT nodes (i.e., the “things”) are composed of many modules, as shown in Fig. 1.1. The primary tasks of these end nodes are the collection of data from sensors, the embedded processing of the retrieved data, and the wireless communication with the network. As IoT nodes should ideally operate without battery replacement, the application of effective low-energy techniques is mandatory to ensure the full functionality of the devices for several months or even for years. In this context, the processing modules play a key role as the collected raw data is filtered by either the microcontroller or by dedicated accelerators to limit the amount of data that needs to be stored and wirelessly transmitted. Thus, size reduction of the processed data can relax the requirements on the storage and the implementation of smaller memories enables lower standby power. Furthermore, the wireless transmission of a smaller amount of data allows to use the radio for a shorter period of time, therefore saving a large amount of active power. Hence, the addition of embedded processing modules to extract only the essence from the collected data is key for the development of energy-efficient devices. This approach is generally referred as *edge computing*.

Given the relevance of the computing tasks that are required to be performed by IoT nodes, several processing modules are often implemented within battery-limited devices. As an analog-to-digital converter (ADC) is used to translate the collected data into a digital signal, digital filters are often required to eliminate the signal components that are out of the frequency band of interest or to perform matched filtering. The need for flexible IoT nodes that can support different applications is fulfilled by a programmable and general-purpose module,

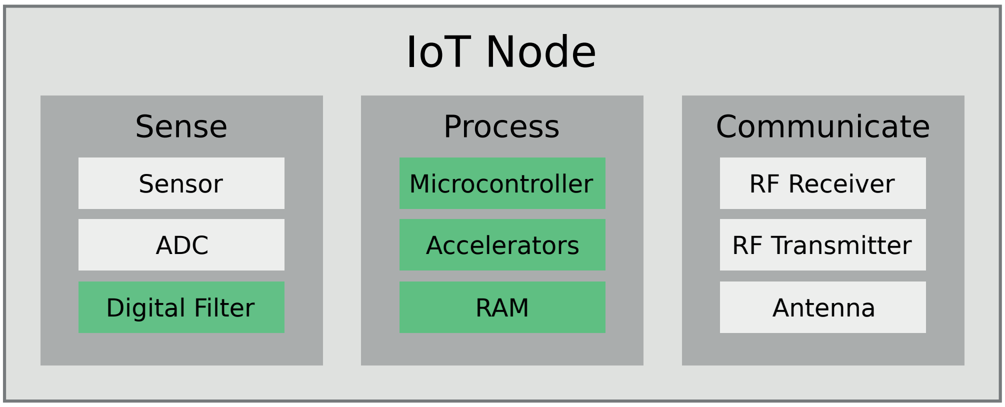


Figure 1.1: Modules embedded in a node for the Internet of Things (IoT). The modules highlighted in green are targeted for energy efficiency by the circuits, techniques, and tools proposed in this thesis.

such as a microcontroller. Furthermore, dedicated accelerators are often implemented due to the increasing demand for complex and specialized tasks, such as encryption and feature extraction. Beside the presence of modules for signal processing, the filtered data is collected or stored in a random-access memory (RAM), during processing or simply waiting for being transmitted.

Due to the need for embedded filtering and feature-extraction tasks, edge computing can grow the number and the complexity of the implemented processing modules. Thus, circuits and techniques that aim at reducing the energy consumption of this large variety of integrated modules are crucial for the design of flexible and intelligent IoT nodes that have to meet the tight energy budget. In this context, dynamic voltage and frequency scaling (DVFS) [2] is one of the most effective technique to reduce the dynamic power consumption in digital circuits due to the quadratic dependence of the supply voltage. As conventional six-transistor (6T) static random-access memory (SRAM) is typically the first module to fail when operated at low voltages, alternative SRAM bitcells have been proposed [3, 4] to operate static embedded memories even in the near-threshold regime. Costly guard bands are typically added on top of the nominal supply voltage to ensure always-correct operation under any operating condition. However, the system can often be operated reliably without the need of these large and conservative design margins as some of the dynamic variations, such as temperature and aging, have rather long time constants. Thus, many error-detection sequentials (EDSs) have been proposed [5–10] to monitor any change on the critical paths of the design to adaptively trim the guard bands whenever they are not needed, therefore maximizing the power savings. For additional power savings when real-time constraints limit the available safe range for voltage scaling, voltage over-scaling (VOS) [11–14] has been proposed for operating below the critical supply voltage and applying various techniques to handle the timing errors that may occur. Even these conventional techniques already proved to effectively reduce the energy consumption of digital circuits, additional energy savings are still required due to the increasing demand for long-lasting IoT devices and the physical limitations of the traditional low-energy techniques (e.g., nanometer circuits are highly impacted by process variations when operated at scaled voltages). Given this context, supplementary circuits and techniques should be explored to ideally eliminate any excess of energy consumption.

The application of essentially any low-energy technique is often limited by the potential degradation of the quality of service (QoS). For example, an excessive application of voltage scaling can lead to a severe failure of the system due to timing violations. Thus, a thorough understanding of how circuits behave when operated *around* the point of first failure (PoFF), which traditionally identifies where an entire system stops working properly, is key for the effective application of conventional energy-efficient methods as well as for the development of future low-energy techniques. In this regard, four main regions can actually be identified:

- *Exact region:* the vast majority of the energy-efficient techniques reported in the literature target digital systems that are supposed to operate in the safe and conservative exact region. Further improvements in this field of research are constantly needed to reduce the energy consumption of functionally-critical components, such as controllers.
- *Near-PoFF region:* the dynamic adaptation to the operating conditions of a system is key for the runtime elimination of potentially any excess of energy consumption. However, this approach often requires the system to operate *close* to the PoFF, therefore increasing the risk of potential errors (e.g., timing violations). In this context, the implementation of error detection and correction (EDAC) mechanisms is generally required to avoid any penalty on the QoS whenever an error occurs.
- *Inexact region:* abandoning the error-free paradigm (i.e, crossing the PoFF boundary) can eliminate some of the limitations of conventional energy-efficient techniques and allow for additional energy savings whenever a small degradation on the QoS can be tolerated. In this regard, a deep understanding and modeling of the inexact circuit operation as well as of potential reliability issues is key to maximize the energy benefits for a limited and graceful degradation on the QoS.
- *Failure region:* the presence of an overwhelming amount of errors is unacceptable as the QoS might be excessively degraded or the the entire system might stop from working. Thus, systems cannot be operated within this region.

1.1 Contributions

The contributions of this thesis focus on circuits, techniques, and tools for the design of energy-efficient digital very-large-scale integration (VLSI) modules when operated in either the exact, near-PoFF, or inexact region, as summarized in Fig. 1.2. A large variety of circuits are optimized for low-energy consumption, such as the clock-distribution network, sequentials, signal-processing modules for the data path, and embedded memories among others. In

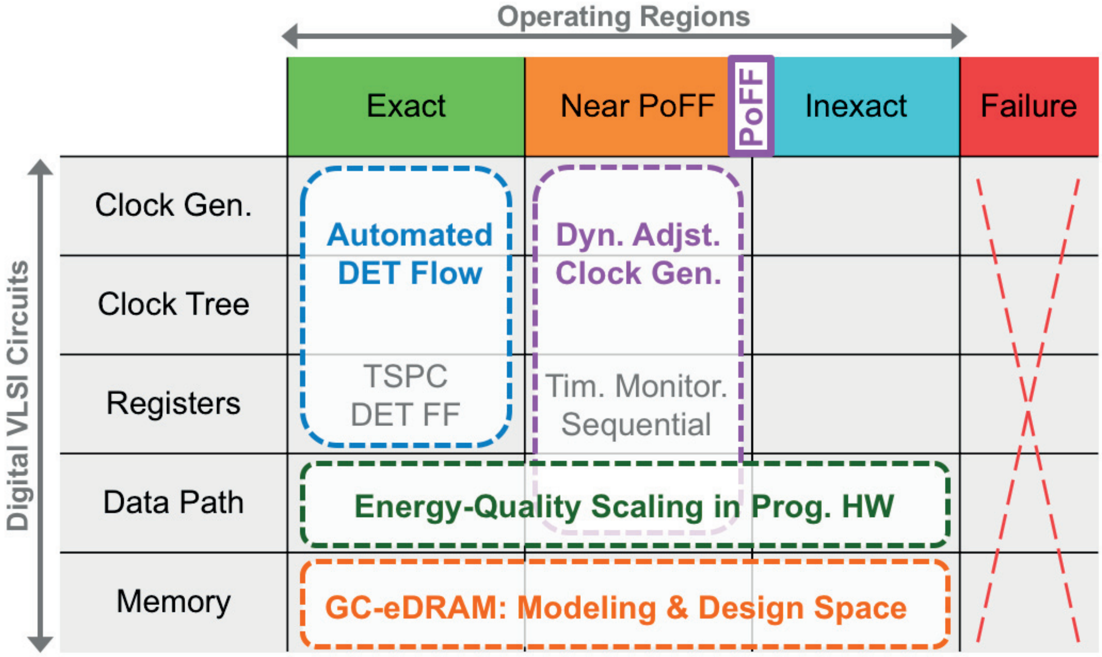


Figure 1.2: Contributions of this thesis classified by targeted operating region and addressed circuit topology.

particular, the energy savings on the distribution of the clock signal due to the adoption of dual-edge-triggered (DET) clocking are enabled by the proposed automated flow that converts a digital block initially designed for single-edge-triggered (SET) operation into a fully DET component. A static true-single-phase-clock (TSPC) DET flip-flop is also proposed to solve race conditions when using both clock phases. Considering the application of DVFS for energy efficiency, the exploitation of dynamic-timing margins is enabled by the described dynamically-adjustable clock generator (DCG) that is capable of immediate and glitch-free changes on the frequency of the generated clock signal to apply the dynamic clock adjustment (DCA) technique. Furthermore, a timing-monitoring sequential (TMS) is proposed for advanced DVFS techniques where transitions on the input data can be detected on either one of the clock phases to select the best timing-monitoring strategy. Focusing on the data path, properties of programmable hardware are used to enable an energy-quality scaling technique on the algorithmic level. In this regard, the power consumption of the multipliers contained in a programmable finite impulse response (FIR) accelerator is reduced by the runtime adaptation of the filter coefficients, to consume less energy whenever a small degradation on the QoS is tolerated and provide the baseline performance whenever needed. Concerning embedded memories, this thesis proposes the first modeling tool for gain-cell embedded DRAM (GC-eDRAM), which proved to be a compact and low-leakage alternative to conventional SRAM. The described tool enables a practical exploration of the vast and complex design space to scale existing GC-eDRAMs or design future memories through the estimation of their maximum operating frequency, availability, bandwidth, and memory density. A more detailed description of the contributions of this thesis is provided as follows.

Dual-Edge-Triggered Clocking for Low-Power Operation

The adoption of a DET clocking scheme in synchronous designs is an effective approach for the reduction of the energy consumed in clock distribution when the system is operated inside the safe and conservative exact region. In this thesis, power trade-offs of DET clocking are initially analyzed, providing the means to rapidly identify the characteristics of systems that have the highest potential to benefit from DET clocking for low-power operation. Furthermore, a design flow for the fully automatic implementation of a DET clocking scheme within the standard digital design flow is proposed. The described methodology can be applied to any digital component and it does not require any overhead in the logic-design process. The proposed design flow enables seamless transformation of a digital block, initially designed for SET operation, into a fully DET component. The systematic and full integration of DET clock gating in the digital standard design flow is presented, to the best of the author's knowledge, for the first time. DET operation is applied to three digital components, implemented in a commercial 40 nm CMOS process technology, showing the power-performance benefits of this approach on different real-life designs with post-layout simulations.

In order to solve race conditions that arise when using both clock phases and that might lead to a loss of data due to overwriting in conventional DET flip-flops, the first static TSPC DET flip-flop is proposed in this thesis. By implementing the cell with TSPC circuits and an internal dual-feedback mechanism, completely static and robust operation is achieved under voltage scaling and process variations. To demonstrate the robustness of the proposed DET flip-flop in nanometer technologies, the register was implemented in a 40 nm CMOS process technology, showing full functionality at a near-threshold supply voltage of 0.5 V and under extensive Monte Carlo statistical simulations for both global and local variations. In addition, the proposed flip-flop provides the lowest CK-to-Q delay and the best power-delay product when compared to other leading DET flip-flops.

Circuits and Techniques to Monitor and Trim Dynamic Timing Margins

According to DCA techniques, the exploitation of dynamic-timing margins for the operation of the system inside the near-PoFF region is enabled by the DCG proposed in this thesis. The DCG is a digitally-controlled ring oscillator (DCRO) that produces a clock signal whose period can be changed at every clock cycle, as required by techniques relying on DCA. To modify the propagation delay inside the ring, the oscillator includes a programmable delay unit implemented with a cascade of AND gates that have been placed with a controlled floorplan to produce an accurate range of clock periods. The cycle-by-cycle operation is ensured by using a one-hot encoded period setting for the programmable delay unit and by sampling this delay setting with a set of additional registers that have been placed close to the programmable delay unit to minimize both their propagation delays as well as the delay of the clock signal they receive from the ring oscillator. Silicon measurements of a 28 nm FD-SOI test chip are provided to characterize the proposed clock generator and demonstrate the exploitation of dynamic-timing margins in a DCA-enabled embedded processor that uses the described DCG as clock source.

A TMS capable of detecting transitions on the input data during either the high or the low phase of the clock to enable and select the best timing-monitoring strategy is proposed. The use of the TMS together with the control of the clock duty-cycle enables three different timing analyses: the conventional monitoring of any setup violation, the measurement of the available and positive timing slack that can be performed either during a safe calibration phase or even at runtime, and the measurement of fast paths (i.e., paths that are far from being setup-timing critical) to evaluate the potential benefits in the exploitation of dynamic-timing margins provided by the DCA approach. The TMS is implemented on a 28 nm FD-SOI test chip and verified with silicon measurements.

Exploiting Hardware Properties at the Algorithm Level for Energy-Quality Scaling

Even though significant energy savings can be achieved through VOS several factors limit the application of such an aggressive energy-efficient technique when operating inside the inexact region: a steep degradation in the QoS is often experienced as soon as the first timing errors appear and the degradation of the QoS can be hardly predicted or controlled. To avoid the use of such risky strategies, in this thesis, energy-quality scaling is enabled by an algorithmic-level technique that reduces the switching activity of multipliers by carefully choosing the programmable parameters of an FIR accelerator. First, an analysis of the switching activity of multipliers based on the number of non-zero generated partial-product bits is described. The obtained results are confirmed by the dynamic-power characterization of the considered multipliers through accurate post-layout simulations, showing that the achievable power savings might differ depending on which of the two input ports is assigned to the constant coefficient. The power consumption of the multipliers implemented in a programmable FIR filter is reduced by perturbing the baseline coefficients of the filter based on the extensive power characterization of the implemented multiplier topology. The implementation of the power-optimized FIR filter does not require any design overhead except for the additional memory that might be required to store the perturbed coefficients. For the proposed technique, the baseline performance is always ensured when operating with the reference coefficients and the power consumption of the FIR filter can be reduced at runtime, when a less accurate operation of the filter is tolerated. The described technique is applied to several FIR filters, demonstrating the obtainable power reductions. These savings are confirmed through measurements of filters fabricated on a 28 nm FD-SOI test chip.

Gain-Cell Embedded DRAMs: Modeling and Design-Space Exploration

GC-eDRAM is a high-density and low-leakage alternative to conventional SRAM and can be operated either inside the safe exact region or even within the inexact region as it gracefully degrades when the refresh rate is relaxed for energy-quality scaling. In this thesis, the first modeling tool for GC-eDRAMs is proposed for a practical exploration of their complex and vast design space and in support of their design and integration within low-power computing systems. The proposed modeling tool is based on input parameters related to technology, circuits, and memory organization of GC-eDRAMs, therefore allowing for an exploration of a large design space. The physical floorplan of the GC-eDRAM is considered in the estimation of memory metrics, therefore accounting for the impact of the memory organization as well as of the load given by the interconnects. An accurate timing estimation is provided by modeling the effect of the deterioration of the stored data in the gain cell. The tool is implemented in a modular structure to allow for the selection or the introduction of different modeling strategies and the code of the tool is open source as well as publicly available. The timing estimation of the modeling tool is validated against transistor-level simulations of different GC-eDRAMs implementations which include both resistive and capacitive parasitics from post-layout extraction as well as against silicon measurements of a previously fabricated GC-eDRAM in 28 nm CMOS process technology. Multiple case studies of design-space exploration are presented based on the proposed modeling tool to find the best design choices that fulfill the most critical memory requirements, such as highest operating frequency, availability, bandwidth, and memory density.

1.2 Thesis Outline

In Chapter 2, the motivation and tradeoffs for the application of DET clocking are first described to identify the conditions where DET clocking is effective in reducing the energy consumption. The intricacies in the design and implementation of DET flip-flops and clock gates are then analyzed and a design flow for the fully-automated implementation of DET clocking is proposed. To evaluate the benefits of DET clocking, three IP blocks are implemented in 40 nm CMOS process technology with both SET and DET clocking schemes. Post-layout power simulations are performed to quantize the energy benefits when DET clocking is applied. Within the same chapter, logic failures due to clock overlap in DET flip-flops are described and solved by the proposition of the first TSPC DET flip-flop. The proposed DET flip-flop is implemented in 40 nm CMOS process technology, verified at the near-threshold voltage of 0.5 V and compared to to the state-of-the-art solutions for DET registers.

Chapter 3 describes a DCG capable of immediate and glitch-free changes on the frequency of the generated clock signal. The proposed clock generator is a crucial module for the application of techniques that aim at trimming dynamic timing margins through DCA. Furthermore, a TMS capable of detecting transitions on the input data on either one of the clock phases to enable and select the best timing-monitoring strategy is described in the same chapter. The proposed TMS can either detect setup violations, warn when a path is *close* to become critical, or even measure fast paths to evaluate potential dynamic timing margins. Both the reported clock generator and the TMS have been fabricated on 28 nm FD-SOI test chips, verified, and characterized with silicon measurements.

In Chapter 4, the impact of VOS on the reliability of a multiplier is analyzed, showing that the application of VOS is a high-risk low-energy technique due to the unpredictable, steep, and severe degradation of the QoS. As an alternative, hardware properties of programmable FIR filters are exploited to reduce the switching activity in multipliers in a proposed energy-scaling technique applied at the algorithmic level. As multipliers account for the largest amount of power consumed in a programmable FIR filter, their dynamic power consumption is extensively characterized for different fixed input operands that coincide with the coefficients of the FIR filter. The power characterization of the multipliers is used to perturbate the coefficients of a baseline filter and obtain a set of power-optimized coefficients for a small degradation on the QoS. The proposed technique is applied and verified with silicon measurements on filter accelerators fabricated on a 28 nm FD-SOI test chip for different topologies of multipliers and filters.

Chapter 5 describes the first modeling tool for GC-eDRAMs. The tool estimates timing, memory availability, bandwidth, and area of GC-eDRAMs based input parameters related to technology, circuits, and memory organization of GC-eDRAMs. The use of the proposed tool is useful for memory design to scale existing the performance of existing memories and predict the metrics of future memories as well as for architectural optimization of systems in which the memories are implemented. The timing estimation of the described modeling tool is validated against both simulated and measured GC-eDRAMs in 28 nm CMOS process technology. The proposed modeling tool is also used to show the intricacies in design optimization of GC-eDRAMs and, based on the results, optimal design practices are derived.

Conclusions and outlook are provided in Chapter 6.

1.3 Selected Publications

This thesis is largely based on the following publications.

Dual-Edge-Triggered Clocking for Low-Power Operation

A. Bonetti, A. Teman, and A. Burg, "An Overlap-Contention Free True-Single-Phase Clock Dual-Edge- Triggered Flip-Flop", *IEEE International Symposium on Circuits & Systems (ISCAS)*, May 2015.

A. Bonetti, N. Preyss, A. Teman, and A. Burg, "Automated Integration of Dual-Edge Clocking for Low-Power Operation in Nanometer Nodes", *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, May 2017.

Circuits and Techniques to Monitor and Trim Dynamic Timing Margins

J. Constantin, A. Bonetti, A. Teman, Christoph Müller, Lorenz Schmid, and A. Burg, "DynOR: A 32-bit Microprocessor in 28nm FD-SOI with Cycle-By-Cycle Dynamic Clock Adjustment", *European Solid-State Circuits Conference (ESSCIRC)*, September 2016.

A. Bonetti, J. Constantin, A. Teman, and A. Burg, "A Timing-Monitoring Sequential for Forward and Backward Error-Detection in 28 nm FD-SOI", *IEEE International Symposium on Circuits & Systems (ISCAS)*, May 2018.

Exploiting Hardware Properties at the Algorithm Level for Energy-Quality Scaling

A. Bonetti, A. Teman, P. Flatresse, and A. Burg, "Multipliers-Driven Perturbation of Coefficients for Low-Power Operation in Reconfigurable FIR Filters". *IEEE Transactions on Circuits and Systems I: Regular Papers (TCAS-I)*, September 2017.

Gain-Cell Embedded DRAMs: Modeling and Design-Space Exploration

A. Bonetti, R. Golman, R. Giterman, A. Teman, and A. Burg, "Gain-Cell Embedded DRAMs: Modeling and Design Space", *Under revision*, 2018.

1.4 Third-Party Contributions

All third-party contributions to the work presented in this thesis are listed in this section.

I worked in close collaboration with Nicholas Preyss during the development of the automated flow for the insertion of the DET clocking scheme. Nicholas supported me in various stages of the project, providing the IP block and the testbench for the case-study A, the development of the DET automatic place-and-route flow, and the timing evaluation of all the considered IP blocks. Adam Teman oversaw the whole project providing many valuable solutions for the place-and-route flow and power simulations. Christian Senning and Reza Ghanaatian provided the IP blocks and the testbenches for the case-studies B and C, respectively.

The proposed DCG has been integrated in DynOR, a DCA-enabled embedded processor, that has been designed, fabricated, and tested as a result of a collaboration within many people: Jeremy Constantin, main frontend designer and leader of the DynOR project, Adam Teman, who supervised the backend tasks, Christoph Müller, who contributed to the backend flow, and Lorenz Schmid, who worked on chip testing.

The programmable FIR filters used for the evaluation of the proposed energy-quality scaling technique and the described timing monitoring sequentials have been integrated in PolarBear, a 28 nm FD-SOI test chip that was the result of a team effort: Lorenz Schmid integrated the accelerators, the timing monitoring sequentials, and the testing circuitry in the chip, while Christoph Müller designed the top-level control logic, lead the backend tasks and gave invaluable contributions during the power measurements of the accelerators.

Different people contributed to the design and fabrication of the described processor implementing a GC-eDRAM as data memory. Jeremy Constantin supported me in the very first design stages to integrate the implemented OpenRISC core in the system. Robert Giterman designed the custom arrays of the GC-eDRAM and provided support for the backend tasks. Christoph Müller designed the implemented body-bias voltage generators as well as their control logic. Ivan Miro-Panades provided the frequency-locked loop (FLL) used for internal clock generation.

2 Dual-Edge-Triggered Clocking for Low-Power Operation

Power reduction in integrated circuits (ICs) continues to be one of the primary objectives in the field of digital system design, especially in light of the increasing throughput required by modern systems [15]. The majority of these systems are fully or primarily synchronous, requiring the distribution of one or more clock signals across the entire chip. The necessary clock networks typically drive a large capacitive load and they are always toggling when clock gating is not active. Thus, a significant amount of power is consumed for clock distribution and can account for 30%–60% of the total chip power [16–20].

A well-known technique for reducing the dynamic power dissipation of a synchronous IC is dual-edge-triggered (DET) clocking. As opposed to the conventional single-edge-triggered (SET) solution, which samples the data only on the rising-edge of the clock, DET operation uses both the rising and the falling edges for data sampling, thereby requiring only half of the clock frequency of the SET approach for the same throughput. As shown analytically by Nedovic *et al.* [17, 18], the resulting dynamic power reduction in clock distribution can exceed 50%, and DET clocking is always more power-efficient if the clock load capacitance of the DET storage elements is less than twice as large as that of their SET counterparts. In addition, since DET flip-flops (DET-FFs) have been shown to be more energy-efficient than their SET counterparts for high-speed applications [21–23] and have comparable or even *lower* propagation delay [17, 21–23], power reduction can often be achieved with a similar or even a slightly higher throughput than with conventional SET clocking.

In addition to the power advantages of DET clocking, the reduced clock frequency also relaxes many of the issues that are introduced by high-speed architectures. The effect of the electromagnetic interference (EMI) typically produced by high-frequency sources is attenuated, thereby reducing noise coupling. Since the transient activity on the current drawn from the supply is reduced, the noise on the power supply voltage caused by the presence of an equivalent impedance is also reduced. DET clocking also eases many of the challenges and constraints on producing a clock signal that is twice the speed of the data, as the maximum toggle frequency of the clock is identical to the maximum toggle frequency of the data. Finally, by enabling DET blocks within a primarily SET system, a high performance block can be implemented with DET standard cells and be operated at twice the speed of the system without the need for generating and distributing an additional faster clock.

The implementation of storage cells that are triggered on both clock edges is a well-researched topic [19, 24–28]. Several works have investigated and compared the performance of DET-FFs with that of conventional SET flip-flops (SET-FFs) [18, 21–23, 29, 30], showing that DET-FFs outperform SET storage cells in both low-power and high-speed applications [21–23]. The main drawbacks of DET-FF implementations are that they occupy a larger silicon area and consume higher leakage currents than SET flip-flops, especially in deep nanometer nodes, making DET operation less appropriate for systems or components characterized by long sleep periods or with frequent clock-gating [18]. For this reason, these tradeoffs have to be carefully analyzed at the system level in order to choose the most power-efficient clocking scheme. Architectures in which the clock network and the registers are active most of the time and consume a large portion of the total power are the most promising candidates for DET clocking. Power-efficient design of the implemented DET-FFs will further contribute to the power savings in register-heavy systems. For example, many signal processing applications, such as cryptography, digital filtering, neural networks, and communication systems are often realized with register-based architectures featuring deep pipeline structures, rendering them perfect candidates for DET implementation, as long as the area penalty can be tolerated.

In this chapter, circuits and an automated design flow are proposed to enable the implementation of DET clocking:

- In Section 2.1, the most promising conditions for achieving low-power operation with DET clocking are identified and a fully-automated design flow for applying DET to a conventional SET design is presented.
- Section 2.2 describes the clock-overlap failure risk in DET registers and proposes a DET flip-flop with a true-single-phase clock that completely avoids clock overlap hazards by eliminating the need for an inverted clock edge for functionality.

2.1 Automated Integration of Dual-Edge Triggered Clocking

Even if many applications are promising candidates for DET clocking, the design of DET systems and their implementation within the digital standard design flow are considered to be cumbersome and are almost entirely neglected in the literature. In fact, only very few digital architectures that incorporate DET clocking have been reported in the literature (e.g., [31]). The emphasis of the majority of all previous publications discussing DET clocking has been on the design and implementation of stand-alone DET-FFs, neglecting the intricacies of the electronic design automation (EDA) for DET-based system integration. In addition, most standard cell libraries are equipped only with SET storage elements and clock-gating circuits and the definition of DET constraints for static timing analysis (STA) and clock-tree synthesis (CTS), especially with clock-gating, is non-trivial. Therefore, DET is often relegated to a minor role in synchronous design, used only for specific applications and in niche products.

In this section, the main practical concerns are reconsidered and addressed to make DET clocking a viable and easy-to-use technique for low-power operation of synchronous digital circuits. To that end, the most promising conditions for large power savings through DET clocking are identified and subsequently a seamless and fully-automated design flow for the integration of DET clocking into a digital IC is presented. The proposed design flow transforms a synchronous block, initially designed for SET clocking, into a fully DET implementation, thereby enabling a rapid and accurate comparison between the two implementations for each block. In this way, the direct tradeoff between the power reduction of DET clocking and its unavoidable area and leakage overheads can be considered at the block level. Thereafter, the system designer can select the best clocking scheme for each component to provide an energy-efficient full system solution, while maintaining the initial SET throughput. This provides the basis for a design methodology which integrates DET and SET clocked components within a single system.

The proposed approach is implemented exclusively with commercial EDA tools and it is applied to three representative digital blocks in a standard 40 nm CMOS process technology. The DET approach was shown to halve the power consumption in the clock tree for all test cases and to significantly reduce the dynamic power of the registers, leading to a 58% total power reduction for one of the benchmark circuits whose power is dominated by registers and clock buffers. All DET implementations are shown to retain similar throughput as compared to their SET counterparts across standard operating corners, showing that the two clocking schemes can be freely interchanged within a given specification.

Contributions The primary contributions of this section can be summarized as follows:

- Power tradeoffs in DET clocking are analyzed, providing the means to rapidly identify the characteristics of systems that have the highest potential to benefit from DET clocking for low-power operation.

- A design flow for the fully-automatic implementation of a DET clocking scheme within the standard digital design flow is proposed. The presented methodology can be applied to any digital component and it does not require any overhead in the logic-design process.

- The proposed design flow enables a seamless transformation of a digital block, initially designed for SET operation, into a fully DET component. This can be used either to benefit from the power savings of the DET approach or for a straightforward and accurate comparison between the two clocking approaches applied to a given block.

- To the best of the author's knowledge, this is also the first work presenting a systematic and full integration of DET clock gating in the digital standard design flow.

- DET operation is applied to three digital accelerators, implemented in a commercial 40 nm CMOS process technology, showing the power benefits of this approach on different real-life designs with post-layout simulations.

The rest of this section is as follows: the motivation for DET clocking and the tradeoffs between applying SET and DET clocking are discussed in Section 2.1.1; a brief overview of the DET-FF and the DET clock gate (DET-CG) used in the proposed benchmark implementations is given in Section 2.1.2, before introducing the proposed DET design flow in Section 2.1.3; Section 2.1.4 presents the application of the proposed flow on three example designs and the resulting power savings; Section 2.1.5 concludes this section.

2.1.1 Motivation and Tradeoffs in DET Clocking

A standard clock-distribution network is composed of digital gates, synchronous registers, and wires that connect them. The dynamic power consumption P^{dyn} of each of the cells in such a network can be described as:

$$P^{\mathrm{dyn}} = f \cdot K \cdot E^{\mathrm{dyn}}, \tag{2.1}$$

where f is the toggling (clock) frequency, K is the activity factor of the gate (defined as the number of transitions per clock cycle) and E^{dyn} is the energy consumed per transition by the gate. This leads to the straightforward concept of power savings through DET clocking: a reduction of the clock frequency by 50% by using both clock edges for state transitions into sequential elements can cut dynamic power consumption due to activity on the clock net in half. Unfortunately, this simple conclusion neglects several issues which limit the efficiency of the DET approach. First of all, while the power reduction of components on the clock network can be substantial, this may only be a small fraction of the total power for some systems, especially those with a small ratio of registers to combinatorial logic or systems with high clock-gating efficiency[1]. Second, the implementation of DET clocking requires the replacement of SET registers with their DET counterparts, and while these gates can be designed to be more energy-efficient, they typically present higher area and leakage power, which may not be acceptable, especially in nanoscaled processes. These factors must be taken into account when deciding upon the clocking scheme of a system. This section evaluates both the power benefits and tradeoffs of applying DET clocking to a given design to provide the system architect with a basis for making this decision.

An analytical evaluation of the power tradeoffs in DET clocking is described in Appendix A. Considering the result of (A.7), it is possible to state that DET clocking is more power-efficient than the conventional clocking scheme when the dynamic power savings (mainly obtained on the clock tree and registers) are larger than the increase in leakage power. In particular, (A.7) depends on different parameters that are analyzed as follows:

- DET clocking can reduce the dynamic power of both registers and clock tree. Therefore DET clocking will be most efficient, when applied to an SET design with a large portion of dynamic energy consumed in the clock tree ($E_{\text{tre}}^{\text{dyn}}$) and registers ($E_{\text{reg}}^{\text{int}}$ and $E_{\text{reg}}^{\text{sw}}$), as well as relatively low leakage power.
- The dynamic power savings depend on the activity factors K_{tre} and K_{reg}, meaning that designs that spend a considerable percentage of time in clock-gated sleep states are generally not recommended for DET clocking.
- Both α_{reg} and β_{reg} depend on the power comparison between the implemented SET-FFs and the DET-FFs and their values can be obtained and improved with analog simulations and various circuit-level design techniques [22, 23]. For a favourable energy-efficient DET clocking, these scaling factors need to be kept as small as possible in order to minimize the leakage power overhead and to save dynamic power in the registers.
- The register scaling factors, α_{reg} and β_{reg}, solely depend on the design of the implemented DET-FFs as compared to the SET-FFs. Reduction of these factors is fundamental for overall power savings, especially when considering register-dominated designs. In general, α_{reg} is often expected to be larger than one due to the higher leakage current that characterizes the DET-FFs, while β_{reg} (i.e., the internal energy ratio of DET vs. SET registers for a single transition) is required to be smaller than one to efficiently save power at high operating frequencies. The requirement on β_{reg} is easily met by many DET-FF topologies, primarily due to the internal power consumption of SET-FFs on the non-sampling edge of the clock – a phenomenon that is non-existent in DET-FFs. For

---

[1] In this context, clock-gating efficiency (G_{eff}) is defined as the number of registers that can be clock-gated (R_{cg}) multiplied by the average number of clock cycles where a register is clock-gated ($\overline{C_{\text{cg}}}$), divided by the product of the total number of registers (R_{tot}) times the total number of clock cycles (C_{tot}):

$$G_{\text{eff}} = \frac{R_{\text{cg}} \cdot \overline{C_{\text{cg}}}}{R_{\text{tot}} \cdot C_{\text{tot}}}.$$

the registers considered in this work, these scaling factors have been estimated considering the average performance of the implemented SET and DET flip-flops, resulting in α_{reg} and β_{reg} being equal to 1.50 and 0.40, respectively.

- The clock tree that is built in the SET design is likely to be very similar to the DET clock tree, since the buffers implemented in the clock tree generally have balanced input-to-output delays to avoid a degradation of the duty cycle in the clock propagation and under the assumption that the load on the clock pin is kept the same for both SET and DET registers. The latter condition can be easily satisfied by having the clock pin of the SET-FFs and of their DET counterparts connected to the same digital gate (i.e., a clock buffer) and ensuring the correct clock distribution inside the standard cell during the transistor-level design phase of the DET-FF. In addition to this, dynamic power savings will still be achieved, even in the case of a more complex DET clock tree, due to the operation of the DET implementation at half of the SET clock frequency. For this reason, both Ω_{tre} and Φ_{tre} can be assumed to be equal to one, in a first-order approximation.

- The conversion of an SET design to a DET implementation, as described in Section 2.1.3, is achieved by replacing SET clock gates and flip-flops with their DET counterparts without any inherent modification of the combinatorial logic. Thus, the power consumption of the logic gates is ideally unchanged and the scaling factors Ω_{log} and Φ_{log} can be approximated to one. However, the implementation of faster DET-FFs might result in the need for additional hold buffers, leading to scaling factors that are slightly larger than one. Nevertheless, the overall power consumption can still be efficiently reduced by DET clocking when this power overhead is limited and assuming that the additional slack from faster DET-FFs can be exploited in other parts of the logic to relax timing constraints.

- The choice on the most power-efficient clocking scheme depends also on the operating frequency. DET clocking is more energy-efficient than SET clocking at higher frequencies, where dynamic power dominates, but is less suited to slow designs, where a large portion of the total power consumption is due to leakage currents.

- In general, the area overhead of applying DET should be taken into account, in addition to the potential power savings. However, for many systems a slight area penalty can be tolerated, especially for pad-limited ICs.

This analysis shows that while not all digital blocks are good candidates for DET clocking, systems and components with certain characteristics will significantly benefit from this approach. However, as previously mentioned, two primary factors still prevent widespread application of DET clocking in digital systems. The first is the lack of DET-FFs and DET clock gates in standard libraries, and the second is the lack of a methodology and guidelines for the integration of DET clocking in the digital standard design flow. Therefore, the next section presents the registers and clock-gating circuits used in this work, followed by a detailed methodology

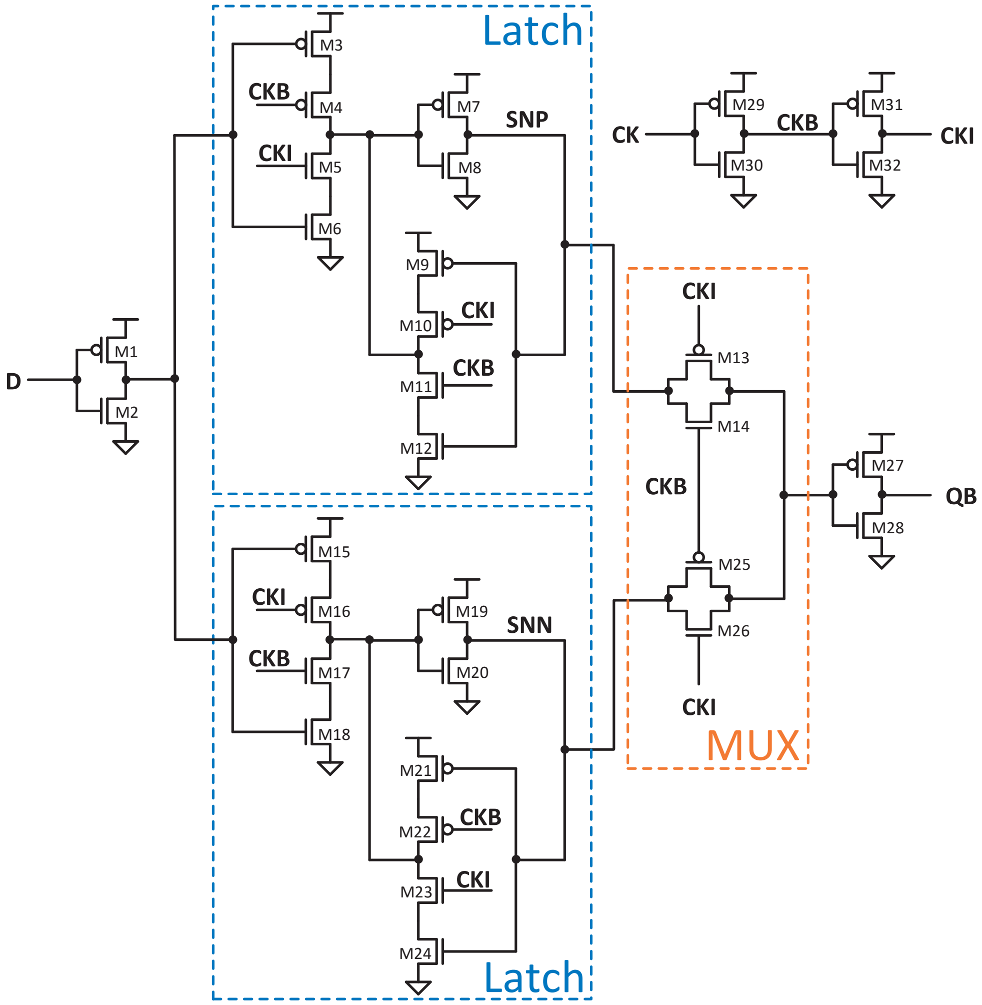


Figure 2.1: Schematic of the DET-TGLM flip-flop used in this work.

for the physical implementation and verification of DET clocking within the digital standard design flow.

2.1.2 Design and Implementation of DET Library Cells

The design of DET-FFs is crucial for achieving both low power consumption and high performance in synchronous digital systems. For this reason, many different implementations of DET-FFs have been proposed in the literature, including the transmission-gate latch-MUX (DET-TGLM) [24], the C^2MOS latch-MUX (DET-C2LM) [25], the pulse-triggered DET-FF [26], the conditional discharge flip-flop (DET-CDFF) [19], symmetric pulse-generator flip-flop (DET-SPGFF) [27] and the C-element flip-flops (DET-CFFs) [32]. Of these, the DET-TGLM is the most popular, due to its simple implementation and its relatively short CK-to-Q delay. In addition, this topology has proven to be one of the most energy-efficient DET-FFs for high-speed

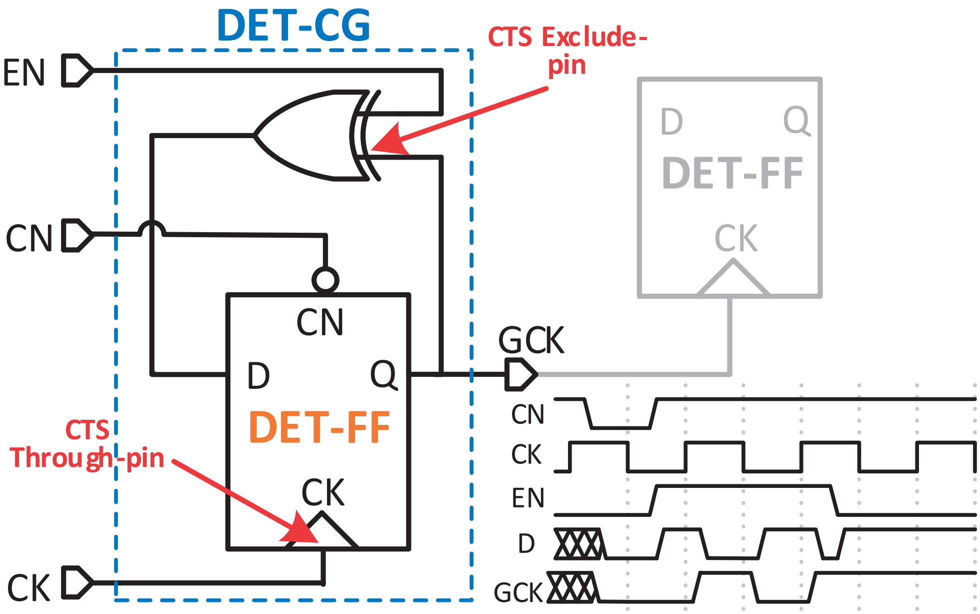


Figure 2.2: Schematic of the DET clock gate (DET-CG) [33] used in this work, characteristic waveforms and clock-tree synthesis (CTS) directives.

operation [21–23]. Thus, in this work, the DET-TGLM topology is chosen for demonstrating the application of DET clocking to full digital blocks. However, all of the proposed topologies are equivalently applicable, provided that they meet the $\beta_{\text{reg}} < 1$ requirement of (A.7).

The DET-TGLM, shown in Fig. 2.1, is composed of two separate latches (M3–M12 and M15–M24), whose storage nodes are connected to an output multiplexer (MUX). During each phase of the clock, only one latch is transparent, while the other one is opaque. This allows the transparent latch to follow the data at the input, while the other latch drives the output (QB) through the MUX. The MUX is based on transmission gates (TGs) (M13–M14 and M25–M26), which have a low propagation delay, to minimize the overall CK-to-Q delay. The total device count of this DET circuit is 32 transistors, while the clock load, defined as the number of transistors controlled by any clock signal, is 16 transistors. As the clock load of the corresponding SET-FF is 12 transistors, the ratio bewtween the DET and the SET clock load is 1.34 that is significantly lower than the clock-load-ratio limit of 2 for energy-efficient DET design, as defined in [17, 18]. The implemented DET-TGLMs also present the same clock-pin load as the replaced SET-FFs to ensure a similar load condition for the clock network. This is done by connecting the clock pin to the input of the inverter that is identical in both types of registers (M29–30 for DET-TGLMs in Fig. 2.1). In addition, while input glitching will result in some internal power consumption, the same happens for a standard SET-FF during one clock phase and, as previously mentioned, since the power consumed in DET-FFs is associated to only one clock edge per cycle, as opposed to SET-FFs, the dynamic power ratio (β_{reg}) is much lower than one, as required to meet the inequality of (A.7).

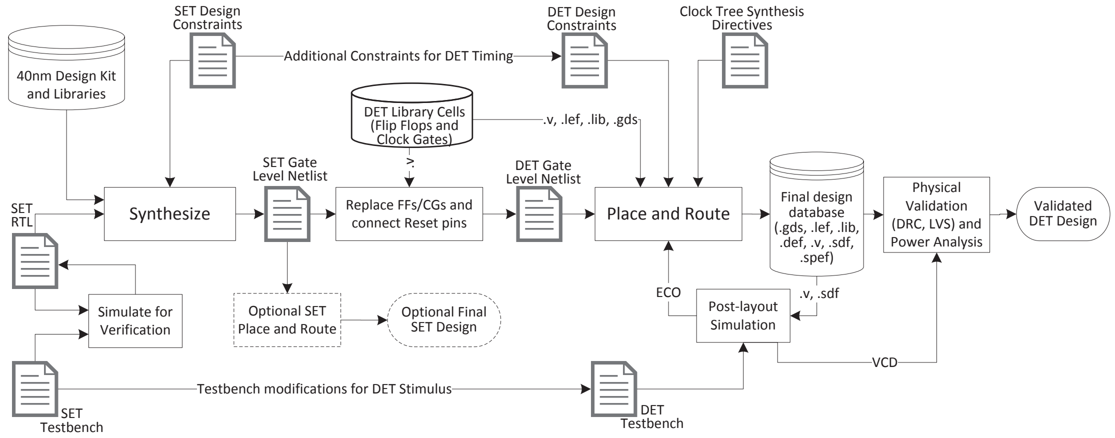

40nm Design Kit and Libraries
SET Design Constraints
Additional Constraints for DET Timing
DET Design Constraints
Clock Tree Synthesis Directives
DET Library Cells (Flip Flops and Clock Gates)
.v, .lef, .lib, .gds
SET RTL
Synthesize
SET Gate Level Netlist
.v
Replace FFs/CGs and connect Reset pins
DET Gate Level Netlist
Place and Route
Final design database (.gds, .lef, .lib, .def, .v, .sdf, .spef)
Physical Validation (DRC, LVS) and Power Analysis
Validated DET Design
Simulate for Verification
Optional SET Place and Route
Optional Final SET Design
ECO
Post-layout Simulation
.v, .sdf
VCD
Testbench modifications for DET Stimulus
SET Testbench
DET Testbench

Figure 2.3: Flow chart for the proposed DET implementation within the digital standard design flow.

The most commonly applied technique for reducing the power consumption of the clock network is clock gating. However, in the majority of the discussions about the potential efficiency of DET operation, clock gating is not addressed. Contrary to the many DET-FF implementations that have been proposed, to the best of the author's knowledge, only two implementations of a DET-CG have been published [33, 34]. In this work, the more straightforward one of these two circuits is chosen, comprising a DET-FF and a feedback XOR gate, as shown in Fig. 2.2. By choosing this circuit, the DET-CG can be defined as a structural Verilog model, rather than a fully custom standard cell, significantly reducing the design effort for integrating clock-gating in a DET design.

To demonstrate the operation of the DET-CG, the inset of Fig. 2.2 provides a set of typical waveforms. The XOR conditionally inverts the output (Q) of the internal DET-FF according to the enable (EN) signal, and feeds the result into the input (D) of the DET-FF. The clock input of the gate (CK) is directly connected to the clock pin of the DET-FF and the gated clock output (GCK) is connected to its Q pin. When the EN input is high, the XOR gate always feeds back the inverted output to the input of the DET-FF, causing the output clock to toggle on both edges of the clock. As soon as EN is set to '0', the DET-FF will continue sampling the same input value, forcing GCK to remain stable.

It is worth noting that unlike an SET clock gate (SET-CG), the output of a DET-CG depends on its initial state, providing either the same or the inverted phase relative to the root clock, depending on the activation cycle. The implications of this characteristic on the digital standard design flow are outlined in detail in Section 2.1.3. In addition, if the output of the DET-FF of the clock-gating cell is not forced to an initial state, the undefined value on the output of the DET-FF will propagate to its input through the XOR gate, creating an undefined logic level. Even though, in reality, this issue is mostly an artifact of the model appearing during gate-level simulation, a resettable DET-FF was used to implement the DET-CG, adding a reset input (CN) to the gate to set the output to a well-defined level during power-on.

2.1.3 Proposed DET Digital Design Flow

While the digital design flow for SET operation is extensively discussed in the literature and in textbooks [35], the implications of implementing a DET clocking scheme are rarely considered. This is probably due to the lack of DET gates in commercial standard-cell libraries along with the non-trivial requirements for DET timing analysis, especially when clock gating is used. This section introduces an automated flow for integrating DET operation into a block that is initially designed for standard SET operation and for then continuing through the digital standard design flow to provide a final, validated design database. This methodology enables an easy investigation of the DET power-efficiency for a given block and a seamless integration of DET blocks with SET blocks in a digital system. The only prerequisite for this flow is the existence of DET-FFs and DET-CGs, such as the ones previously described.

The proposed DET flow, illustrated in Fig. 2.3, is composed of three phases: first, a register-transfer level (RTL) description is provided for SET operation and standard synthesis is performed with SET timing constraints. Then, the design is translated into an equivalent DET netlist, primarily by replacing the SET registers and clock-gates with their DET counterparts. Finally, the adapted netlist is loaded back into the EDA tools and the digital standard design flow is continued with DET timing constraints to provide a final design database. The following subsections describe the intricacies of each of these phases, pointing out the directives and constraints that are required by the EDA tools for accurate DET timing analysis.

2.1.3.1 Initial SET Synthesis

The synthesis stage of the proposed flow follows the common SET digital standard design flow. An SET RTL description of the design is verified with a logic simulator and then mapped to a standard-cell library using a synthesis tool and a set of SET design constraints, including the automatic insertion of clock gates. At this stage, care must be taken to restrict the mapping of registers to SET-FFs for which equivalent DET gates are available in terms of interface and functionality (set, reset, scan, drive strength, etc.). Considering the gate-level implementation of the sequential elements, the synthesis tool instantiates SET master-slave D flip-flops with C^2MOS gates and latch-based clock gates, both provided by the standard library. The synthesized netlist is a fully operational SET version of the design, which can optionally be used for automated place-and-route (APR) and for the derivation of a final design.

The definition of timing constraints is a crucial stage in the proposed design flow and methodology for mixing SET and DET blocks within a given system. In the proposed flow, a reference frequency (f_{ref}) is defined according to the SET clock frequency, with the DET clocking frequency derived from this one ($f_{\mathrm{det}} = f_{\mathrm{ref}}/2$). All interface constraints (i.e., input and output delays) are defined with a virtual clock toggling at f_{ref}, as virtually all STA tools assume I/O toggling according to a single-edged clock. This is further clarified through the example Synopsys design constraints (SDC) commands of Fig. 2.4. By defining the interface constraints relative to a clock that is always toggled at the original SET frequency and is separate from

```
# create virtual reference clock with SET frequency
create_clock -name reference_clock -period $t_ref

# set i/o delays relative to reference clock
set_input_delay $input_dly -clock reference_clock \
        [remove_from_collection [all_inputs] [get_ports clk_in]]
set_output_delay $output_dly -clock reference_clock [all_outputs]

# Core logic clock definition:
#       For SET (e.g., during synthesis)
create_clock -name core_clk -period $t_ref  [get_ports clk_in]
#       For DET (e.g., during DET P&R)
create_clock -name core_clk -period [expr $t_ref/2] [get_ports clk_in]
```

Figure 2.4: Example of SDC commands for defining a virtual reference clock and interface constraints.

the core logic clock, the interface constraints can be left unchanged when applying the DET clock frequency during the APR stage. This definition not only ensures correct STA with regard to interface delays, but also allows to keep constraint definitions and testbench structures synchronized throughout all stages of the flow.

Note that DET clocking incurs stricter design requirements on the clock signal than SET clocking and may incur some overhead in the clock network due to the need for slightly increased margins (depending on how well the duty cycle can be balanced across all corners), primarily due to the relevance of duty cycle variation on timing. Ideally, a 50% duty cycle as well as sharp rising and falling transitions are desirable for the clock signal provided to the DET flip-flops. To account for this during synthesis, the clock uncertainty constraint (intended primarily for jitter) must be adjusted according to the specifications of the primary clock source of the system and to the expected duty-cycle uncertainty. Nevertheless, fast and balanced transition times are generally ensured by the cells of the clock standard-cell library, which are used for clock tree synthesis during APR. Any remaining deviations in duty cycle caused by inequalities in rise and fall times along the clock network will then be accounted for by the STA tools according to gate and parasitic wire delays, allowing the APR tool to ensure correct timing of the design across variations. Any resulting overhead from more stringent timing constraints in the critical paths is fully accounted for in the reported results.

2.1.3.2 Post-Synthesis Conversion to DET

Once the SET netlist is obtained, the conventional SET-FFs and clock gating cells have to be substituted with their corresponding DET versions. This substitution is easily performed by simple string replacements on the gate-level netlist, assuming that a corresponding DET gate is available for each SET-FF and SET-CG. Under the assumption that all rules of synchronous design [35] have been followed during the design and there are no macros included, the resulting netlist is now a functionally equivalent DET version of the original design where the registers are now sensitive to both clock edges and the DET-CGs perform the same clock-gating operation of the original SET-CGs.

One further consideration is the need to force the DET-CGs into a defined state, as described in Section 2.1.2. This requires connecting the DET-CG reset pin (CN in Fig. 2.2), which is usually not present in DET-CGs, to the global reset network. In order to not over-complicate the DET conversion script, the reset connection can be performed in the synthesis tool after reloading the modified netlist.

2.1.3.3 DET Automatic Place and Route Flow

The translated netlist is now the starting point for a fully-DET-aware APR flow. A working SET APR flow is assumed for the design, therefore only the differences required for the DET version will be explained. As previously described, the interface timing constraints are defined relative to a virtual clock, and therefore they can be used without any modification. The only required change is the constraint on the clock input of the DET block to $f_{\mathrm{ref}}/2$ to reflect that now both edges are active. An example of this modification is shown in Fig. 2.4.

Without the addition of clock gates to the design, the DET clocking scheme is handled properly by most STA tools, without further modifications to the constraints, and therefore the APR process can continue without any further modifications. However, the integration of the DET-CG, as described in Section 2.1.2, requires the definition of several additional constraints and directives in order to correctly time and optimize the design.

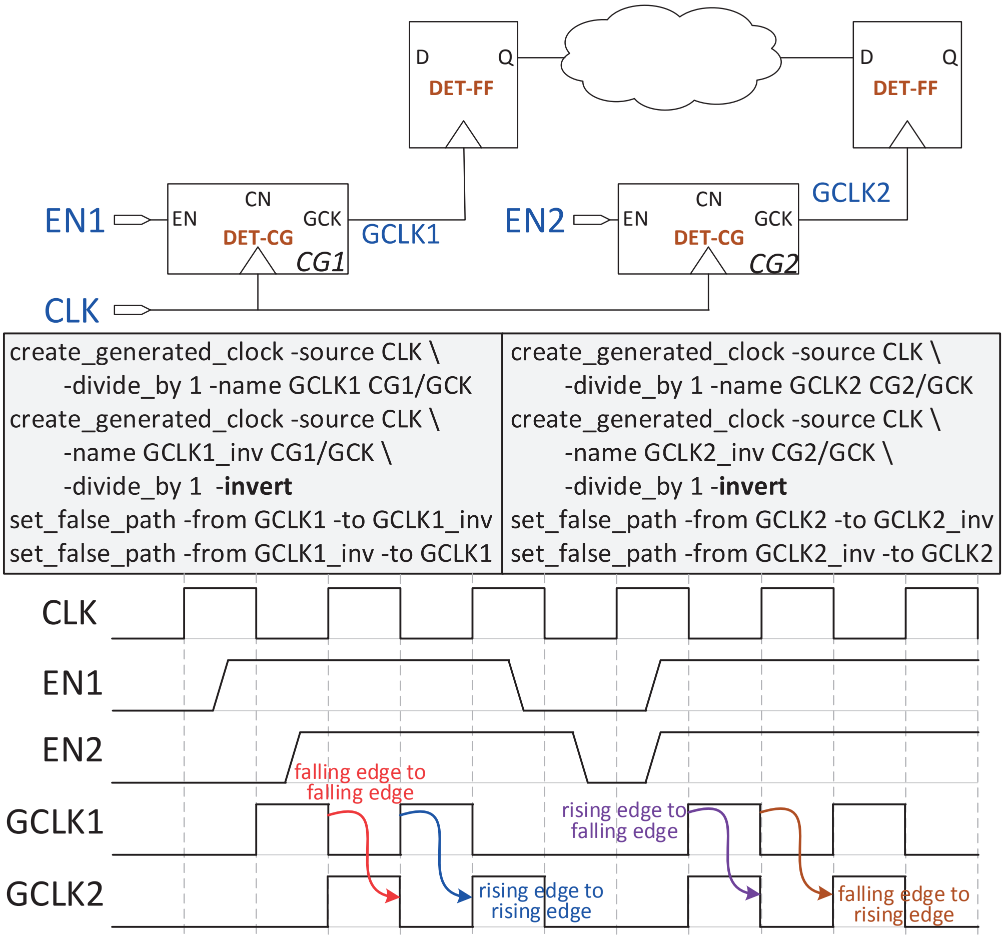
Clock Tree Synthesis Directives for DET-CG While standard SET-CGs, often provided within a standard-cell library, are generally recognized by CTS tools, care must be taken to ensure that custom-made gates, such as the DET-CG of Fig. 2.2, are treated appropriately. First, in order to ensure that the clock is traced through the DET-CG, the CK pins of the DET-CGs must be defined as through-pins. Second, to ensure that the tool does not attempt to skew-balance internal paths of the DET-CG, the feedback path to its XOR gate must be excluded from the CTS, if the DET-CG is provided as a structural module and not a closed standard cell. These directives are pointed out in Fig. 2.2.

Gated-Clock Phase Definitions The application of clock gating to a DET design introduces a non-conventional clocking phenomenon: due to the fact that the DET-CG can be enabled or disabled following either clock-edge, the various gated clocks in the design may have both a 0 degree or a 180 degree phase relationship. Therefore, on paths between registers gated by separate DET-CGs, the launch and capture registers could be triggered by either the same or opposite clock edges. This is demonstrated by the waveforms shown in Fig. 2.5 for a generic timing path. To ensure that the paths between all combinations of rising and falling edges are evaluated by the STA engine, a pair of generated clocks with opposite phases are defined on the GCK pin of each clock-gate. However, this definition also creates timing checks that can never occur in a DET design, such as rising-to-rising and falling-to-falling timing paths between registers clocked from the same source. Therefore, a false path must be defined between clocks that originate from the same DET-CG. Example constraints are shown in Fig. 2.5 in the SDC format. Defining these constraints will result in a correctly synthesized and constrained clock distribution including clock gates, which allows the DET design to be correctly timed by conventional STA tools.

2.1.3.4 Methodology for Mixed SET/DET Integration

The definition of a virtual reference clock, as described in Section 2.1.3.1, provides a simple and straightforward method to modify the testbenches developed for the original SET design for DET post-layout verification. In fact, we only need to derive a clock that is phase-aligned with the virtual clock at half its frequency and feed this divided clock into the top-level module clock port. Since the stimuli are defined according to the virtual clock at twice the speed of the clock that is fed into the block, they will, in essence, toggle at both internal clock edges, without any additional modifications to the RTL code. Therefore, by following the design flow, presented in the previous subsections, any block or component can quickly and easily be evaluated for operation with a DET clocking approach, including accurate power estimation through back-annotation of post-layout delays into a functional testbench.

This seamless transformation between SET and DET clocking schemes also provides the foundations for integration of SET and DET components within a single system. To achieve this, all that is needed is to create a divided DET clock in the top level block, and propagate either the fast SET clock or the slow DET clock to the corresponding blocks in the design hierarchy. This can be achieved a-priori in the RTL, with such a modification in mind, or on an existing design, using standard EDA tools. At an RTL level, each hierarchical block should be designed with both types of clocks at their interface, such that the selected clock to be fed into each block can be set simply by modifying the port connections of the module instantiation. For an existing design, a similar approach to the one used for inserting DET-CG reset connection can be used, i.e., disconnecting the clock port of a certain module and reconnecting it to the divided clock at the top level. As a design methodology, both SET and DET implementations of each component would be created, and can thereafter quickly be evaluated for both throughput and power consumption by simply switching between the two implementation options (SET and DET) of each block.



```
create_generated_clock -source CLK \
      -divide_by 1 -name GCLK1 CG1/GCK
create_generated_clock -source CLK \
      -name GCLK1_inv CG1/GCK \
      -divide_by 1  -invert
set_false_path -from GCLK1 -to GCLK1_inv
set_false_path -from GCLK1_inv -to GCLK1
```

```
create_generated_clock -source CLK \
      -divide_by 1 -name GCLK2 CG2/GCK
create_generated_clock -source CLK \
      -name GCLK2_inv CG2/GCK \
      -divide_by 1 -invert
set_false_path -from GCLK2 -to GCLK2_inv
set_false_path -from GCLK2_inv -to GCLK2
```

D
Q
DET-FF
D
Q
DET-FF
EN1
EN
CN
GCK
DET-CG
CG1
GCLK1
EN2
EN
CN
GCK
DET-CG
CG2
GCLK2
CLK
CLK
EN1
EN2
GCLK1
GCLK2
falling edge to falling edge
rising edge to falling edge
rising edge to rising edge
falling edge to rising edge

Figure 2.5: Waveforms showing edge combinations due to DET clock gating and example design constraints to ensure proper STA.

2.1.4 Implementation and Results

In order to demonstrate the proposed flow and to evaluate the efficiency of DET clocking, the presented design flow was applied to three SET benchmark circuits that, due to their high register count, were identified as promising potential candidates to achieve significant DET power savings. In particular, all the considered test cases are register-based and they are implemented without the need of any IP hard macro. For these experiments, the DET-FFs described in Section 2.1.2 were custom designed with Cadence Virtuoso and characterized with Cadence Liberate to provide the required .lib, .lef, .v, and .gds files for APR and STA. The DET-CGs were provided as a structural verilog model, without any need for custom layout or characterization. The designs were synthesized and mapped to a 40 nm CMOS standard-cell library with Synopsys Design Compiler and Mentor Graphics ModelSim was used for both RTL and post-layout logic verification. Replacement of the SET-FFs and SET-CGs with the DET cells was done with a custom Perl script and connection of the DET-CG reset pins was done using the netlist manipulation commands of Synopsys Design Compiler. The resulting

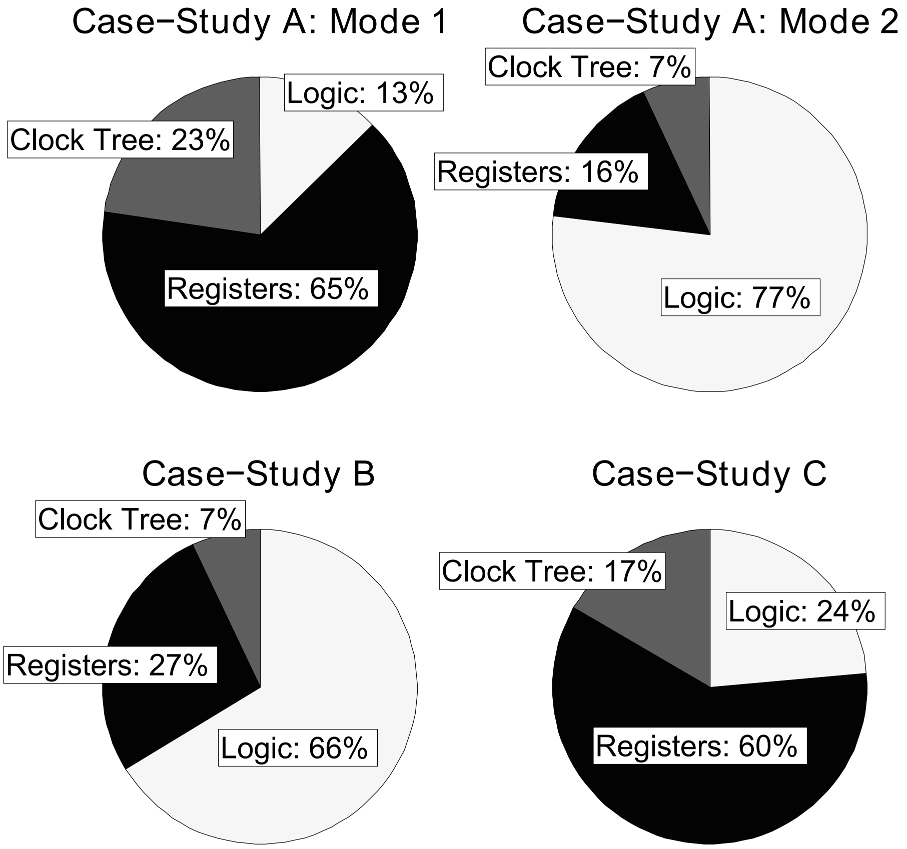


Figure 2.6: Breakdown of the total power consumption for the SET implementation of each case study.

DET gate-level netlist was fed into Cadence Encounter Design Implementation (EDI) for APR, and vector-based power analysis was performed within the EDI environment to obtain an accurate power breakdown, as presented below.

2.1.4.1 Test Cases and Implementation

Three IP blocks in a SET synchronous design style were selected as test cases to demonstrate the proposed design flow. Case study A (CS-A) is a reduced-state sequence estimator of a millimeter-wave wireless receiver, case study B (CS-B) is a digital prefilter for IEEE 802.11n, and case study C (CS-C) is a digital correlator of a power-optimized wake-up receiver. For CS-A, two separate modes of operation were considered with significantly different clock-gating efficiency, in order to emphasize the dependency of the potential power savings of the chosen clocking scheme on the actual runtime behavior (i.e., K_{tre} and K_{reg} in (A.7)). In the first mode of operation, CS-A exhibits a very high activity factor in its registers and only very short logic

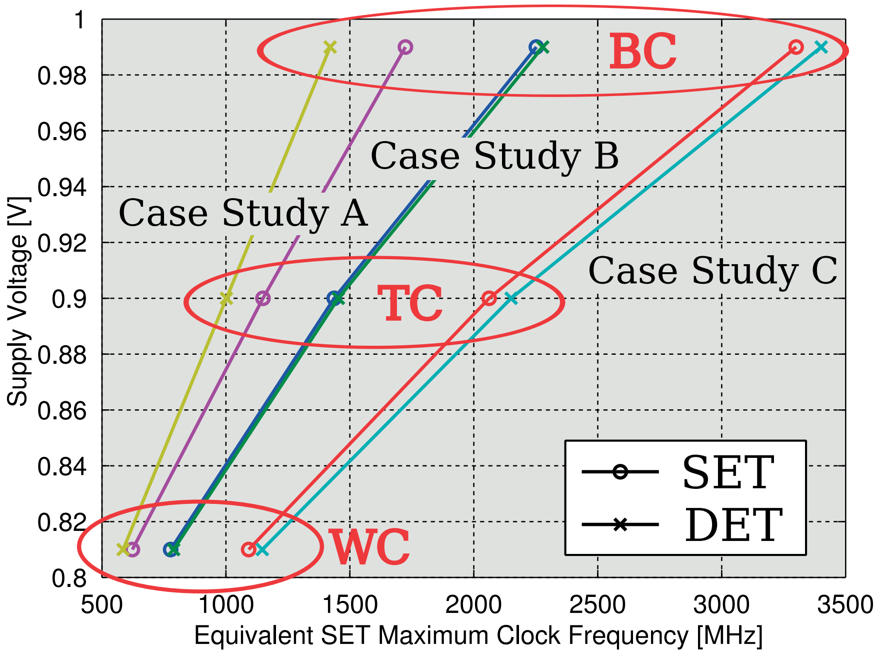


Figure 2.7: Equivalent SET maximum clock frequency achieved at different technology-process corners under a specific set of design constraints for each of the three case studies.

paths (i.e., with very little logic between register stages) are excited, showing the flip-flops to consume 65% of the total power in the SET implementation. In the second mode of operation, the percentage of registers power drops to 16%, due to higher clock-gating efficiency as well as due to a larger number of logic gates that are excited and therefore consume more dynamic power. This power difference is also present in the clock tree that is responsible for 23% of the total power in the first mode of operation and is reduced to only 7% when switching to the second operation mode, making the logic the main contributor to the total power consumption. Vector-based power measurements on post-layout netlists enable accurate evaluation of both clocking schemes under actual operating scenarios, showing that the first operating mode is expected to mostly benefit from the application of DET clocking to CS-A. The power breakdowns for all the SET implementations of each case study are provided in Fig. 2.6 for reference, where it is possible to identify CS-C as another good candidate for the implementation of DET clocking since the registers and the clock tree account for approximately 77% of the total power consumption.

Each of the blocks was first implemented with a standard SET flow to arrive at the application-specific target clock frequency. This operating frequency was used as the reference frequency (f_{ref}) for the DET flow, in order to target an equivalent throughput for a fair comparison of the clocking schemes. It is important to note that design constraints for both clocking schemes were not set to achieve the maximum operating frequency but rather to meet the requirements given by the design or the application. In order to verify that the

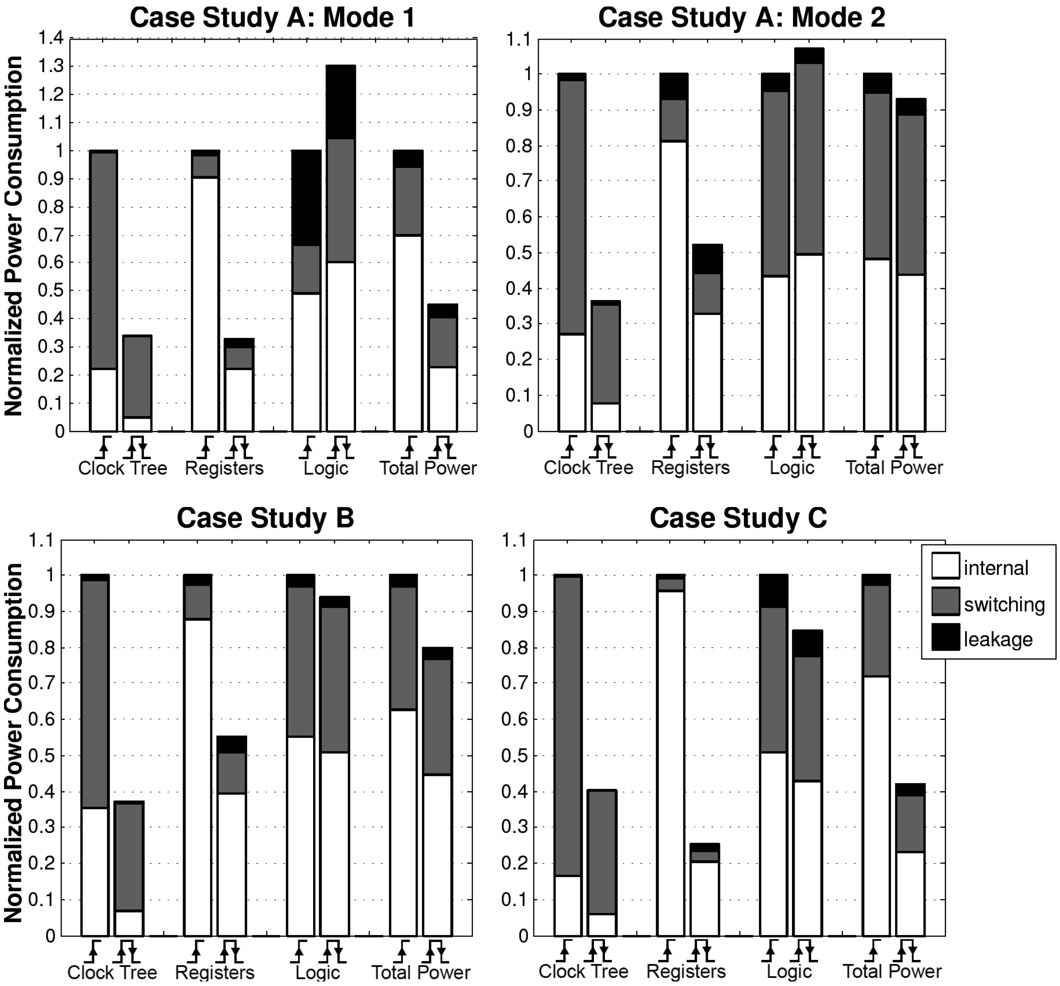


Figure 2.8: Power breakdown for each of the case studies, considering the contribution from the clock tree, registers, combinatorial logic, and the total power consumption. Each bar plot is normalized to the power consumption under SET clocking and it is further divided into internal, switching, and leakage power.

corresponding SET and DET implementations achieve a comparable throughput, the equivalent SET maximum clock frequency is reported in Fig. 2.7 that is defined as follows: while for the SET implementations the equivalent SET frequency coincides with the maximum operating clock frequency that ensures the correct functionality of the block (i.e., absence of timing violations), in the DET designs the equivalent SET frequency is derived by doubling the largest achievable operating clock frequency. Considering the three case studies, following full APR, the targeted throughput is always achieved for both SET and DET schemes for all the considered process corners, i.e., worst case (WC), typical case (TC), and best case (BC) corner. This result shows that, even under harsh timing constraints, the application of the DET flow does not cause any significant performance degradation across technology process corners.

Table 2.1: Power Breakdown Details of Case Study B

	Clock Tree		Registers		Logic		Total	
	SET	DET	SET	DET	SET	DET	SET	DET
Internal Power [mW]	0.64	0.13	6.01	2.69	9.36	8.62	16.01	11.44
Switching Power [mW]	1.13	0.54	0.65	0.79	7.04	6.79	8.82	8.12
Leakage Power [mW]	0.03	0.02	0.18	0.32	0.52	0.50	0.73	0.84
Total Power [mW]	1.80	0.69	6.84	3.80	16.92	15.91	25.56	20.4

2.1.4.2 Power Savings

The primary motivation for the application of DET clocking is the potential for power savings. As previously mentioned, an accurate power analysis was performed on each of the three test blocks, and for both operating modes of CS-A, according to vector-based simulations on post-layout netlists. The resulting power numbers are presented in Fig. 2.8, showing the power consumption for both clocking schemes. For each case study, the simulated clock frequency in the DET design is half of the SET clock frequency in order to provide equivalent throughput in both implementations. The presented power consumptions are divided into power consumed in the clock tree, by the registers, and by the combinatorial logic, as well as a summary of the total power consumption. Each of the bars in the plot is further broken down into internal power, switching power, and leakage power to provide a basis for a more in-depth analysis. The bar plots of the DET power consumption in this figure are normalized to the consumption under SET clocking for visual comparison. Absolute numbers considering the power breakdown of CS-B are provided for reference in Table 2.1.

Dependency on Registers and Clock Distribution The extracted power numbers are well aligned with the expected results as well as with the power tradeoffs predicted by (A.7) in Section 2.1.1. All cases, except mode two of CS-A, experience significant total power savings when implemented with DET clocking, due to the large fraction of the total power consumption of the clock tree and the registers in the SET implementation (Fig. 2.6). In relation to the inequality of (A.7), the high number of registers and clock buffers result in large values of both $E_{\text{reg}}^{\text{int}}$ and $E_{\text{tre}}^{\text{dyn}}$, as well as elevated activity factors K_{reg} and K_{tre}. This is clearly visible in the first mode of operation in CS-A where the DET design provides 56% savings on the total power consumption that drops to only 6% in the second operating mode, where the clock-gating efficiency is significantly higher. It is also worth observing the very significant power savings through DET in CS-C, which consumes 58% less power than the SET implementation. In this design, clock gating is not implemented, since registers are required to operate continuously, which makes DET clocking a very beneficial choice. This behavior can be observed once again

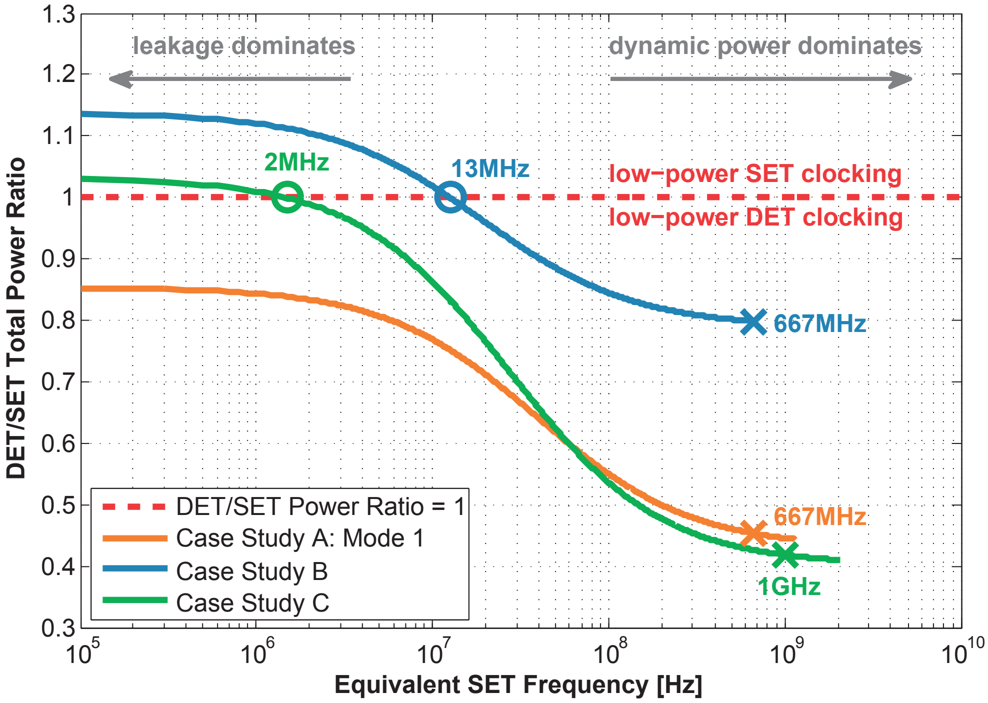


Figure 2.9: Dependency of the DET/SET total power ratio (DSPR) on the equivalent SET frequency that is defined in Section 2.1.4.1. The operating points of the presented simulations are marked with a cross, while the points where SET and DET power consumptions are estimated to be equal have been marked with a circle.

in the registers power breakdown of CS-C in Fig. 2.8, where the leakage power is only a very limited portion of the total power. Considering all the blocks, the power savings on the clock tree vary between 59% and 65% while the power savings in the registers are bounded between 44% and 74%, which is a direct result of the lower operating frequency and the lower power consumption of the DET-FFs compared to their SET counterparts.

Buffer Overhead for Hold Fixing An interesting observation in Fig. 2.8 is that mode one of CS-A experiences an increase in the logic power when DET is implemented. This is due to the relatively large number of buffers required for hold fixing, since this design has many short data paths with hold requirements that become more stringent when the faster (i.e., shorter CK-to-Q delay) DET-FFs are substituted for the conventional flip-flops. However, the increase in logic power consumption is a negligible share of the total power consumption in mode one, which is effectively reduced by the power savings in both registers and clock tree.

Dependency on Operating Frequency As mentioned in Section 2.1.1, the dynamic power savings obtained with DET clocking strongly depend on the frequency, and, in particular,

Table 2.2: Area Overhead of the Standard Cells (SCs)

Case Study	Number of SCs	Registers	SET-SCs Area	DET-SCs Area	Area Overhead
A	229 K	18.4 %	572,009 μm^2	642,505 μm^2	+ 12.33 %
B	25 K	17.4 %	48,764 μm^2	58,300 μm^2	+ 19.56 %
C	38 K	9.4 %	52,245 μm^2	57,953 μm^2	+ 10.93 %

they are maximized for high-frequency operation where the DET-FF leakage overhead is overcome by the dynamic power savings. This dependency is shown in Fig. 2.9, where the ratio between the DET and the SET total power consumption is estimated across different equivalent SET frequencies, defined in Section 2.1.4.1, for all the case studies where the DET clocking significantly reduces the overall power consumption. This DET/SET power ratio (DSPR) figure-of-merit shows which of the two considered clocking schemes is the most power efficient for a certain equivalent SET frequency. In particular, if this ratio is less than one, then DET clocking is more power-efficient, while SET clocking is preferable for low-power operation when DSPR>1. The DSPR is equal to one where both the SET and DET implementations provide the same power consumption. In Fig. 2.9, the operating points derived from the power simulations are marked with a cross (×). Starting from these points, the power ratio has been extrapolated and plotted across different equivalent SET frequencies using (A.3) and (A.6). As expected, the DET/SET power ratio rapidly decreases for high frequencies, since DET clocking is more power-efficient in this frequency range, while the SET scheme is generally a good option for slow-operating systems where leakage power dominates. In particular, CS-B and CS-C are more power-efficient if implemented with DET clocking as soon as they are operated at a minimum equivalent SET frequency that is larger than 2 MHz.

For a visual comparison, the operating points crossing the DSPR threshold are indicated with a circle (○). On the other hand, mode one of CS-A never crosses the unit power threshold due to the unexpected lower leakage power of the DET design. Since the total leakage power consumed by CS-A is mainly given by the digital gates implemented in the logic, this improvement is due to a better design optimization performed by the APR tool in the DET implementation. Considering the case studies where a large amount of power is consumed in the clock tree and the registers, this figure confirms that DET clocking is power-efficient at high operating frequencies when the dynamic power dominates and the relative power savings are maximized, as predicted in (A.7).

2.1.4.3 Area Overhead

The DET-FFs are usually larger than their SET counterparts, and this has an impact on the silicon area required by the DET implementation of each case study. In this study, both the resettable and non-resettable versions of the DET-TGLM, presented in Section 2.1.2, have been provided with different output strength (X1, X2, and X4). The implemented DET registers result in an area overhead that varies between 40–80% when compared to the SET-FFs provided by the commercial standard cell library,

Considering the size of the designs, the number of standard cells (SCs) implemented in each of the analyzed case-studies is reported in Table 2.2, together with the SC area for both SET and DET designs. In the considered case-studies, the registers are a relatively large portion of the designs and they account for up to 18.4% of the total number of cells. The resulting area overhead depends on the percentage of registers in the design, on the type of the replaced flip-flops and on any impact on the surrounding logic caused by the design optimization of the DET design by the APR tool. The DET area overhead is less than 20% for all the DET designs – a relatively small cost compared to the significant benefit of the obtained power reduction.

2.1.5 Conclusion

The motivations and tradeoffs involved in the choice between SET and DET clocking for low power consumption have been described in this section where the power reduction due to the use of DET clocking was demonstrated on three case studies in a 40 nm CMOS process technology, achieving up to 58% power savings for a register-heavy design when compared to the conventional SET implementation. The power benefits in clock distribution are enabled by the proposed design flow which is one of the first works to directly address the intricacies of the physical implementation of DET designs, especially in the light of clock-gating insertion. In the presented methodology, all components are initially designed as standard SET blocks and subsequently all the SET clocking elements are replaced with their DET counterparts after synthesis, before continuing to APR. With implementations of both clocking schemes at hand, the power-area tradeoff can be evaluated and the best clocking strategy can be chosen.

2.2 An Overlap-Contention Free True-Single-Phase Clock DET Flip-Flop

Several implementations of registers that can be triggered on both clock edges are reported in the literature [19, 24–28]. DET storage cells are generally characterized by a lower power consumption [21–23] and a smaller propagation delay [17, 21–23] when compared to conventional SET-FFs. However, few of the different DET topologies reported in the literature have been examined in deeply-scaled process technologies under voltage scaling, commonly used for the implementation of energy-efficient systems. In particular, in the presence of considerable process variation, the use of both clock phases can introduce some extent of clock overlap, which can lead to race conditions and other detrimental circuit behavior. For example, when considering the traditional DET-TGLM, process, voltage and temperature (PVT) variations can cause this overlap to increase to a point, where the currently held data is over-written, resulting in a fatal logic error.

Contributions The main contributions of this section are summarized as follows:

- The risk of clock-overlap failure is solved by presenting the first static true-single-phase-clock (TSPC) DET-FF.
- By implementing the DET-FF with TSPC circuits and an internal dual-feedback mechanism, completely static operation is achieved, enabling robust operation under voltage scaling and process variations.
- The cell was implemented in a 40 nm CMOS process technology, showing full functionality at a near-threshold supply voltage (V_{DD}) of 0.5 V and under extensive Monte Carlo (MC) statistical simulations. Also, the proposed register provides the lowest CK-to-Q delay and the best power-delay product (PDP) when compared to other leading DET-FFs.

This section is organized as follows: Section 2.2.1 presents the clock-overlap hazard in the traditional DET-TGLM circuit. The proposed static dual-edge-triggered flip-flop with true-single-phase clock (SDET-TSPCFF) is presented in Section 2.2.2 to address this hazard and enable low-voltage operation. Section 2.2.3 provides simulation results for the proposed cell and a comparison with other popular DET-FF implementations. Section 2.2.4 concludes this section.

2.2.1 Clock-Overlap Failure Risk in DET-TGLM Cells

This section explains the risk of failure in DET-FFs due to clock overlap. The DET-TGLM was chosen to demonstrate this hazard, as it is the most popular DET-FF implementation. Accordingly, a brief overview of the DET-TGLM is provided in Section 2.2.1.1, followed by a

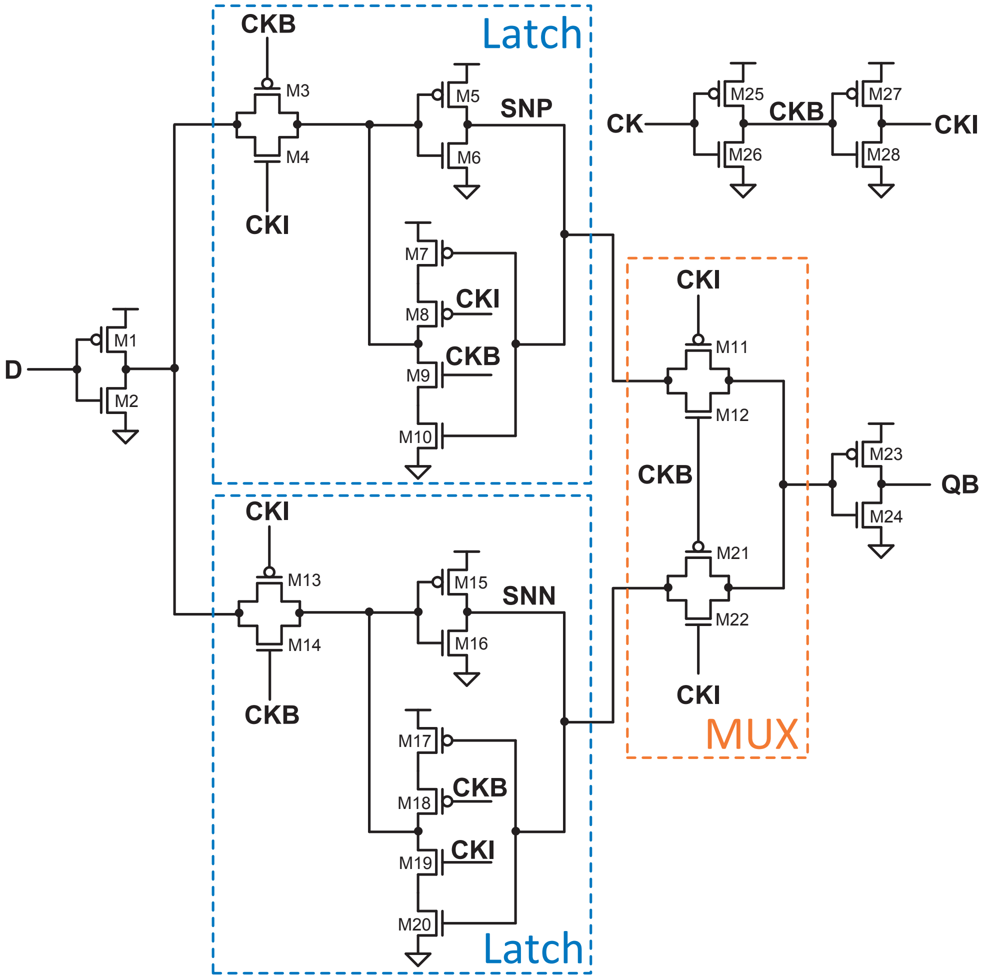


Figure 2.10: Schematic of the DET-TGLM.

detailed analysis of the risk of failure due to clock overlap in Section 2.2.1.2. Note that while this discussion is specific for the DET-TGLM, a similar analysis applies to many other DET-FF implementations.

2.2.1.1 Overview of the DET-TGLM

As explained before in Section 2.1, among the various DET-FFs, the DET-TGLM is one of the most commonly implemented topologies, primarily due to its simple structure and straightforward behavior. Two values are stored internally in two separate latches that are connected through an output MUX, as illustrated in Fig. 2.10. The latches are implemented with input transmission gates (M3–M4 and M13–M14), inverters (M5–M6 and M15–M16), and clocked inverters (M7–M10 and M17–M20), while the MUX is exclusively composed of transmission gates (M11–M12 and M21–M22). Each latch is transparent during a different phase of the clock, and the value stored in the opaque latch is passed through the MUX to the output.

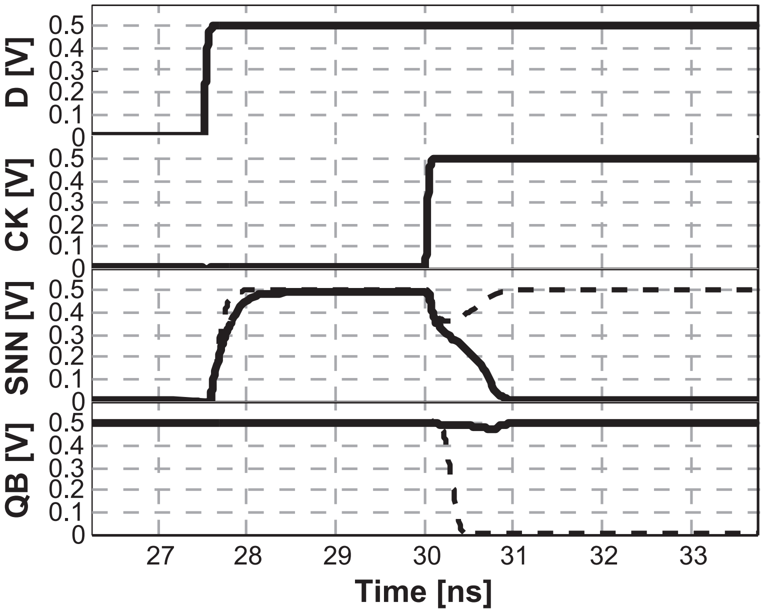


Figure 2.11: Voltage waveforms of the signals affected by the clock overlap in the DET-TGLM. The failure is represented with continuous lines while the segmented lines show the case when the circuit overcomes the hazard.

In addition to its simple structure, this register has been shown to be one of the most energy-efficient DET-FFs for high-speed operation [29]. During its respective transparent window, each latch passes the input data to its cascaded transmission gate (SNP and SNN in Fig. 2.1), such that the data only needs to propagate through the output MUX on the next clock edge. This provides a short t_{cq}, which makes the DET-TGLM suitable for high-frequency applications. Finally, this circuit does not rely on pulse-triggered circuits or precharge (dynamic) conditions, such as those required by [19, 26, 27], making it less sensitive to variations from technology and voltage scaling.

2.2.1.2 Clock-Overlap Failure Risk

The static operation of the DET-TGLM provides inherent robustness; however, one problematic feature remains – its dependency on both clock phases for functionality. To accommodate this need, the inverted clock signal is internally generated with an inverter (M25–M26). A second inverter (M27–M28) is implemented to internally buffer the input clock and ensure a controlled and fast slew rate of CKI. Thus, CKB and CKI share the same value during an interval of time that is equal to the intrinsic delay of the inverter composed by M27–M28. The time during which both clock signals are high is defined as positive clock-overlap (PCO), while negative clock-overlap (NCO) occurs when both clock signals are low. These overlap phases occur immediately after each transition of the clock signal that is used to generate the inverted one. Since both clock signals are equal during such an overlap, there is always one

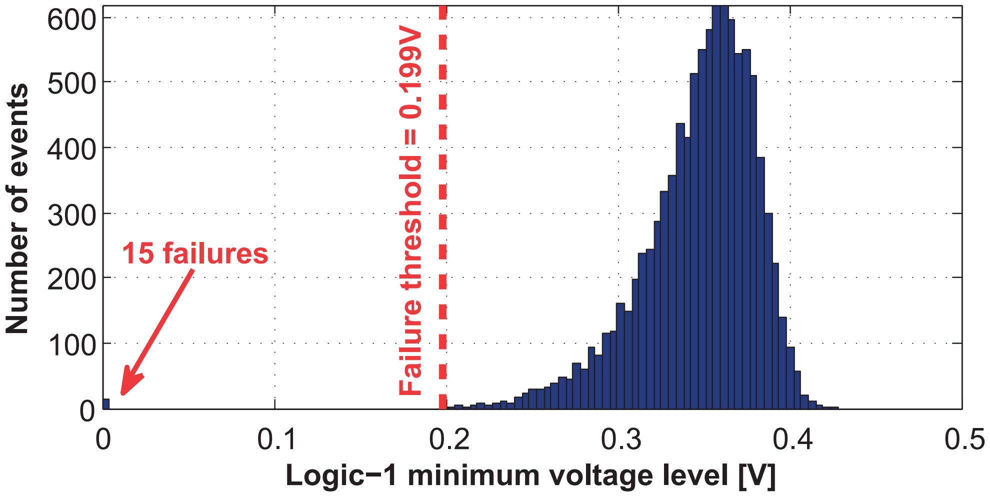


Figure 2.12: Distribution of the minimum voltage value reached by the storage node SNN during the clock overlap for 10,000 Monte Carlo (MC) runs.

type of transistor (either NMOS or PMOS) turned on in each transmission gate of the MUX (M11–M12 and M21–M22). A conducting path is therefore generated between the inputs of the MUX in the DET-TGLM, causing an internal race between the values that are stored in the two latches. The clock overlap time is heavily dependent on PVT variations and wire parasitics. If the overlap is too large, the voltage value stored in one latch will overwrite the value stored in the other latch, resulting in a storage failure.

Fig. 2.11 demonstrates the behavior of a DET-TGLM gate under a typical hazardous disrupt. In this example, the clock is initially low and a logic-0 is stored at SNP and passed through the top transmission gate to the output. During this phase, the bottom latch is transparent, passing a logic-1 from the input (D) to SNN. After the rising edge of the clock, both CKI and CKB are low during the NCO and the PMOS transistor in each transmission gate is conducting. If the NMOS that is pulling down SNP (M6) drives more current than the PMOS that is driving SNN (M15) and if the overlap time is sufficiently long, the voltage value on SNN will drop until it is overwritten by a logic-0 through the cross-coupled feedback of the bottom latch. Following the overlap period, this logic-0 value is latched and driven through the MUX to provide the wrong value at the output. The transient waveforms of SNN and QB during a failing event are shown as a solid line in Fig. 2.11, while a case where the circuit overcomes the hazard is shown with a dotted line. The same failure risk can be studied for the case where CKB and CKI are high, and a logic-0 value stored at SNP is the critically affected value.

As previously described for the case of NCO, a failure occurs when the voltage at SNN drops below a critical threshold that results in a latched logic-0 level. To evaluate the probability of such an occurrence, statistical MC simulations are employed, applying global and local process variations to a DET-TGLM gate during a NCO phase. Fig. 2.12 displays the obtained distribution of the minimum voltage level of SNN achieved during transient simulations for

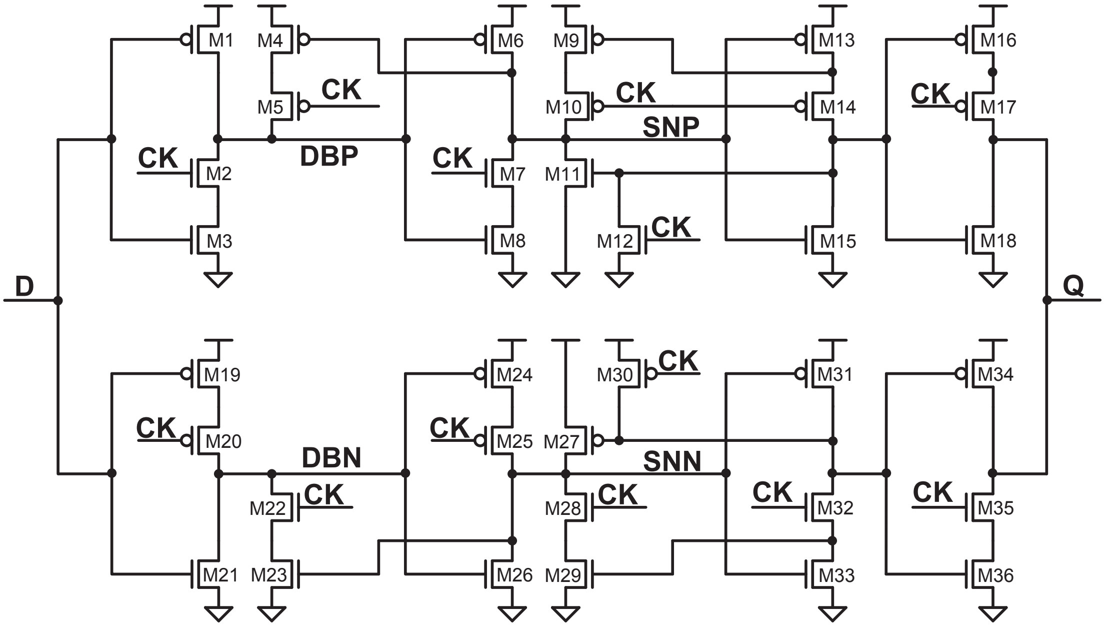


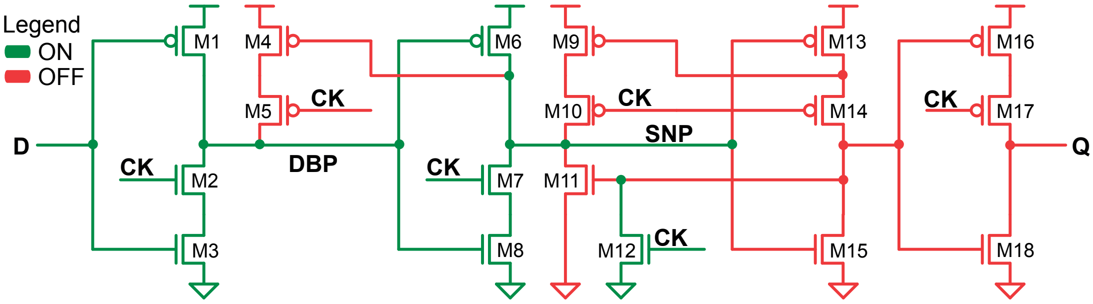
Figure 2.13: Schematic of the proposed SDET-TSPCFF.

10,000 MC samples applied to a DET-TGLM implemented in a standard 40 nm CMOS process. The simulations were run with a near-threshold V_{DD} of 0.5 V at 125°C. The distribution of Fig. 2.12 shows that the logic-1 stored in SNN can temporarily drop from 0.5 V (i.e., supply voltage) to a median minimum value of 0.36 V during the NCO phase. The reported minimum voltage level is worsened by the effect of global and local process variations and a failure threshold can be estimated at 0.199 V, which is the minimum voltage level for a stored logic-1 that is still overcome by the gate without causing a failure. Out of the 10 k samples, 15 resulted in a storage failure, as can be seen by the non-zero probability of voltage levels centered around 0 V. Thus, it is clear from the presented distribution and the large number of failures that the DET-TGLM is not a viable candidate for near-threshold operation.

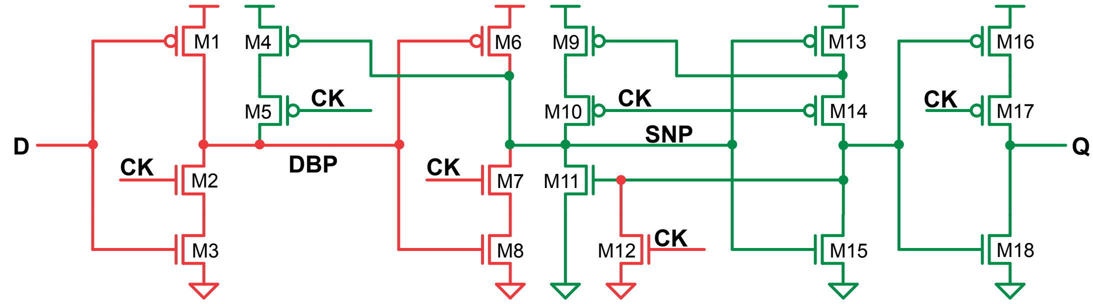
The described risk of clock-overlap failures can be overcome by isolating the storage nodes of the DET-TGLM from potential aggressors. For example, clocked inverters can be used instead of transmission gates to implement the output MUX of the DET-TGLM, at the cost of a higher transistor count. In this thesis, this circuit implementation is referred as isolated-storage latch-MUX (DET-ISLM). As an alternative solution, the overlap of different clock phases and the related risk of potential failures can be avoided by the use of a TSPC scheme [36, 37] in DET-FFs, whose operation does not require to invert the clock.

2.2.2 Proposed SDET-TSPCFF

In the previous section, the traditional DET-TGLM gate was shown to be unsuitable for near- or sub-threshold operation in scaled technologies, due to the risk of clock overlap failures. In



(a) Transparent phase with clock equal to one



(b) Opaque phase with clock equal to zero

Figure 2.14: Top branch of the SDET-TSPCFF.

order to overcome these risks, a fully-static TSPC alternative to the DET-TGLM and to other dual-phase solutions is proposed. Other TSPC DET-FFs have been shown in the past [19, 26, 27, 38, 39]; however, they all rely on temporary dynamic storage [38, 39] and/or generated pulses [19, 26, 27], which make them sensitive to both process variations and voltage scaling.

The schematic of the proposed SDET-TSPCFF is shown in Fig. 2.13. Similar to other latch-MUX DET-FFs, new data is written to an internal storage node during one clock phase and subsequently latched and driven to the output following the clock transition. This is achieved without the need for an inverted clock signal by implementing the storage elements with a pair of TSPC latch-MUX branches (M1–M18 and M19–M36). These branches are loosely based on the classic TSPC latch [40] with the addition of two internal feedback mechanisms that ensure strong data levels and fully-static operation to enable robust low-voltage functionality.

To further explain the circuit operation and its feedback mechanisms, we will focus on the top branch (M1–M18) of the SDET-TSPCFF, as shown in Fig. 2.14 for both high and low phase of the clock, with the opposite branch operating in a completely symmetric fashion. When CK is high, as shown in Fig. 2.14(a), devices M1–M8 act as a buffer, passing the value at D to SNP. This buffer does not encounter any contention with other parts of the circuit, as M5, M10 and M11 are all cut off. In addition, in this state, the output of the top branch presents a high-impedance to Q, as M17 cuts off the pull-up to this node and M12 pulls down the gate of M18, cutting off the pull-down to the output. When CK goes low, shown in Fig. 2.14(b), the current state of SNP is latched, since M7 cuts off the pull-down and M2 cuts off the pull-down

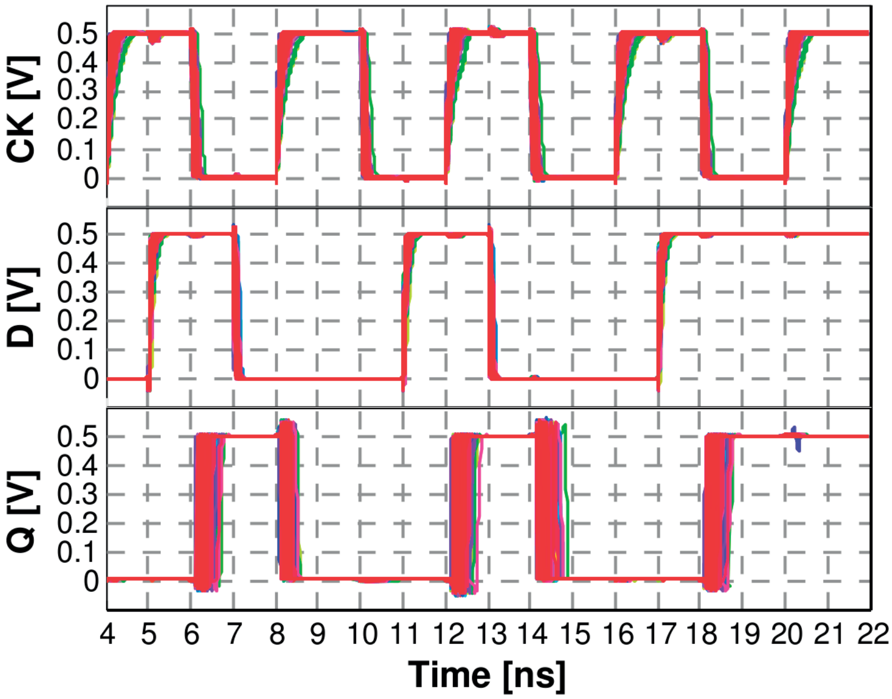


Figure 2.15: Monte Carlo family plots of the SDET-TSPCFF for 10,000 runs.

path to DBP, disabling a pull-up through M6. It is essential to ensure that DBP does not drift and possibly turn on M6, and therefore, a feedback path from SNP to M4 maintains a logic-1 at DBP if SNP was latched at logic-0. Moreover, in this state, devices M9–M15 comprise a cross-coupled inverter that holds the level at SNP through a strong positive-feedback loop. Finally, devices M16–M18 function as a tri-state inverter, selectively and robustly passing the storage value to the output.

While the 36 transistors required to implement this gate are larger than the 28 transistors required by the DET-TGLM or many of the other DET-FFs, the additional area enables static overlap-contention free operation, thereby providing variation-tolerant functionality at scaled supply voltages and for advanced process technologies.

2.2.3 Simulations and Results

The performance of the proposed register is evaluated considering two groups of simulations. First, the resilience of the storage cell against failures is tested with MC simulations to show its robustness. Second, the SDET-TSPCFF is characterized in terms of speed and power consumption in order to compare it with other popular static DET-FF implementations. All circuits were implemented with standard-V_T transistors in a 40 nm CMOS process technology for comparison.

Table 2.3: Summary of the compared DET-FFs [1]

	DET-TGLM [24]	DET-ISLM	DET-C2LM [25]	This work
Transistor count	30	32	22	38
Clock load	16	16	8	14
CK-to-Q delay (t_{cq}) [ps]	230.9	213.0	395.3	149.9
Internal power [2] (P_{int}) [nW]	344.4	343.9	386.2	287.5
Total power [2] (P_{tot}) [nW]	431.6	431.2	495.3	449.9
Leakage current (I_{leak}) [nA]	12.7	13.1	15.4	17.1
Power-delay product [3] (PDP) [aJ]	99.7	91.9	196.8	67.5

[1] Based on the testbench proposed in [30].

[2] At 500MHz input clock frequency and 25% data activity.

[3] PDP = $t_{\mathrm{cq}} \cdot P_{\mathrm{tot}}$

In the first considered testbench, all possible combinations of data are written inside the storage cell, and subsequently checked at the output during the next clock phase. The output value is continuously sampled and failures are reported if it differs from the expected value. Both process and mismatch variations are taken into account while running MC simulations. Furthermore, near-threshold operation is targeted by setting V_{DD} to 0.5 V. This set of runs is executed for each of the following temperatures: 0°C, 25°C, and 125°C. An example of the family plots obtained through a set of simulations is shown in Fig. 2.15. The proposed register provided the correct output for all 10,000 samples at each temperature point, indicating robust functionality under these operating conditions.

The performance of the proposed cell is compared with other latch-MUX-based storage cells that do not rely on pulse generation. The characterization of the storage cells is performed using the testbench proposed in [30], where several state-of-the-art DET-FFs are simulated and compared. All the simulations were applied to 40 nm CMOS implementations of the considered circuits with a V_{DD} of 0.5 V and at 25°C. The frequency of the input clock is 500 MHz, corresponding to a cell throughput of 1 GHz, and the data activity is 25% (i.e., input data toggles on average every four clock cycles).

The results of the comparison with other DET-FFs is summarized in Table 2.3, showing that in addition to solving the clock-overlap failures, the proposed SDET-TSPCFF also provides the lowest t_{cq}. The DET-C2LM shows the worst performance in terms of speed and total power consumption, as it highly relies on tristate logic which compromises its performance at near-threshold operation. Thus, much wider transistors are required in order to operate correctly at low voltages, resulting in a severe area and power consumption penalty. Both delay and power consumption of the DET-ISLM are very similar to the ones of the DET-TGLM, as these DET-FFs only differ by the output MUX. The advantage in t_{cq} of the proposed circuit as compared to the DET-TGLM is due to the reduced conductivity of its transmission gates at scaled voltage supplies. The leakage and total power consumption of the presented register is slightly higher than those of the DET-TGLM; however its PDP is lower.

2.2.4 Conclusion

The failure risk due to clock-overlap in popular DET-FFs that implement transmission gates as output MUX was demonstrated in this section, showing an unacceptable error-rate at near-threshold voltages in a 40 nm CMOS process technology. A fully-static TSPC DET-FF that solves this clock-overlap risk has been presented in this section and has been shown to be fully functional at a similar operating point, under local and global process variations, and at a wide range of temperatures. Also, the proposed register was found to provide the best CK-to-Q delay and power-delay product among popular DET-FFs.

3 Circuits and Techniques to Monitor and Trim Dynamic Timing Margins

The vast majority of digital systems relies on synchronous operation of its sequential elements for the management of their internal states as well as for the communication with other systems. For this, a global and periodic clock signal is used as time reference to change the state of the registers in the system. In an ideal system, the delays of the paths between registers are constant and the registers receive the exact same clock signal, to be perfectly synchronized. However, in reality, the presence of environmental changes as well as the use of a network to distribute the clock across the whole system introduce changes in the propagation delays of the paths and differences between the arrival time of the clock signals at each register. These variations are either spatial or temporal and, in clock distribution, they are generally referred to as skew and jitter [40], respectively.

The presence of these uncertainties on both data paths and clock distribution forces the designer to introduce design guard bands (i.e., margins) that prevent timing violations under all operating conditions. Even though the introduced margins are mandatory for correct functionality of the synchronous system, they might degrade the maximum frequency at which the system can operate and they generally increase the complexity as well as the power consumption of the clock network. The impact of guard bands on both timing and power is even more pronounced in nanometer nodes and at scaled voltages. In fact, both process variations and dynamic variations such as changes in temperature, aging, and voltage noise have a more severe impact on nanometer circuits when operated in the near-threshold regime, resulting in a very high timing uncertainty [2].

However, environmental parameters vary with different time scales; for example, the impact of temperature and aging varies rather slowly, while droops on the supply voltage last for a very short amount of time. Thus, the system often operates under non-critical conditions, where the timing conditions are met even for a small or even no guard band. Several techniques have been proposed to trim guard bands at runtime with the use of dynamic voltage and frequency scaling (DVFS) and timing-monitoring circuits, such as replica circuits [41] and error-detection sequentials (EDSs) [42]. Timing-monitoring circuits can track any change on the critical paths of the design, allow the system to adapt, and reduce the guard bands as well as their impact on timing and power consumption while always ensuring a reliable operation.

Beside the presence of timing variations due to slow environmental changes, as the critical path might not always be excited, the timing profile of a digital circuit is also characterized by the presence of *dynamic* timing margins. Thus, for further timing and power improvements, cycle-by-cycle DVFS techniques have been proposed to rapidly trim any available dynamic timing margin by either increasing the operating frequency to maximize the throughput or by applying voltage scaling for fixed throughput rates to reduce the power consumption [43, 44].

In this chapter, circuits and techniques are proposed to exploit dynamic timing margins and for timing monitoring:

- A custom-designed clock generator capable of immediate and glitch-free adjustments of the clock period for the exploitation of dynamic timing margins is described in Section 3.1.
- Section 3.2 presents a timing-monitoring sequential that can be programmed at runtime to either detect setup-timing violations or to measure the available positive timing slack, depending on the desired timing-monitoring strategy.

3.1 A Clock Generator for Cycle-by-Cycle Dynamic Clock Adjustment

The basic assumption for the design of synchronous systems is that the clock period must be set according to the critical-path delay. However, the critical path is not always excited [45], therefore dynamic timing margins exist. Constantin *et al.* [43] showed that the critical-path delay in a conventional microprocessor core can significantly vary depending on the type of the executed instruction. Thus, depending on the current state of the pipeline, the clock period can be dynamically adapted to meet the delay of the longest excited path to reduce the temporary timing margin and therefore improve the average throughput and/or energy-efficiency of the microprocessor core. This technique is referred to as dynamic clock adjustment (DCA) [43].

One of the key aspects when applying the DCA technique to a microprocessor is that the pipeline state together with its corresponding timing margins can change at every clock cycle. Thus, in order to effectively benefit from the DCA approach, a clock source that is capable of immediate adjustments of the generated clock period is required. However, clock generators generally do not support this strict requirement due to the conventional assumption that the output clock period is static.

Contributions For this section, the main contributions are the following:

- A dynamically-adjustable clock generator (DCG) capable of clock-period adjustments on a cycle-by-cycle basis and that can be implemented as primary clock source to enable the DCA technique is proposed.
- The DCG is custom designed with standard cells that have been placed in a controlled and regular floorplan to maximize the linearity of the output clock-period range.
- The immediate and glitch-free change of the generated clock period is ensured by internally double-sampling the input delay setting and by a careful design and controlled layout of the cells involved in this timing-critical path.
- The proposed clock generator is designed in 28 nm FD-SOI process technology and it has been fabricated on a test chip for its characterization together with a 32-bit microprocessor core to demonstrate the benefits of the DCA technique.

In this section, a more detailed overview of the DCA technique and the design intricacies behind its application are first given in Section 3.1.1. The proposed DCG is described in Section 3.1.2. Characterization of the DCG based on silicon measurements together with the application of DCA to a microprocessor core are presented in Section 3.1.3. Section 3.1.4 concludes this section.

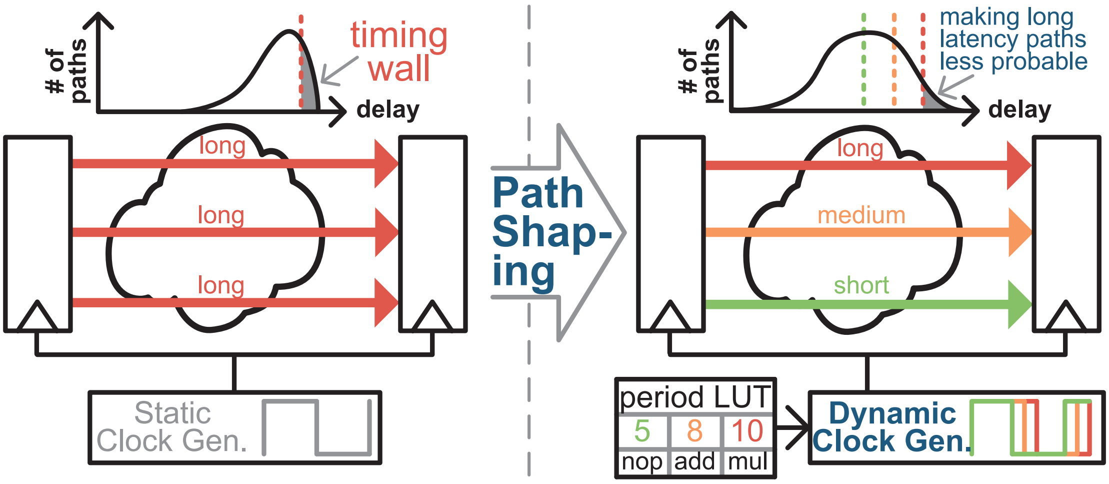


Figure 3.1: The effect of path shaping and the application of the dynamic clock adjustment (DCA) technique [44].

3.1.1 Concept of Dynamic Clock Adjustment

The DCA approach is presented in [43] and relies on the observation that the critical path of a design is not always excited. When this is the case, shorter paths limit the maximum operating frequency, therefore dynamic timing margins exist and they can exploited by either reducing the clock period to maximize the throughput or by applying voltage scaling for fixed throughput rates to reduce the power consumption. Considering a microprocessor core, it has been shown that, with proper design, the excitation of the paths temporarily limiting the maximum operating frequency during each clock cycle depend on the type of instructions that are currently residing in the processor pipeline.

The netlist of the processor produced by a conventional synthesis is generally characterized by a large number of paths with a delay close to be critical [43]. When looking at the path-delay profile of the design, this results in a “timing wall” which limits the dynamic timing margins and therefore the potential benefits from the application of DCA, as shown in Fig. 3.1. This limitation can be overcome by instructing the synthesis tool to optimize not only the critical paths but also the sub-critical paths for shorter delay at the cost of only 5-13% power increase in 28 nm FD-SOI, as presented in [43].

Beside applying path shaping to maximize the dynamic timing margins, the design of a processor requires to support DCA to benefit from it, as described in [44]. To this end, a DCA module is integrated within the processor for providing the cycle-by-cycle clock period setting to the DCG based on the type of instructions that are in-flight in the processor pipeline, as shown in Fig. 3.2. In particular, the DCA module partially decodes the fetched instruction to determine its type and, based on this information, retrieves the corresponding clock period from a lookup table (LUT). We note that in the case of [44] it is sufficient to only consider the timing of the execution stage, since all critical and near-critical paths are within this stage.

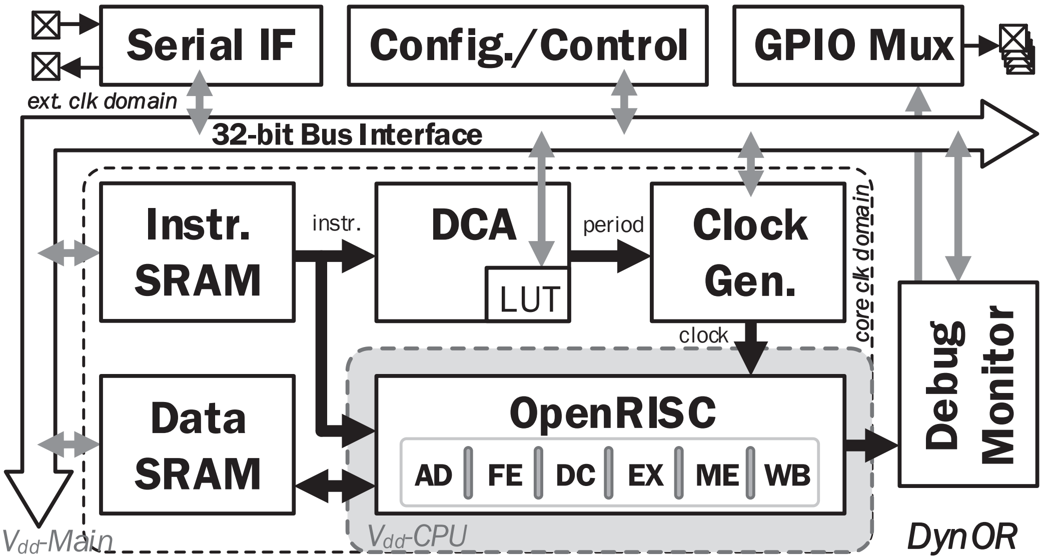


Figure 3.2: Architecture of a microprocessor supporting dynamic clock adjustment (DCA) [44].

The DCG receives the value of the clock period from the DCA module and it generates the corresponding clock signal for the subsequent clock cycle when the instruction resides in the timing-critical execution stage of the processor pipeline.

3.1.2 Dynamically-Adjustable Clock Generator

The DCG, whose schematic is shown in Fig. 3.3, is a digitally-controlled ring oscillator (DCRO) [46] that produces a clock signal with a period that can be changed at every clock cycle, as required by the DCA. To modify the propagation delay inside the ring, the DCG includes two programmable delay lines (PDLs): the DCA PDL and the static PDL. While the static PDL ensures a minimum and constant delay, the DCA PDL can be reprogrammed at runtime and is designed to ensure a cycle-by-cycle adjustment of the produced clock period without producing glitches on the clock signal. Each PDL is implemented with a cascade of AND gates, providing 64 delay settings and a high linearity of the produced range of clock periods as a custom floorplan is used for the placement of these cells within the DCG.

The cycle-by-cycle operation of the DCG is ensured by the use of a one-hot encoded period setting for the DCA PDL and by the sampling of this delay setting with a set of additional registers that have been placed close to the DCA PDL to minimize both their propagation delays as well as the delay of the clock signal they receive from the ring oscillator (*ck_d1_n* in Fig. 3.3). These registers are clocked as soon as the falling edge of the clock has propagated through the DCA PDL. Thus, the delay setting is stable at the input of the DCA PDL as soon as the next rising clock edge arrives, avoiding glitches on the clock signal as well as any contamination of the generated clock period. The delay setting of the static PDL is assigned before enabling the DCG and it is never modified at runtime, therefore its control is not

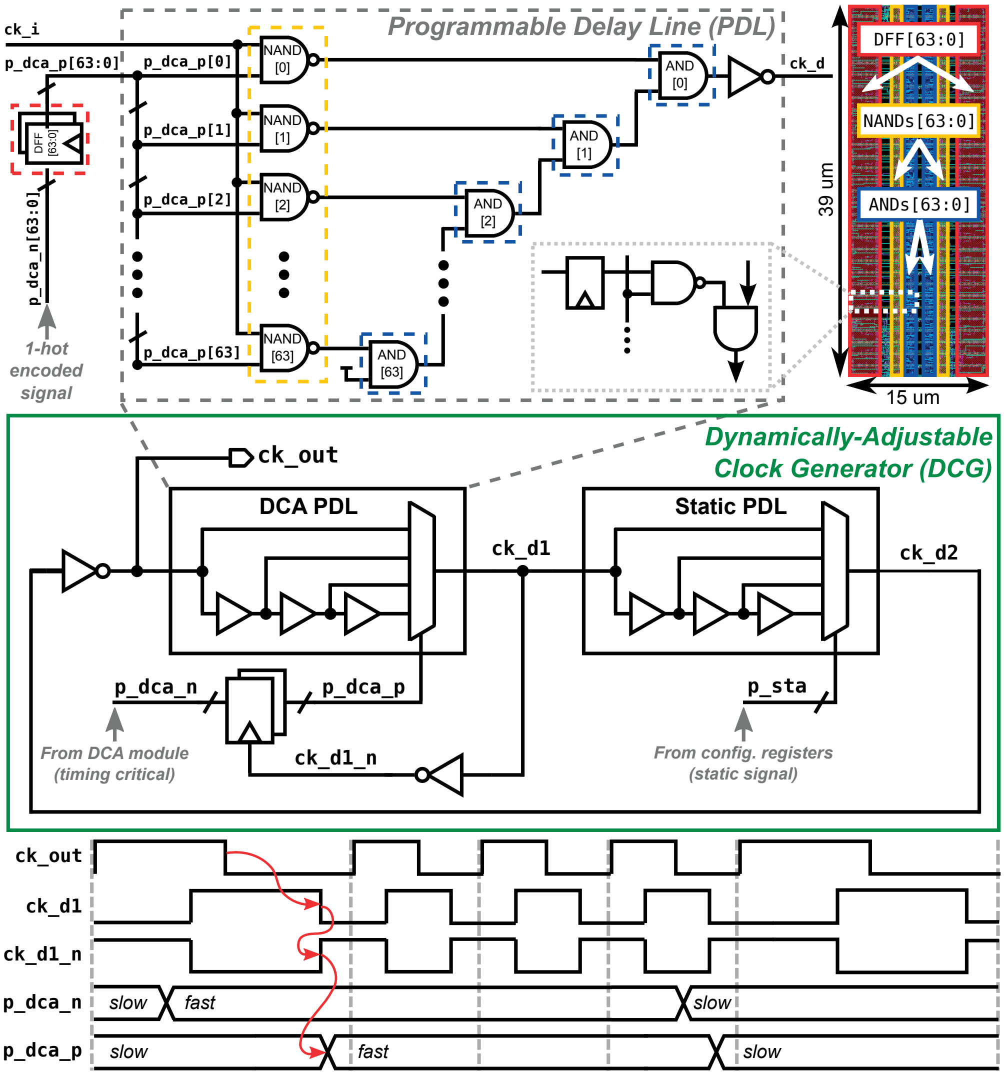
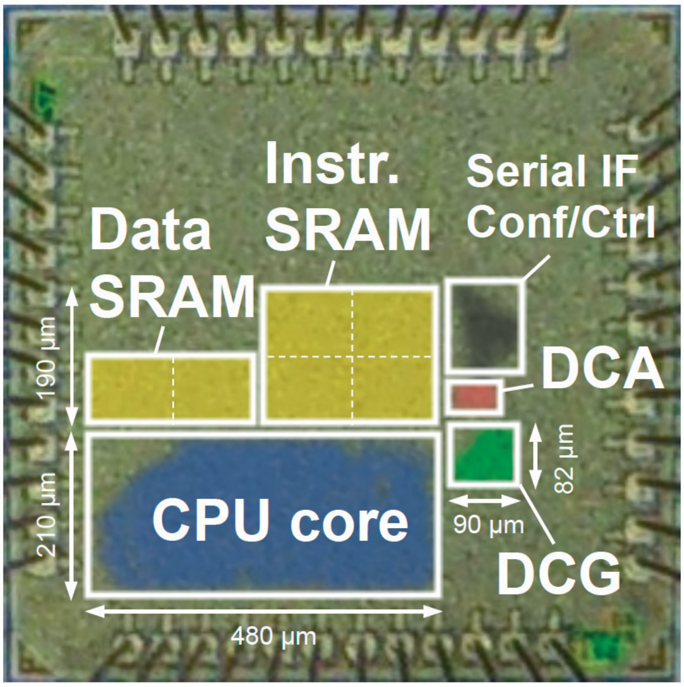


Figure 3.3: Schematic of the dynamically-adjustable clock generator (DCG), layout of the programmable delay line (PDL) and internal timing.

timing critical. Moreover, the delay of the static PDL can be increased to relax the tight timing constraint of the DCA PDL as a longer static PDL delay allows for more time for the delay setting to stabilize at the input of the DCA PDL. For reference, the internal timing of the DCG is shown on the bottom side of Fig. 3.3.

3.1.3 Test Chip and Measurements Results

The DCG has been fabricated on a test chip in 28 nm FD-SOI process technology together with a DCA-enabled CPU core to characterize the clock generator and to show the potential of the DCA technique when applied to a microprocessor. The DCG occupies a total silicon area of 7360 μm^2 while the complete system uses 0.24 mm^2 of the entire 1.2 mm^2 die, as shown in the



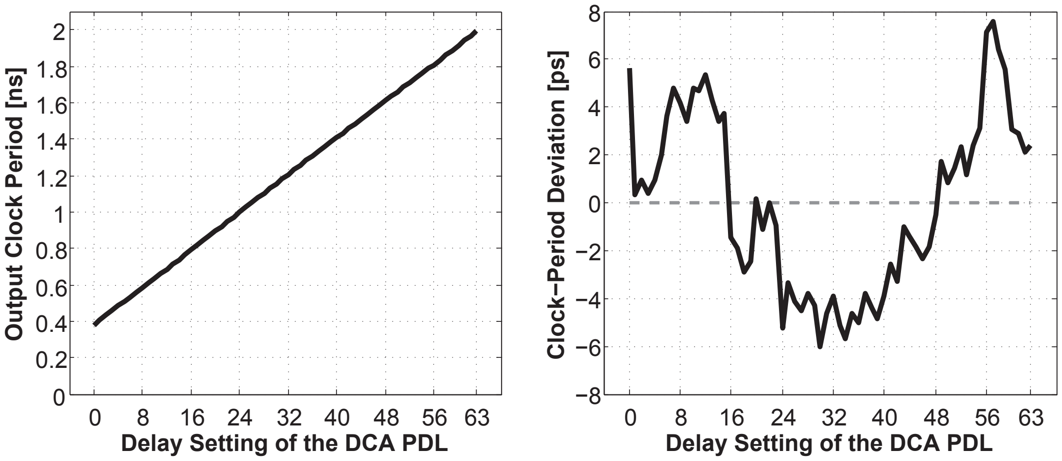
Technology	UTBB FD-SOI 28 nm RVT
CPU	32b, 6-stage OpenRISC with dynamic clock
Die area	0.24 mm^2 (chip: 1.2 mm^2)
Memory	16 KB (IM) + 8 KB (DM)
Gate Equiv.	172 k (logic) + 144 k (SRAM)
V_{dd} range	0.40 V – 1.10 V (cpu min: 0.26 V)
Freq. range	24 – 1306 MHz
Power range	0.31 – 112 mW
Energy range	13 – 86 pJ/Op

Figure 3.4: Die micrograph of the test chip in 28 nm FD-SOI process technology implementing the dynamically-adjustable clock generator (DCG) and the microprocessor supporting the dynamic clock adjustment (DCA) technique [44].

die micrograph in Fig. 3.4. The DCG can be supplied with either the same or a higher voltage than the CPU as they both reside in dedicated voltage domains, therefore allowing to tune the generated range of clock periods.

3.1.3.1 Characterization of the DCG

The two PDLs used to implement the DCG provide 64 delay settings. Nevertheless, measurements of the CPU showed that the use of only the DCA PDL is sufficient to cover the range of clock periods required by the DCA approach and that the internal timing of the DCG is met without the need of increasing the default minimum delay setting (chosen based on the post-layout timing analysis) of the static PDL. For this reason, the presented measurements of the DCG are all conducted with the fastest delay setting assigned to the static PDL. The range of clock frequencies generated by the DCG is from 365 MHz to 1906 MHz and from 502 MHz to 2654 MHz while the average step granularity is 34 ps (3 FO4) and 26 ps (3 FO4) for a supply voltage of 1.0 V and 1.2 V, respectively. The linearity of the output clock period at 1.2 V is shown in Fig. 3.5 together with its deviation from the function obtained with linear regression which is always bounded within a ±8 ps (±1 FO4) range.



(a) Range of produced output clock periods.

(b) Deviation of the produced output clock periods from the function obtained with linear regression.

Figure 3.5: Measurements of the output clock periods produced by the DCG for every delay setting of the DCA PDL while the static PDL is set to its shortest delay and for a voltage supply of 1.2 V.

3.1.3.2 Application of the DCG to a DCA-Enabled Microprocessor

The DCA approach is applied to a 32-bit 6-stage in-order OpenRISC [47] microprocessor with one integer ALU, supporting single-cycle 32-bit multiplications. The fabricated CPU is designed to support the DCA by shaping its path-delay profile to maximize dynamic timing margins. A DCA module is integrated in the system together with the proposed DCG that is used as main clock source. A typical die operates at a maximum static frequency of 576 MHz when using a supply voltage of 0.8 V and 1.0 V for the CPU and the DCG, respectively. For this voltage-frequency point, the total power consumption is 31.3 mW, where the CPU accounts for 69% and the DCG for 6% while the remaining power is consumed by the memories.

Two applications are chosen to measure the speedups and power savings given by the DCA approach: matrix multiplication and median calculation. While the matrix multiplication requires the execution of a large number of multiplications which excite the worst paths of the CPU, the median calculation, as it is based on sorting, performs a more balanced set of instructions, therefore the longest paths are less frequently excited. Speedups are calculated relative to the baseline speed achieved by the CPU when clocked at the constant maximum operating frequency. Measurements are performed on 25 dies, for different supply voltages (0.6 V, 0.8 V, and 1.1 V), and at room temperature (25°C). When the DCA module is specifically calibrated for the median calculation, considering all measured dies over different supply voltages, the speedup is 25% on average and can reach up to 41%. Even when the DCA module is specifically calibrated for the matrix multiplication application, an average speedup of 6% is achieved. The presented application speedups can be traded off for a reduction in power consumption by applying voltage scaling. These measurements have been performed at fixed throughput rates, achieving up to 15% power reduction for the median calculation when decreasing the supply voltage of the CPU from 0.80 V to 0.75 V at 521 MOp/s. A more detailed analysis of the DCA approach and its benefits is out of the scope of this thesis as it has already been presented and demonstrated in [43, 44, 48].

3.1.4 Conclusion

The critical path of a synchronous might not always be excited, therefore dynamic timing margins exist and they can be exploited by DCA. A DCG has been proposed in this section to enable the DCA approach whose application showed to provide either up to 41% throughput increase or up to 10% power savings on a microprocessor fabricated in 28 nm FD-SOI. The internal double-sampling of the input delay setting as well as the careful design and controlled layout of the cells involved in timing-critical paths enable immediate and glitch-free changes on the frequency of the clock signal generated by the DCG which was measured to produce an output-frequency range from 365 MHz to 1906 MHz with an average step granularity of 34 ps at 1.0 V on a 28 nm FD-SOI test chip.

3.2 A Timing-Monitor Sequential for Forward and Backward Error-Detection

Dynamic voltage and frequency scaling (DVFS) is a well-known and effective technique to reduce the power consumed by digital circuits. However, its application to designs in nanometer nodes is challenging due to the large effects that process and dynamic variations have at near-threshold operation. This lead to the introduction of costly guard bands that are added on top of the nominal supply voltage to ensure always-correct operation under any operating condition. However, the system can often be operated reliably without the need of these large and conservative design margins as some of the dynamic variations, such as temperature and aging, have rather long time constants. Thus, error-detection sequentials (EDSs) have been proposed to monitor any change on the critical paths of the design to adaptively trim the guard bands whenever they are not needed, therefore maximizing the power savings.

Many EDSs have been reported in the literature [5–10], with a particular attention on reducing the design overhead of the error-detecting part of the registers [7, 8]. These *in-situ* timing monitors are usually implemented at the end point of the most critical paths. The basic principle of these registers is to sample the input data twice: once at the active clock edge and once during an error-detection time-window that starts with the active clock edge and generally coincides with a part of the high-phase of the clock. If the two sampled values differ, a late-arriving transition on the data input of the register is detected, which flags a setup timing-violation.

When referring to the literature, the Razor family [5, 6, 8] is probably the most known and most popular collection of EDSs. The first published implementation of the Razor register consists of a flip-flop augmented with a shadow latch that samples the data input with a delayed clock [5] while in RazorII the error-detecting window relies on pulse-generation [6]. The overhead of the error detector is minimized in iRazor where a latch senses a virtual rail to detect late-arriving transitions, requiring the addition of only three extra transistors [8]. Intel presented an EDS in [9], where a time-borrowing latch and a shadow flip-flop sample the input data on the falling and rising edge, respectively. Tadros *et al.* [10] presented an error-detecting latch in [10] where the timing-error monitor is insipired by the sense amplifier that is generally implemented in SRAMs.

Even though the EDS is designed to detect late-arriving signals, short paths that arrive at the same end point may excite the data input of the register within the error-detection window, albeit in the same clock cycle in which the data is launched, before the next active clock edge. This type of event erroneously signals a violation in the timing monitor. Common approaches to overcome such short-path problems are the addition of hold buffers to short paths or the reduction of the length of the error-detection window [49].

A limitation of many EDSs is their inability of warning when the delay of a path becomes *close* to the critical-path delay as they only generate an error signal after a timing violation has already occurred, usually to activate an error-correction mechanism. However, this comes with a severe overhead (e.g., pipeline stalling). As an alternative, a timing-fault sensor which measures the available positive timing-slack of a path has been proposed in [50].

With these two alternatives (EDS vs timing-fault sensor), the choice of either monitoring setup timing-violations or measuring the available timing-slack of certain paths is relegated to the design phase. Sequentials allowing to choose either one of these timing-monitoring modes would enable to select at runtime the most desirable timing-monitoring strategy. However, to the best of the author's knowledge, they have not been proposed yet. Also, a limitation of the timing-fault sensor is that it only allows to monitor paths that are *close* to become critical. Nevertheless, as the critical path of a design is not always excited, the activated paths might have a significantly shorter delay compared to the critical path, depending for example on the instruction executed by a microprocessor core [43]. To also include such fast paths in the slack measurement, for example for adjusting DCA LUTs, the available timing-slack needs to be measured for a wide detection-window. Such a wide measurement window allows to exploit even large dynamic timing margins to either increase the operating frequency or by applying dynamic voltage scaling at runtime depending on the instruction executed in the core [44].

Contributions The main contributions of this section are summarized here:

- A timing-monitoring sequential (TMS) designed in 28 nm FD-SOI which is capable of detecting transitions on the input data, during either the high or the low phase of the clock, here defined as forward-detection mode (FDM) and backward-detection mode (BDM), respectively, is proposed.

- While FDM provides detection of late-arriving transitions, as in conventional EDSs [8], the additional BDM functionality of the TMS enables measurement of the available timing-slack as in the timing-fault sensor [50], without affecting the safe concurrent operation of the circuit.

- The size of the detection windows in FDM and BDM are set by the high and the low phases of the clock, respectively, therefore the length of these windows can be adjusted by controlling the duty cycle of the clock.

- By adjusting the low phase of the clock in BDM, the TMS can measure the timing slack of paths that are far from being critical. This capability can provide insightful information to either exploit available dynamic-timing margins [43, 44] or for offline diagnostics.

- Results from post-layout simulations are provided to evaluate the performance of the TMS, and both FDM and BDM are verified with measurements from a test chip fabricated in 28 nm FD-SOI process technology, which included a 16-bit digital multiplier, implemented with TMS cells.

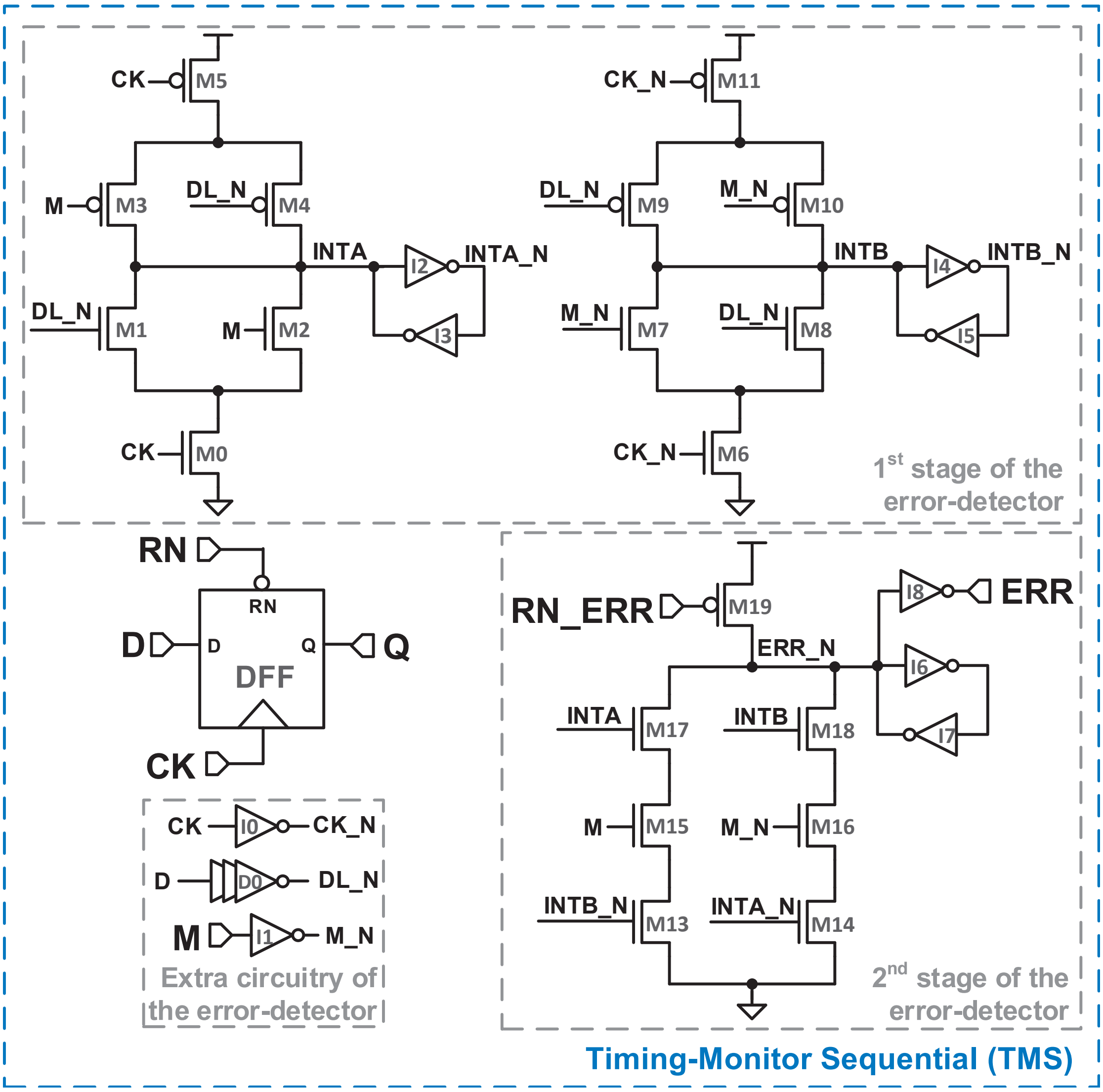


Figure 3.6: Circuit schematic of the timing-monitoring sequential (TMS).

In this section, the proposed TMS is presented in Section 3.2.1. The timing analyses enabled by the TMS are discussed in Section 3.2.2. Section 3.2.3 presents post-layout simulation results of the TMS and silicon measurements of a 16-bit multiplier implemented with TMS cells. Section 3.2.4 concludes this section.

3.2.1 Proposed Timing-Monitoring Sequential

The proposed TMS is composed by augmenting a resettable D flip-flop (DFF) with a two-stage error detector that provides two modes of operation. The TMS is, in fact, capable of detecting any input transition occurring either during the high phase or during the low phase of the clock, depending on whether it is set to forward-detection mode (FDM) or backward-detection

Table 3.1: Truth table of the internal error signals in the TMS

Case Number	M	CK	D	INTA	INTB	Error-Detector
0	0	1	0	0	previous	active
1	0	1	1	previous	1	active
2	0	0	0	1	0	idle
3	0	0	1	1	0	idle
4	1	1	0	0	1	idle
5	1	1	1	0	1	idle
6	1	0	0	previous	0	active
7	1	0	1	1	previous	active

mode (BDM), respectively. The circuit schematic of the TMS is shown in Fig. 3.6, where the input ports are clock (CK), input data (D), active-low reset of the DFF (RN), active-low reset of the output error (RN_ERR) and the error-detection mode select (M), while the output ports are output data (Q) and the error flag (ERR). By assigning logic-0 or logic-1 to input M, the operation mode of the TMS is set to FDM or BDM, respectively. Whenever a transition on the input data D is detected in the active time-window, the output ERR rises and remains high. This error flag is reset by enabling the RN_ERR input to prepare the TMS to detect any transition in the next detection window.

The error-detection circuitry of the TMS is shown inside the gray dotted lines in Fig. 3.6. The input data to the error detection circuit is delayed and inverted by a delay line (D0) to provide a delay margin that ensures that any transition of the input data happening at the beginning of the active time-window is correctly detected as an error. The error detector is divided into two stages, where the first stage (M0–M11 and I2–I5) generates two internal signals (INTA and INTB). These signals show if the input data D has reached either both or only one of the logic states during the error-detection window, which corresponds to the presence or the absence of a transition on D, respectively. The second stage (M13–M19 and I6–I8) receives INTA and INTB and if a transition on D has been detected during the active window it drives the output error signal ERR to logic-1. As the internal nodes INTA, INTB and ERR_N are driven by dynamic logic (M0–M19), latches (I2–I7) are added to each of these nodes to ensure that their logic state is kept even when there is no activity on the input ports of the TMS. Dynamic logic was used to reduce the transistor count in the error detector, while also minimizing the delay of the error generation.

The complete truth table of the TMS is provided in Table 3.1. For each case, it is specified if the error detector is active and a case number is assigned for reference. The error detector generates an error (ERR=1) when one of the following conditions is met:

- In FDM (M=0), when INTA=0 and INTB=1.
- In BDM (M=1), when INTA=1 and INTB=0.

Considering the FDM, during the low phase of the clock INTA and INTB are forced to logic-1 and logic-0, respectively, regardless of the input data value. Therefore, any transition on D does not generate any error (cases #2 and #3 in Table 3.1). In the subsequent high-phase of the clock, the condition on INTA and INTB to generate an error is met only if both cases #0 and #1 manifest within the same high-phase of the clock, which corresponds to having a transition on D during the detection window. Note that the error is generated irrespective of the polarity of the transition (rising or falling). The same concept can be applied to BDM, where an error is generated only if both cases #6 and #7 take place within the same low-phase of the clock, which corresponds to having a transition on D during the detection window. Examples of the error detection in both FDM and BDM are illustrated with voltage waveforms in Fig. 3.7.

3.2.2 Timing Analyses Enabled by the TMS

The use of the TMS together with the control of the clock duty cycle enables three different modes for timing analysis. Any one of these modes can be enabled at runtime by choosing the TMS operating mode together with setting the clock duty cycle and frequency. The three timing-analysis modes enabled by the TMS presented in this section are shown in Fig. 3.8 and are described hereafter.

Setup-violation monitoring (SVM) When operated in FDM, any transition on the high phase of the clock is detected, thereby providing conventional detection of late-arriving transitions [8], as shown in Fig. 3.8(a). The TMS will detect any setup timing violation, therefore this timing-monitoring mode can be integrated with any of the previously published error-recovery techniques.

Timing-slack monitoring (TSM) When selecting BDM and setting the high-phase of the clock to be as long as or comfortably longer than the critical path of the design, the TMS can detect any increase in the critical path delay without incurring an actual setup timing-violation, as depicted in Fig. 3.8(b). The error-detection window is equal to the low phase of the clock and can be adjusted by controlling the duty cycle of the clock. The main drawback for this timing monitor mode is the speed limitation given by the timing margin, corresponding to the low phase of the clock, which needs to be added on top of the maximum operating frequency.

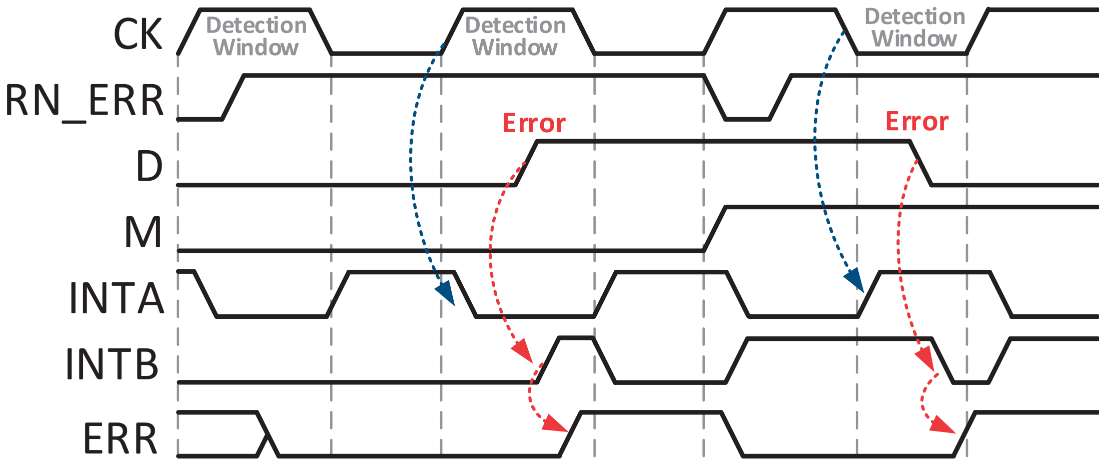


Figure 3.7: Examples of error detection in both forward-detection mode (FDM) and backward-detection mode (BDM).

In addition, the lower bound of this margin is given by the minimum low phase of the clock at which the TMS still functions correctly.

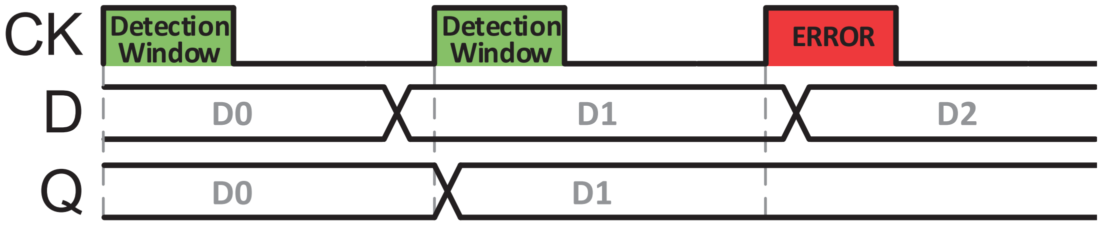
Fast-path measurement (FPM) The timing slack of paths that are significantly faster than the critical path can be measured using the TMS in BDM. In this case, the duty cycle of the clock needs to be adjusted until the high phase of the clock meets the propagation delay of the path under analysis, as shown in Fig. 3.8(c). For this operating mode, the frequency of the clock can be set according to the critical path of the design to ensure an execution free of any timing violations. The analysis of paths that are far from being critical can give insightful information about different sub-blocks in the system and it can enable adaptive dynamic voltage and frequency scaling, whenever the critical path is not excited [43, 44].

3.2.3 Simulations Results and Measurements

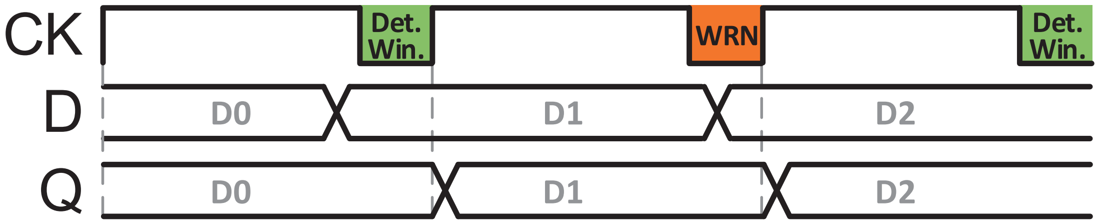
The performance of the proposed TMS designed in 28 nm FD-SOI process technology is evaluated with post-layout simulations and the results are presented in this section. The FDM and BDM of the TMS are also verified with silicon measurements of a 16-bit multiplier that was fabricated in the same technology.

3.2.3.1 Post-Layout Simulations of the TMS

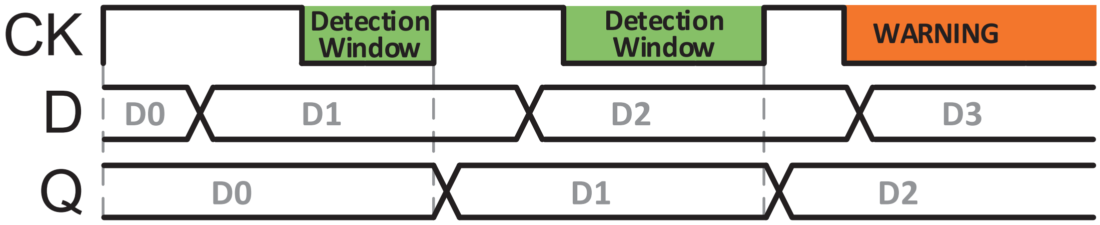
The operation of the timing-monitoring modes, presented in Section 3.2.2, requires control over the duty cycle of the clock. However, the value of the duty cycle is limited by the TMS, as well as by the other sequential elements in the design, because of the existing requirements on the minimum time length of both low and high phases of the clock. For this reason, the



(a) Setup-violation monitoring (SVM).



(b) Timing-slack monitoring (TSM).



(c) Fast-path measurement (FPM).

Figure 3.8: Timing-monitoring modes enabled by the TMS.

minimum low phase and the minimum high phase of the clock that allow correct operation of the DFF as well as of the error detector in the TMS for any timing-monitoring mode have been measured with post-layout simulations at 0.9 V, considering a typical process corner and 25°C. The results are presented hereafter.

For the conventional SVM mode, during which timing errors should be detected, a short high phase of the clock is required to relax the extra hold buffering caused by the short-path problem that typically affects EDSs [49]. In this scenario, the TMS is able to flag late-arriving transitions for a high phase of the clock as low as 90 ps (7 FO4). Considering the TSM mode which enables the measurement of the available timing slack, the detection window (low phase of the clock) represents an additional timing margin on the maximum operating frequency, which has to be minimized. Simulations show that the TMS is capable of correctly sampling the data as well as detecting transitions for a low-phase of the clock as short as 140 ps (11 FO4). The fastest path that can be measured in FPM mode is given by the minimum high phase of the clock, where the TMS is capable of sampling the data as well as detecting a transition during the low-phase of the clock. For this mode, the TMS provides correct operation for a high phase of the clock as short as 50 ps(4 FO4).

Table 3.2: Comparison between a baseline flip-flop and the TMS

	D-flip-flop	TMS	Overhead
Area [μm^2]	3.75	12.56	3.35×
Leakage current [nA]	1.61	4.21	2.62×
Clock-to-Q delay [ps]	53.45	69.79	1.31×
E_c without activity [fJ]	3.57	16.66	4.67×
E_c with 100% activity [fJ]	7.61	19.40	2.55×
E_c with 100% activity and detected error [fJ]	N/A	27.07	N/A

The overhead of the error detector in *in-situ* timing monitors is a well-researched topic and very low-complexity circuits have been proposed [7,8]. Despite the fact that the proposed TMS was not optimized for area, speed, or power, the resulting overhead compared to a baseline DFF has been measured in post-layout simulations for completeness. The results are reported in Table 3.2 where E_c is the energy-per-cycle, reported for 0% and 100% activity factors on the data input (D) of the DFF, as well as with and without an error-detection event. It is worth mentioning that, in a large design, this overhead would only affect the registers that are actually replaced with TMS cells. Therefore, the actual overhead due to the use of TMS cells highly depends on the considered design and on the number of monitored end-points.

3.2.3.2 Measurements of a Multiplier Using TMS Cells

A test structure consisting of a 16×16-bit radix-4 Booth-recoded digital multiplier with TMS cells for sampling the outputs was implemented in a 28 nm FD-SOI test-chip to verify the proposed *in-situ* timing monitor. Unfortunately, as the implemented internal clock generator on this test chip did not provide control on the duty cycle of the clock, only few basic checks could be performed. For this reason, the presented measurements are conducted using a 50% duty cycle.

At 0.7 V, the operation of the fabricated multiplier is verified up to a maximum frequency (f_{max}) of 398 MHz. When operating in SVM mode, the TMS cells start generating error signals as soon as the operating frequency is larger than f_{max}. On the other hand, when operating with a frequency close to, but not larger than f_{max}, no error signal is generated, proving the correct operation of both the FDM and the SVM mode. It is worth noting that, for the presented design, when operating *close* to f_{max}, the short-path problem [49] is not present, as the contamination delay of the multiplier is longer than half clock period. However, as expected, the generation of false error-signals can be provoked by setting the operating frequency to a much lower value than $f_{\max}$, therefore making the high phase of the clock longer than the contamination delay of the data path.

The BDM of the TMS, where the input transitions are detected on the low phase of the clock, was also verified with measurements. For this case, the test consisted in initially operating the multiplier with a relaxed clock frequency. In this condition, it ensured that all the input transitions of the TMS happened during the high phase of the clock, therefore avoiding the generation of any error signal. The clock frequency was subsequently increased up to the point where excitation of the critical path of the multiplier resulted in a transition of its end point during the low phase of the clock, which was captured by the corresponding TMS. Based on the achieved frequency and considering the 50% duty cycle, it was possible to derive the maximum operating frequency of the multiplier *without* incurring an actual timing violation. With this test, the maximum frequency of the multiplier was correctly measured with an error of less than 1%.

3.2.4 Conclusion

A TMS that is capable to monitor either dynamic timing margins or potential timing violations, therefore enabling the runtime choice of the best timing-monitoring strategy has been described in this section. The proposed circuit is capable of detecting transitions on either the high or the low phase of the clock and simulations show the TMS to operate at 0.9 V with either a high or a low clock phase as short as 90 ps and 140 ps, respectively. Both detection modes of the proposed TMS were verified on a fabricated test chip in a 28 nm FD-SOI process technology which showed the TMS to measure the maximum frequency of a digital multiplier at runtime, with an accuracy of 1%, and without incurring in any timing violation.

4 Exploiting Hardware Properties at the Algorithm Level for Energy-Quality Scaling

The proliferation of portable devices over the last decade, as well as the growing interest in future Internet of Things (IoT) applications [51], have contributed to an increasing demand for near-sensor data analysis and filtering to reduce the amount of information to be wirelessly transmitted, which is key to reduce system energy consumption [52, 53]. Due to significant increase in cost of dedicated application specific circuits in deep nanometer technologies, programmable general-purpose platforms are typically required to support the diverse requirements of different applications [54]. In this scenario, a possible solution are software-programmable ultra-low power (ULP) architectures [55] with dedicated, but reconfigurable accelerators for costly core computational kernels [56]. Programmable finite impulse response (FIR) filters are fundamental building blocks for many digital signal processing (DSP) applications, therefore they are one of the most widely implemented accelerators [57]. In addition, they are often responsible for a relatively large portion of the power in the system as they are a key kernel that might even operate continuously, for example as a key component for down-sampling (i.e., data reduction) early in the signal processing chain or as matched filter to detect wake-up events. Therefore, it can be expected that methods for reducing the power consumption of FIR filters can have a large impact on a variety of IoT systems and applications.

When operating at high clock frequencies, dynamic switching energy generally dominates power consumption. This dynamic power consumption is described by the well known equation:

$$P_{\mathrm{d}} = \alpha \cdot f \cdot C \cdot V_{\mathrm{DD}}^2, \tag{4.1}$$

where α is the switching activity factor, f is the operating frequency, C is the switching capacitance, and V_{DD} is the supply voltage. The quadratic dependency of P_{d} on V_{DD} leads to the straightforward conclusion that voltage scaling can be an efficient technique to minimize the power consumption of digital systems, albeit at the cost of a large increase in the delay of the digital gates. The clock frequency f is generally set according to the throughput requirements of the system, while ensuring that the clock period is longer than the delay of the longest timing path in the system under worst-case conditions at a given V_{DD}. If throughput requirements are sufficiently low, voltage scaling is an effective means to improve energy efficiency, due to the quadratic dependency of (4.1) on V_{DD}.

For additional power savings when real-time constraints limit the available safe range for voltage scaling, voltage over-scaling (VOS) [11–14] has been proposed for operating below the critical supply voltage and applying various techniques to handle the timing errors that may occur. Linear prediction [11], reduced-precision redundancy [12], and adaptive error cancellation [13] are such advanced low-power design techniques that enable VOS, according to the algorithmic noise tolerance approach. Despite the fact that these techniques have been shown to result in significant power savings, the majority of them implement arithmetic operations with ripple-carry adders (RCAs) which are one of the slowest implementations of adders and therefore they can dramatically limit the performance of the underlying system. Furthermore, the end points of the critical-timing paths are usually the output most-significant bits (MSBs) in RCA architectures. In this scenario, as soon as the sub-critical operating region is entered, the MSBs will be the first output bits to become unreliable, resulting in a large degradation of the overall quality even when applying a limited amount of VOS. Another approach, proposed by Whatmough *et al.* [14], proposes to operate below the critical supply voltage using a modified carry-merge adder and the critical path delays of some of the least-significant bits (LSBs) are reshaped and increased in order to limit the magnitude of the errors caused by timing violations. Even if this technique results in a small timing penalty, the performance of the filter significantly drops as soon as the point of first failure (PoFF) is passed, resulting in a small voltage guard-band when operating in the sub-critical region.

Operating below the critical supply voltage has proven to be very challenging, since it either requires slow arithmetic units (i.e., based on RCA architectures) or results in a small voltage guard-band. These properties render the use of VOS a difficult endeavour that may be too risky for today's industry. For this reason, an alternative approach to VOS is to reduce the switching activity in the digital circuits, represented by the factor α in (4.1). With this approach, additional power savings can be obtained with an acceptable and controlled degradation of the quality.

Programmable FIR filters [58, 59] are required in reconfigurable systems to provide flexibility to support a wide variety of applications, which is one of the major challenges in the IoT market [53, 54], and even different FIR kernels are often needed within the same application. In this scenario, different sets of filter coefficients defined by the various applications as well as their individual filter kernels are derived during design time of the (software) application and reassigned at runtime to the programmable FIR accelerator. The low-power technique proposed in this chapter exploits this programmability by choosing approximate values of the coefficients that limit the switching activity inside the filter, instead of always designing the filter to produce exact results, thereby achieving power savings at the expense of degraded filter quality. In addition, this approach provides the capability to scale the power consumption of the filter at runtime.

Based on the observation that most of the power consumed in FIR filters is due to multiplications, different techniques aimed to reduce power consumption in multipliers have been proposed [60–66]. These techniques include optimizing the value of successive coefficients assigned to iteratively-decomposed FIR filters based on their Hamming distance [66], the design of approximate multipliers for low-power operation [60–63], performing parallel multiplication by coefficient partitioning to enable voltage scaling [65], and reusing previously computed values through the factorization of the coefficients to reduce the complexity of the filter [64]. While these techniques can be efficient in reducing the power consumption, they all suffer from undesirable drawbacks. For example, the techniques proposed in [60, 61, 64, 65] require a very large design effort as compared to the baseline implementation of the filter, and the design in [66] only applies to iteratively-decomposed FIR filters. In addition, both the approximate multipliers [60, 61] and the reuse of partial products enabled by the factorization of the coefficients [64] can only be applied at design time for one specific filter, but they neither allow for acceleration of different filters for different applications, nor for an adjustment of the energy-quality tradeoff to scale the power consumption at runtime.

In this chapter, these limitations are addressed and a technique is proposed to perturbate the coefficients of a baseline FIR filter based on an extensive power characterization of the implemented multipliers in order to achieve dynamic power savings at the expense of a small degradation in quality. This power characterization was used to derive an algorithm that modifies the baseline filter coefficients to reduce the dynamic power consumption of the multipliers while maintaining an acceptable degradation of the filter quality. Since the proposed technique does not require any change in the design of the FIR accelerator hardware, it retains full flexibility and it allows for runtime adaptive scaling of the filter performance to trade-off power for quality.

Such adaptive scaling is first demonstrated with simulations on four different multiplier designs as baseline building blocks for FIR filters. In an example, where the maximum performance degradation is limited to 3 dB on the filter stopband, 14.6% to 25.6% power savings are achieved in the considered FIR accelerators when compared to a filter design that is agnostic of the hardware. The proposed technique was applied to two fabricated reconfigurable FIR filters, implemented as part of a 28 nm FD-SOI test chip. Silicon measurements confirmed the power benefits of the proposed technique, showing that the optimized coefficients even result in a power reduction of 33% when compared to the baseline performance[1].

Contributions The specific contributions of this chapter can be summarized as follows:

- The reliability of VOS is evaluated with silicon measurements of a multiplier, showing that the operation of a digital circuit below the critical supply voltage, where timing errors occur, is characterized by serious limitations: the application of VOS to timing-

---

[1] The achievable power savings can vary depending on the considered design of the FIR accelerator, filter specifications, as well as layout and circuit details (e.g., driving strengths and parasitics).

critical circuits results in a steep quality degradation and the quality loss is highly complex to either predict or control. Thus, an alternative approach is required to trade additional power savings for a *controlled* quality degradation.

- A cross-layer approximate computing technique on the algorithm/architecture level is proposed to reduce power consumption through switching activity reduction by carefully choosing programmable parameters of an FIR accelerator at runtime.
- An analysis of the switching activity of multipliers based on the number of non-zero generated partial-product bits is described. Furthermore, the dynamic power consumption of the considered multipliers has been characterized with accurate post-layout simulations, showing that the achievable power savings might differ depending on which of the two input ports is assigned to the constant coefficient.
- The power consumption of the multipliers contained in a programmable FIR filter is reduced by perturbing the baseline coefficients of the filter based on an extensive characterization of the implemented multiplier topology.
- The baseline performance is always ensured when operating with the reference coefficients and the power consumption of the FIR filter can be reduced at runtime, when a less accurate operation of the filter is tolerated. In addition, the implementation of the power-optimized FIR filter does not require any design overhead except for the additional memory that might be required to store the perturbed coefficients.
- The presented technique is applied to several FIR filters, demonstrating the obtainable power reductions with a controlled quality degradation. Power savings are confirmed through measurements of filters fabricated in a 28 nm FD-SOI process.

This chapter is organized as follows: Section 4.1 describes different topologies of digital multipliers. The effect of VOS on multipliers is analyzed in Section 4.2. Section 4.3 estimates the switching activity for different multiplier topologies. A post-layout characterization of the power consumed by the multipliers is presented in Section 4.4. Section 4.5 describes the proposed algorithm for perturbating the coefficients of filters and the resulting power savings vs. frequency response of the optimized filters. Section 4.6 concludes this chapter.

4.1 Digital Multipliers

Digital multipliers can be implemented with a wide range of topologies based on the desired number representation, as well as on design requirements, such as area and speed [67]. Two of the most common topologies have been considered as an example for the analysis of the internal switching activity, which is strongly related to the dynamic power consumption: the radix-2 Baugh-Wooley (BW2) multiplier [68] that presents a simple and straight-forward implementation, and the radix-4 Booth-recoded (BR4) multiplier, known for its more complex structure and high-speed performance. Both topologies have been implemented to perform

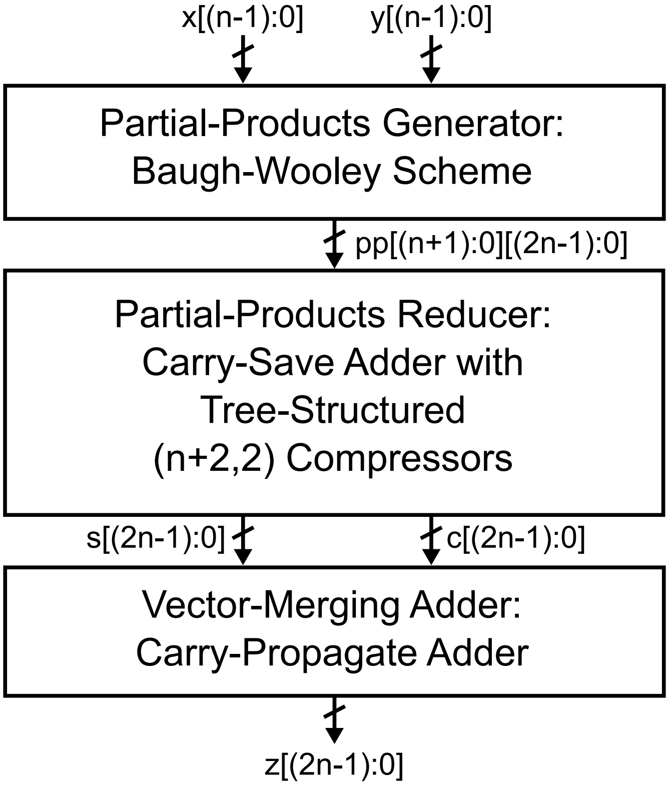

x[(n-1):0]
y[(n-1):0]
Partial-Products Generator: Baugh-Wooley Scheme
pp[(n+1):0][(2n-1):0]
Partial-Products Reducer: Carry-Save Adder with Tree-Structured (n+2,2) Compressors
s[(2n-1):0]
c[(2n-1):0]
Vector-Merging Adder: Carry-Propagate Adder
z[(2n-1):0]

Figure 4.1: Structure of a signed $n \times n$-bit radix-2 Baugh-Wooley multiplier.

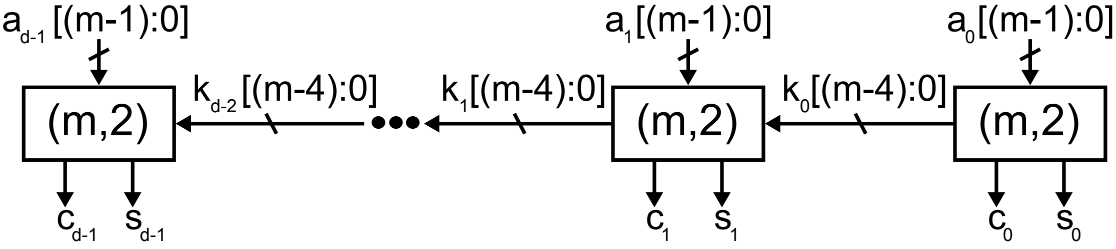


Figure 4.2: Structure of a generic partial-products reducer implemented as a carry-save adder with (m,2) compressors [68].

signed multiplication, using fixed-point two's complement as number representation. While these multipliers differ in the partial-products generator (PPG), they include the same partial-products reducer (PPR) and the same vector-merging adder (VMA), here implemented as a carry-save adder with (m,2) compressors [68] and a carry-propagate adder, respectively. The RTL representations of both the BW2 multiplier and of the carry-save adder with (m,2) compressors have been taken from the VHDL Library of Arithmetic Units proposed in [68]. The considered topologies are reviewed with more details in the following subsections.

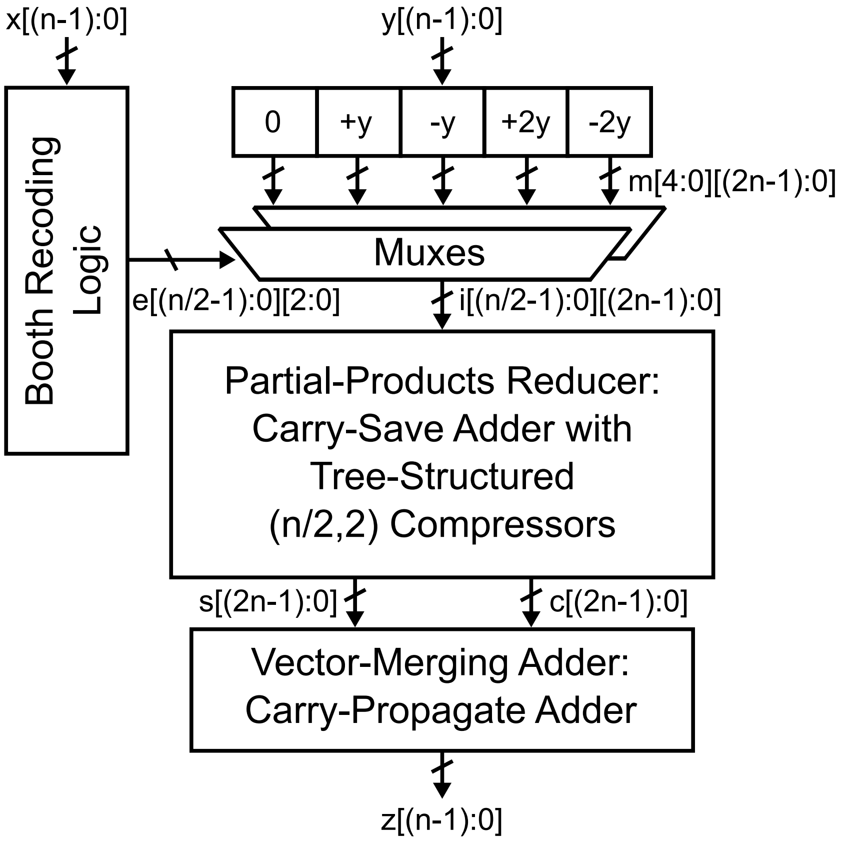


Figure 4.3: Structure of a signed $n \times n$-bit radix-4 Booth-recoded multiplier.

4.1.1 Radix-2 Bough-Wooley Multiplier

The BW2 multiplier is a simple structure which can achieve medium operating speed with moderate silicon area. Fig. 4.1 illustrates the operating principle of this topology. Two n-bit input operands are passed to the PPG that implements the Baugh-Wooley scheme [67, 68], which feeds the PPR implemented as a carry-save adder with (m,2) compressors [68] as shown in Fig. 4.2. For the considered implementation, the half adders (HAs) and full adders (FAs) instantiated inside each of the compressors are connected in a tree structure to reduce the critical path of the PPR. The sums and carries of the PPR are then passed to the VMA implemented as a carry-propagate adder that provides the final result of the multiplication.

4.1.2 Radix-4 Booth-Recoded Multiplier

Since radix-2 Baugh-Wooley multipliers are rather slow, a fast multiplier that uses Booth recoding is also considered. A BR4 multiplier, shown in Fig. 4.3, has been considered, where the PPR is fed with less than half of the partial products of those in the BW2 multiplier, thereby providing a much shorter critical path. As opposed to the symmetric BW2 multiplier, in the BR4 topology, the two input operands are processed differently, since x is passed to the recoding logic that decides which multiples of y should be fed to the PPR. For the considered implementation, a carry-save adder with (m,2) compressors [68] is used for the PPR and a carry-propagate adder is used for the VMA, as in the BW2 multiplier.

4.2 Voltage Over-Scaling

Voltage scaling is one of the most effective as well as popular techniques to reduce the power consumption in digital circuits due to the quadratic relationship betweeen dynamic power consumption and supply voltage, as previously shown in (4.1). To achieve even higher power savings, voltage over-scaling (VOS) has been proposed [11–14]. The basic idea of VOS is to operate digital circuits at a sub-critical supply voltage, where timing errors might occur. In this section, the potential benefits as well as the limitations of VOS are analyzed by considering its effects on a multiplier.

4.2.1 Overclocking a Multiplier

For this analysis, a 16×16-bit BR4 multiplier is considered as it is one of the most popular architectures due to its capability of running at high clock frequencies. In particular, the effects of VOS are evaluated by overclocking the multiplier as any increase on the operating frequency can be traded for power savings through VOS at a constant throughput[2]. Both inputs and output of the considered multiplier are sampled by sets of registers which are interfaced to additional testing circuitry that is designed to apply arbitrary input operands and to read the output of the multiplier without incurring additional timing violations after the sampling of the multiplier output even when the multiplier operates in the sub-critical range. The multiplier as well as the testing circuitry are fabricated on a test chip in 28 nm FD-SOI and the results presented in this section are based on the corresponding silicon measurements.

The maximum operating frequency that ensures an error-free operation of the multiplier ($f_{\max}$) is initially measured for a nominal voltage of 0.7 V. This measurement is performed by applying a set of 1k pairs of random operands to the inputs of the multiplier and by verifying the correctness of the output. It is worth noting that the excitation of any path within the multiplier depends not only on the operands that are computed within one clock cycle but also on the pair of operands that were multiplied in the previous cycle. For this reason, the set of previously-computed operands were also randomized for each of the 1k test sets. Results show that the considered multiplier correctly computes all the operands for an $f_{\max}$ of 256 MHz.

The same random sets of 1k operands have been applied and the sampled multiplier output have been checked when operating the multiplier at a clock frequency higher than $f_{\max}$. Results are shown in Fig. 4.4 where the considered frequency overclocking ranges from 0% (256 MHz) to 50% (384 MHz). The figure reports the reliability of each of the 32 output bits of

---

[2] The behaviour of the over-clocked multiplier is used to predict the behaviour of the multiplier at sub-critical voltages as it simplifies the measurement procedure.

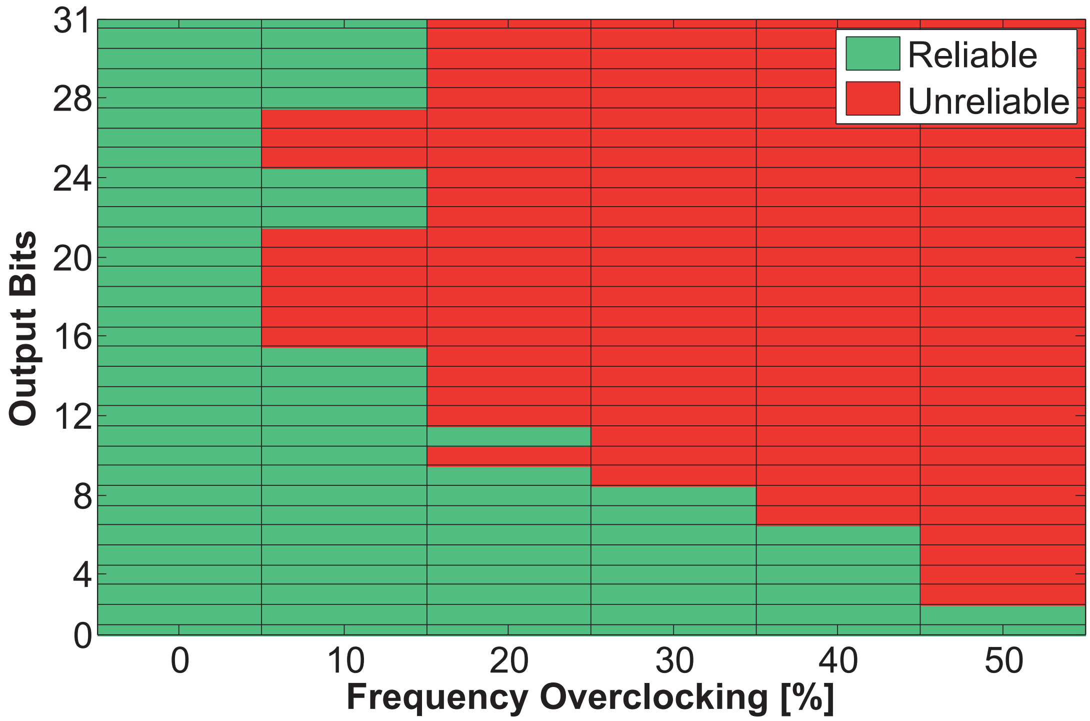


Figure 4.4: reliability of the 32 output bits of a 16×16-bit BR4 multiplier when overclocking is applied.

the multiplier for the different frequencies. For this analysis, an output bit is considered to be *unreliable* as soon as, for any of the computed operands, it differs from the expected value (i.e., bit flip) due to overclocking. In addition, once an output bit becomes unreliable for a given frequency, it is also considered to be unreliable for any higher frequency.

It is worth noting that even if the testing circuitry reliably monitors when the output registers fail to sample the expected result of the multiplication due to overclocking, it does not capture when the setup constraint *starts* to be violated. In fact, when entering the overclocking region, the setup contraint is initially violated for a small amount of time, resulting only in an increased clock-to-output delay of the sampling register while the input data is still correctly captured and latched. However, the delay increase associated with this onset of a setup violation affects the subsequent paths that have the output of the considered register as start point and therefore it might cause further timing violations. As any additional timing violation highly depends on the design of the subsequent stages of the pipeline, it is out of the scope of the analysis presented in this thesis. Thus, the designed testing circuitry only monitors the effect of larger overclocking on the output registers beyond the point where they fail to sample the correct result of the multiplication.

Two main considerations can be derived from the measurements of the overclocked multiplier reported in Fig. 4.4. First, the majority of the 32 output bits of the multiplier become unreliable

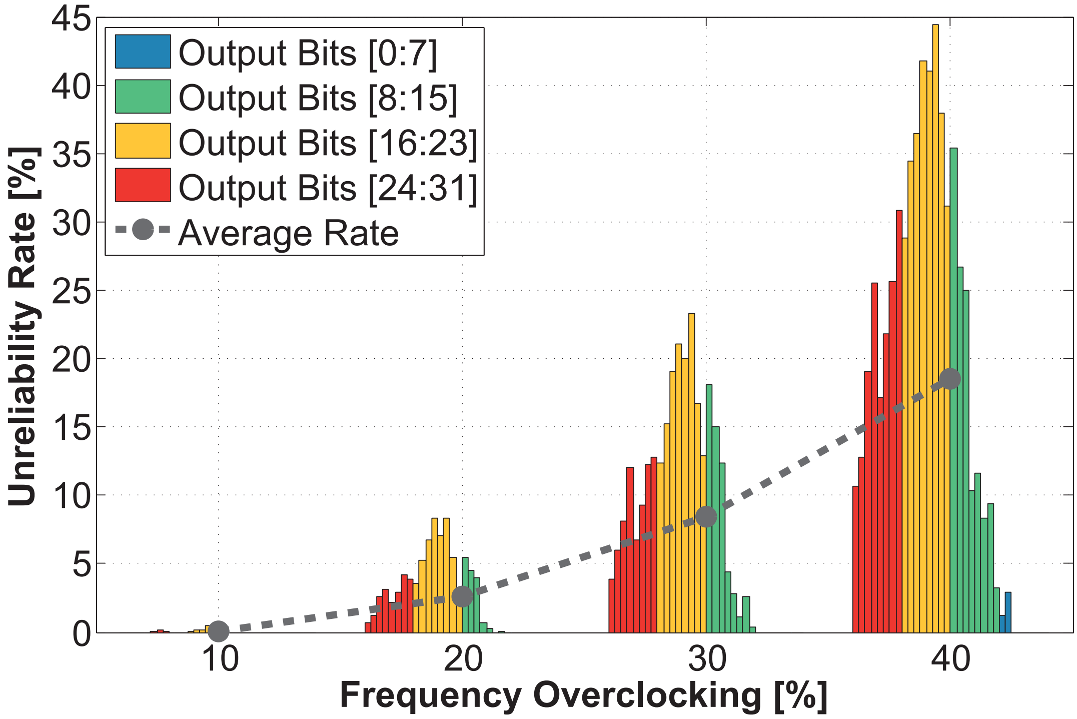


Figure 4.5: Reliability rate of the 32 output bits of a 16×16-bit BR4 multiplier when overclocking is applied.

as soon as overclocking is applied. In particular, the unreliable bits are 28% and 68% of the total output bits for a frequency increase beyond f_{max} of only 10% and 20%, respectively. Second, few of the LSBs are actually even more reliable than the remaining bits at high overclocking rates. For example, the first 6 LSBs (corresponding to only 18% of the output bits) are reliable up to 40% overclocking. This phenomenon can be explained considering that each of the output bits is an end point of paths that fit into one of the three categories described below:

- *Optimized slow paths*: these are slow paths that have been optimized by the timing engine of the synthesis software to reduce their delay and meet the setup timing constraint, therefore they determine the critical path of the design.
- *Unoptimized slow paths*: slow paths that are *close* to become critical, but are generally not optimized during synthesis as a reduction of their delays would not decrease the critical delay, but would only increase complexity and energy consumption of the synthesized circuit.
- *Unoptimized fast paths*: as for the unoptimized slow paths, fast paths do not require any delay optimization as they are far from being critical.

Any tree-structured multiplier is characterized by a large number of both optimized and unoptimized paths with delays close to be critical and this causes the majority of the output bits to become unreliable as soon as overclocking is applied, as shown in Fig. 4.4. However, few of the output LSBs are end points of fast paths, as they are computed by low-complexity circuits that are composed of only few gates and therefore they are more resilient to overclocking. Overall, the degradation on the output quality is significant even for small overclocking rates, as the MSBs are among the first output bits to fail.

Even though the results in Fig. 4.4 provide a picture of the multiplier reliability when overclocking is applied, any effect caused by data dependency is hidden as the worst-case result among the 1k sets of input operands is always considered. In fact, when considering all the paths sharing the same output bit as end point, there might be sets of input operands that excite only the fastest paths within this group or that might not even excite any path at all. This effect is shown in Fig. 4.5 where the *probability* of an output bit to be unreliable is reported against the amount of overclocking.

On average, the unreliability rate of the 32 output bits is 0.1% but it quickly rises up to 19% for an overclocking rate of 10% and 40%, respectively. For a given overclocking rate, the output bits show different unreliability rates as the probability of exciting paths beloging to an end point as well as the delay of the excited path change with the position of the considered output bit. For example, the worst reliable bit reaches an unreliability rate of 9%, that is three times the average, for an overclocking rate of only 20%.

4.2.2 Limitations of Voltage Over-Scaling

Based on the analysis of the presented measurements, the potential limitations of VOS are summarized as follows:

- The application of VOS to tree-structure and timing-critical circuits (e.g., high-speed multipliers) results in a steep degradation of the quality of service (QoS) as soon as the sub-critical operating region is entered. This is mainly due to the large amount of near-critical paths in these designs.

- Potential candidates that might benefit more from VOS are RCA-based arithmetic operators as they include a large variety of slow and fast paths within their path-delay profile. However, as they rely on one of the slowest implementations of adders, this design choice can drastically limit the maximum operating frequency of the system in which they are integrated and VOS can hardly overcome this initial speed penalty.

- Predicting the reliability of a digital circuit (e.g., a multiplier) under timing errors is highly complex as the excitation of timing-critical paths highly depends not only on the current value of applied inputs (e.g., operands) but also on the previously-applied values. This peculiarity largely increases the number of possible conditions on the inputs, therefore increasing the complexity of any attempt to predict the output under timing violations for QoS analysis and limiting our ability to shape the input profile to minimize the occurrence of timing violations.

- When applying VOS, the violation of a setup constraint in a register might result in an increase of its clock-to-output delay. As this phenomenon affects the subsequent paths where the start point coincides with the output of the register, other timing violations might occur. The effect of these additional timing errors is very complex to predict as it highly depends on the timing profile of the subsequent paths.

For all the reported considerations, operating a digital circuit below the critical supply voltage where timing errors occur results to have several critical limitations. Thus, beside the application of conventional voltage scaling, an alternative approach is required to enable additional power savings for a *controlled* degradation on the QoS.

4.3 Switching Activity in Multipliers

Another approach for achieving savings on the dynamic power consumption of digital circuits is to reduce their switching activity. In order to achieve this, part of the design can be silenced by carefully choosing the applied inputs. To this end, the switching activities of the BW2 multiplier and of the BR4 multiplier are first analyzed in this section.

4.3.1 Radix-2 Bough-Wooley Multiplier

Considering the structure of a generic multiplier, the PPR is usually one of its most complex parts which generally consumes the largest amount of power. For this reason, in order to reduce the power consumption of the BW2 multiplier, the switching activity of the implemented PPR is analyzed at first.

The primary operation of the PPR is to shift and add the partial products. This multi-operand addition is generally implemented with HAs and FAs as the main building blocks. The switching activity of these gates can be reduced by increasing the probability of having stable logic-zeros at their inputs, which corresponds to forcing more partial-product bits to zero. This can be achieved by providing a large number of bits that are equal to zero in at least one of the two input words of the multiplier. For example, forcing the LSB of one input operand of the 4-bit BW2 multiplier, shown in Fig. 4.1, to zero would result in 15% of the generated partial-product bits and correction terms to be equal to zero, regardless of the other input. For this reason, any circuit where one of the multiplier inputs is fixed over a considerable number of cycles and where some flexibility is available to choose this input (with moderate QoS degradation) provides potential for power savings.

Fig. 4.6 plots the number of generated partial-product bits equal to one in a 4×4-bit Baugh-Wooley PPG for all possible input combinations. This plot provides a means to identify

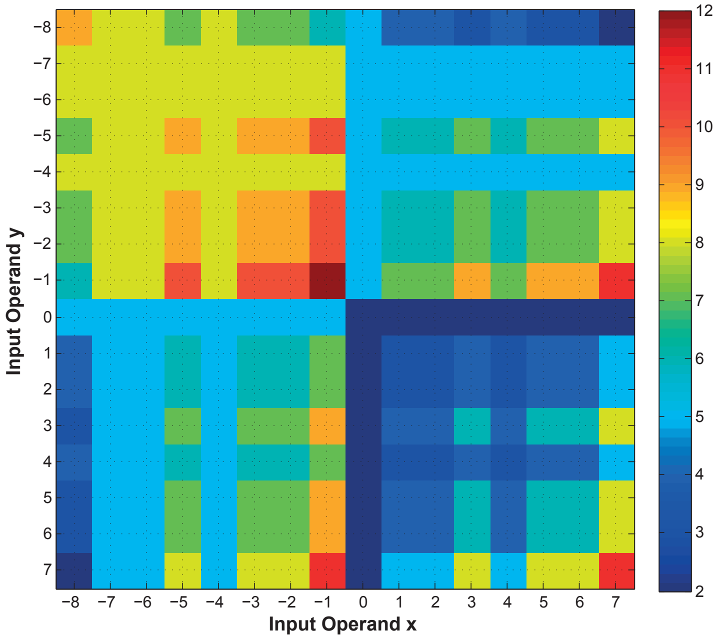


Figure 4.6: Number of the generated non-zero partial-product bits in a signed 4×4-bit radix-2 Baugh-Wooley multiplier for all input combinations.

operand values that ensure – on average – a limited number of non-zero partial-product bits, and therefore, low switching activity, regardless of the value of the other operand. To emphasize this point, an input with the value 0 (0000 in binary notation) would ensure low power consumption for any value of the other input operand, whereas the value −1 (1111 in binary notation), which is only one LSB distant from zero, generates several non-zero partial-product bits resulting in a much larger power consumption. In an application that can tolerate approximations, this property can be exploited to carefully choose constant multiplier coefficients to reduce power consumption at runtime.

To evaluate the potential of the idea, the switching activity is evaluated while one input operand is held constant and the other input operand receives random data. We shall refer to the former as the *coefficient operand* and to the latter as the *data operand*. Specifically, an 8×8-bit BW2 multiplier is considered, where the input x is defined to be the coefficient operand and y is defined as the data operand. In a first step, the average number of non-zero partial-product bits per coefficient operand is considered, while assigning any possible value to the data operand, as plotted in Fig. 4.7. It is worth noting that the number of non-zero partial-product bits in the BW2 multiplier does not depend on which input port receives the constant coefficient operand, and therefore, the graph obtained when choosing y as the constant operand would be identical. From Fig. 4.7, it can be noticed that the difference in the number of non-zero partial-product bits can be significant even for two coefficient operands that are almost equal. Hence, we can conclude that a small variation on the coefficient operand might result in large power savings in the adder tree that sums up the partial products.

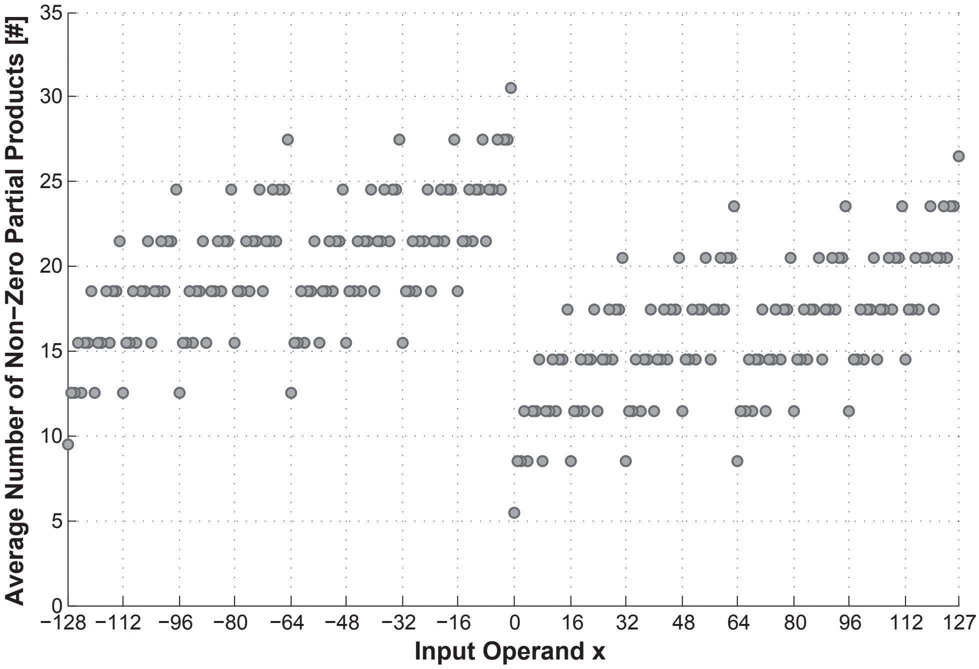


Figure 4.7: Average number of the generated non-zero partial-product bits per input operand in the 8×8-bit radix-2 Baugh-Wooley multiplier.

4.3.2 Radix-4 Booth-Recoded Multiplier

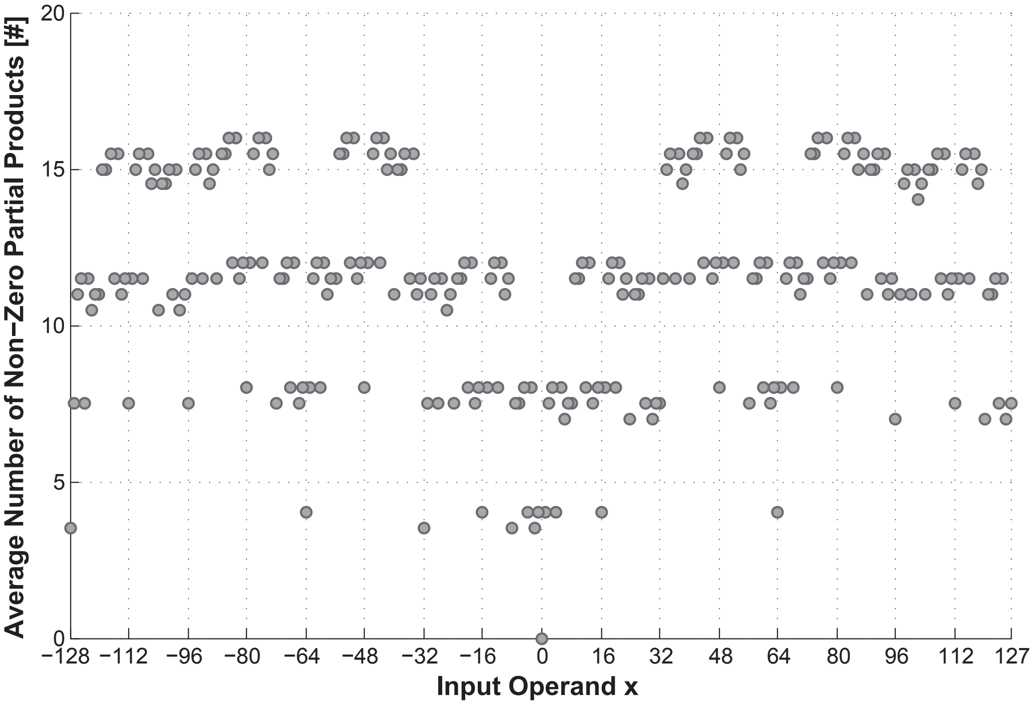
The non-zero partial-product bits per input operand analysis on the PPG, previously shown for the BW2 multiplier, was also applied to the 8×8-bit BR4 multiplier. The average number of non-zero partial-product bits per input operand is shown in Fig. 4.8(a) and in Fig. 4.8(b) for inputs x and y used as the constant coefficient operand, respectively. As expected, Fig. 4.8(a) and Fig. 4.8(b) differ due to the asymmetric nature of this topology. However, while this analysis provides some insight toward the switching activity of the multiplier, this metric does not include the activity of the PPG, which, in the case of the BR4 topology, is a more complex structure than the radix-2 Baugh-Wooley PPG as it includes both the Booth recoding logic and the logic that produces the multiples of y. Therefore, further means of analysis, such as the gate-level simulations provided in Section 4.4, are essential to obtain a better estimation of the multiplier power consumption.

4.4 Gate-Level Characterization of Multipliers

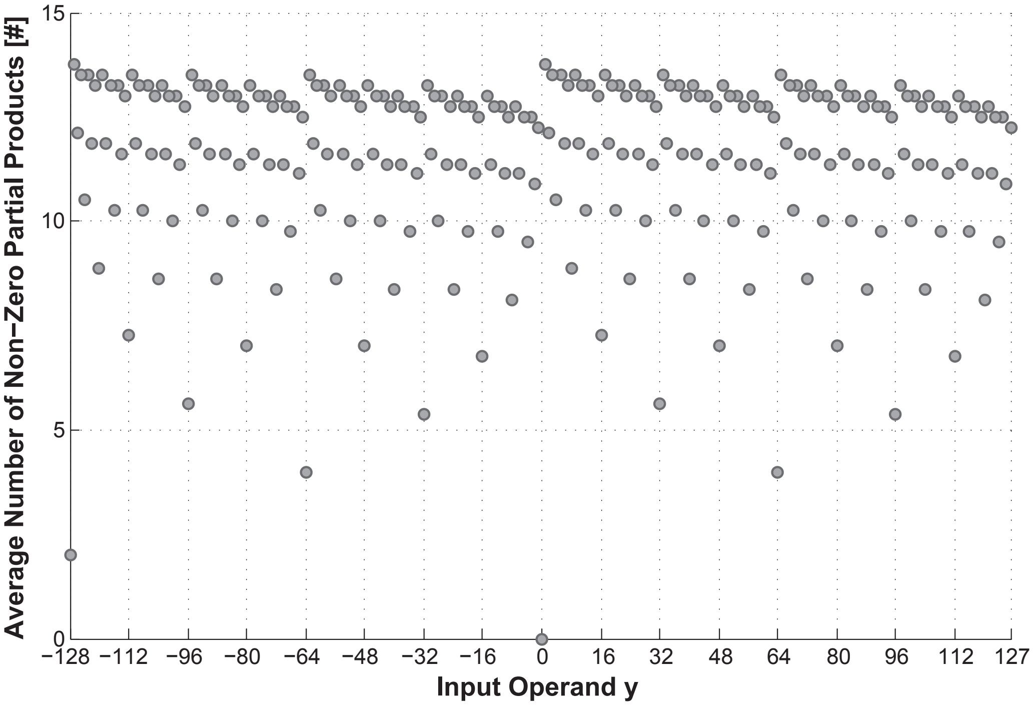
The analysis presented in Section 4.3 provides the motivation for carefully choosing and adjusting constant coefficients[3] for a multiplication by analyzing the number of non-zero partial-product bits that are created by each possible choice of the coefficient operand. How-

---

[3] Coefficients are considered to be *constant* as they do not change for an extended period of time, but they are still fully programmable.



(a) The coefficient operand is the input x



(b) The coefficient operand is the input y

Figure 4.8: Average number of the non-zero partial-product bits in an 8×8-bit radix-4 Booth-recoded multiplier.

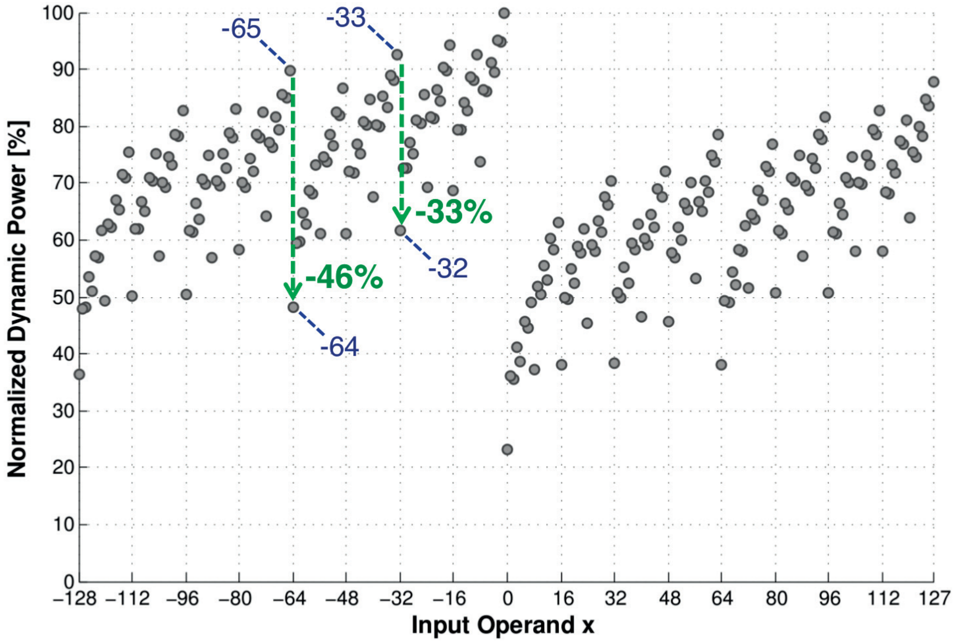


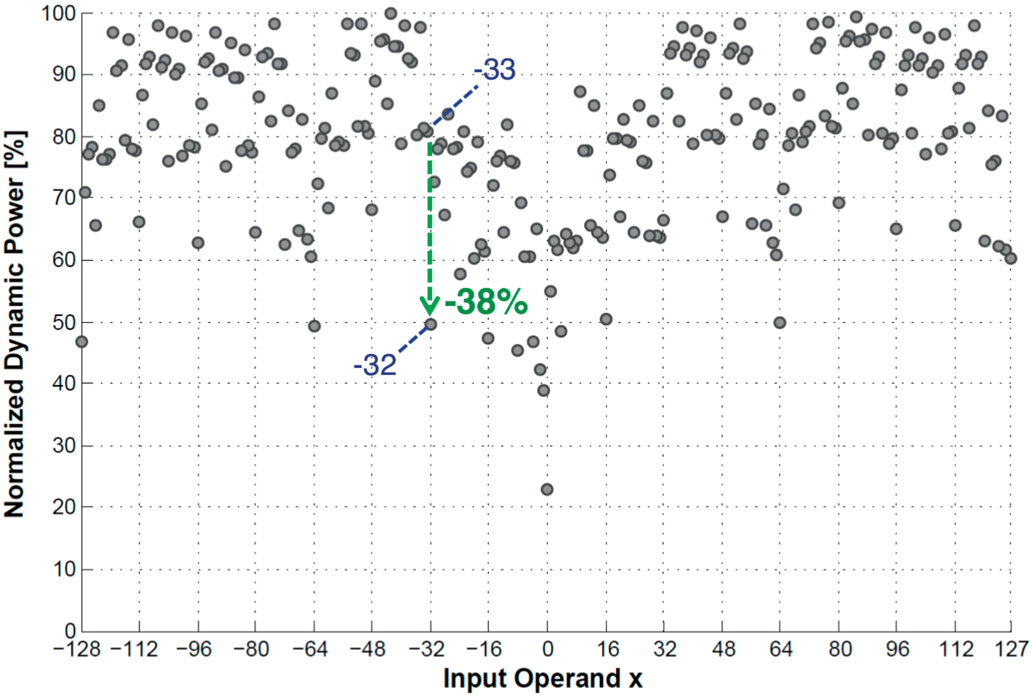
Figure 4.9: Normalized dynamic power consumption per input operand x in the 8×8-bit radix-2 Baugh-Wooley multiplier.

ever, this first-order approach provides only a first-order estimate of the overall dynamic power consumption. For a more accurate analysis, especially when considering more complex topologies, such as a Booth-recoded multiplier, gate-level simulations are required to arrive at an effective choice of the constant coefficient operand. In addition, for multiplier topologies with asymmetric timing paths, where the input operands excite paths having different delays, the choice on which operand is assigned to the coefficients should also be made based on the results given by a timing analysis, to avoid any speed limitation.

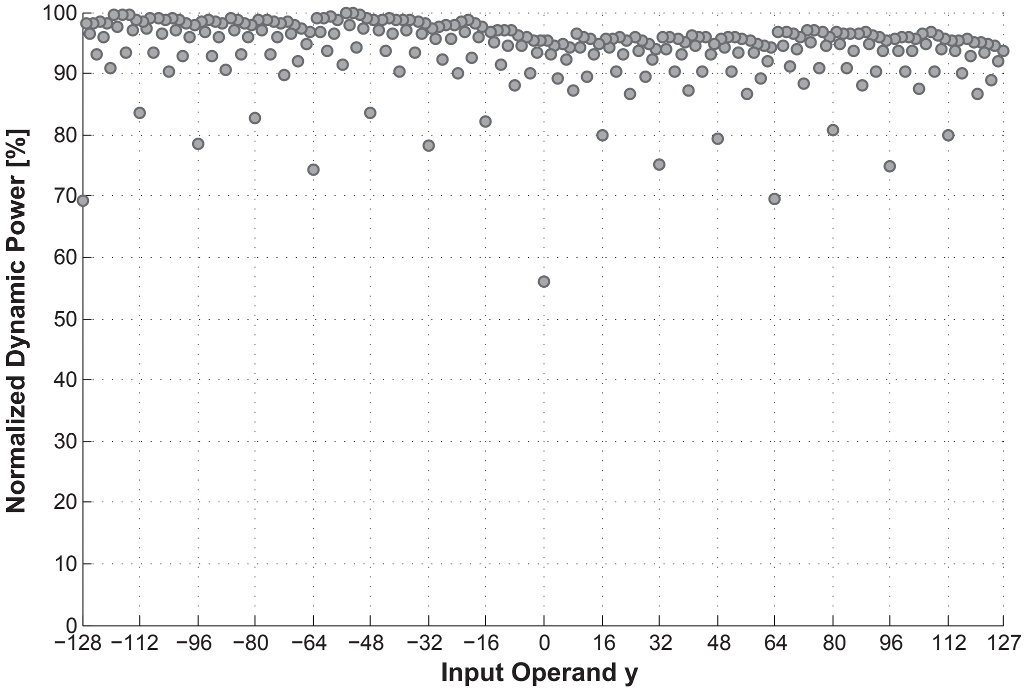
The following subsections present power and timing analyses on gate-level implementations of the BW2 and BR4 multipliers, introduced in the previous section. The multipliers were synthesized and mapped to a 28 nm FD-SOI standard cell library using Synopsys Design Compiler and placed-and-routed using Cadence Encounter. Considering the low-power design space, synthesis and automated place-and-route were performed at a near-threshold supply voltage (V_{DD}) of 600 mV. The presented data for power and timing was extracted from post-layout implementations based on dynamic vectors using Cadence Tempus and Voltus.

4.4.1 Power Characterization

In order to characterize the power consumption of a multiplier implementation, a constant value was applied to the coefficient operand input, while a sequence of 1000 independent uniformly distributed random values were assigned to the data operand. All 2^n coefficient values were considered for each $n \times n$-bit multiplier, and the dynamic power consumption was



(a) The coefficient operand is the input x



(b) The coefficient operand is the input y

Figure 4.10: Normalized dynamic power consumption in an 8×8-bit radix-4 Booth-recoded multiplier.

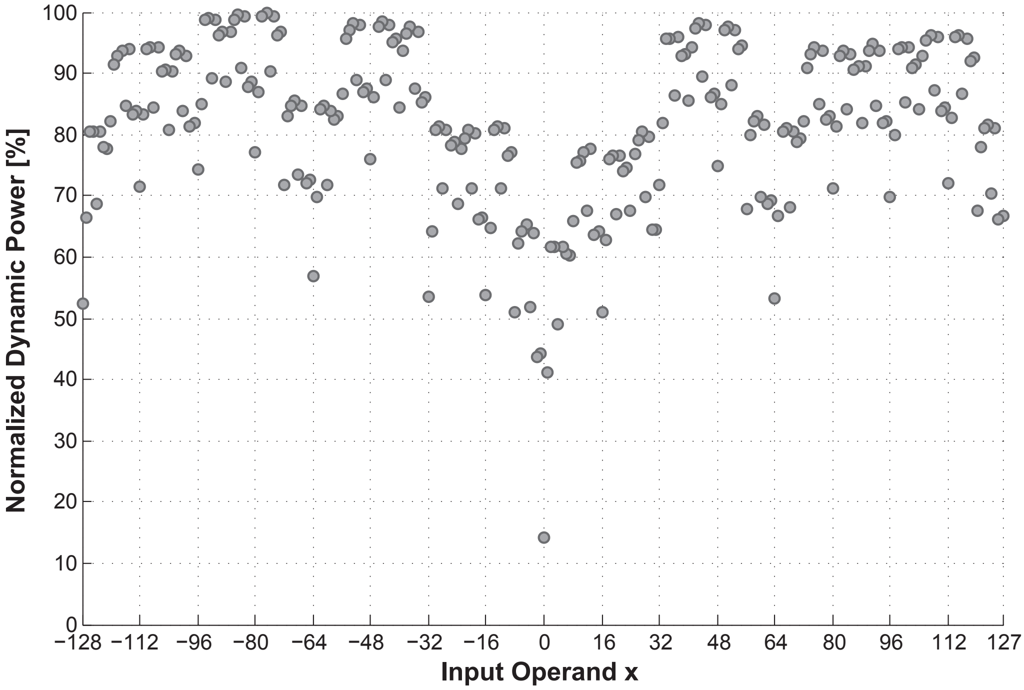
extracted for each coefficient operand value. The presented power analyses consider 8×8-bit multipliers; however, larger bit-widths were also tested and provided similar results.

Due to the symmetric structure of the BW2 multiplier, the power characterization is almost identical when either port is used for the coefficient operand. Hence, only results for x as the coefficient operand are reported for this topology. Fig. 4.9 plots the normalized power consumption of the 8×8-bit BW2 multiplier as a function of the value of the coefficient operand. As expected, the results from the power simulations follow the same profile as the number of non-zero partial-product bits shown in Fig. 4.7. This confirms that in BW2 multipliers the largest amount of power is consumed in the PPR and that the perturbation of the constant coefficients to values that result in a low percentage of non-zero partial-product bits can be an effective way to reduce dynamic power consumption. For example, by choosing the approximated value of –65 for the coefficient operand instead of –64, over 40% of the dynamic power can be saved.

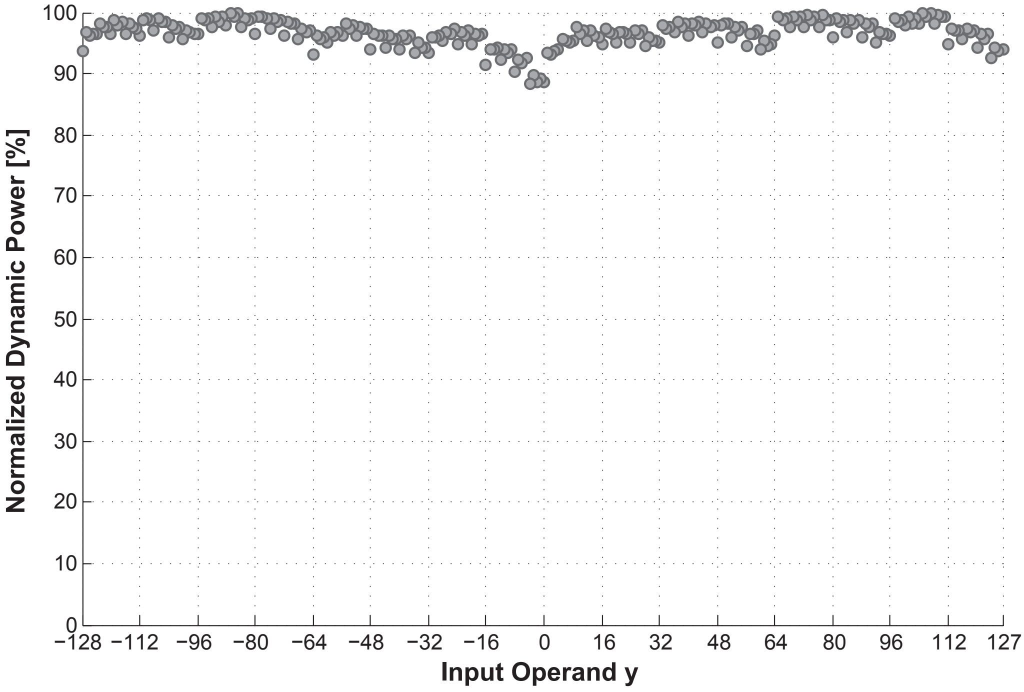
When applying a similar analysis to the BR4 multiplier, the two architectural choices where the x and y ports are connected to the constant coefficient operand separately must be considered, due to the asymmetric nature of the gate-level implementation. The two corresponding normalized dynamic power consumptions per input operand are provided in Fig. 4.10(a) and Fig. 4.10(b) for the x and y ports chosen as coefficient operands, respectively. The results show that, when choosing port x for the constant coefficient operand, the potential for power savings from approximations of this coefficient is significantly higher than when choosing port y for the constant coefficient operand. This behavior is due to the asymmetric nature of the BR4 multiplier, as shown in Fig. 4.3. In particular, some recoded values of x can cause a large number of the operands summed in the PPR to be equal to zero, which significantly reduces the power consumption since the PPR is one of the main contributors to the power consumed by the entire multiplier. Furthermore, if x is constant, the Booth recoding logic does not present any switching activity. On the other hand, when y is kept constant, little power variation is observed. This is due to the fact that the logic that produces the multiples of y does not consume a significant amount of power and there are no specific values that ensure low switching activity in either the Booth recoding logic or in the PPR[4]. Having identified the input x as the port that enables the largest power savings, constant coefficient operands should be always assigned to this port. For example, whenever the proposed technique is applied, the perturbation of the coefficients assigned to the input x gives over 30% of dynamic power savings approximating –33 with –32. Interestingly, it is noted that even if the proposed technique is not applied, x is the preferred choice for the constant coefficient as even the average power consumption over all coefficient choices is 5% below that of a design where y is constant.

---

[4] As expected, this property is not represented by the analysis on the switching activity of Section 4.3.2, since only the non-zero partial-product bits provided to the PPR were considered. Therefore, for complex topologies, such as the BR4 multiplier, more accurate power simulations are required for a proper evaluation of the possible power savings.



(a) Input operand x



(b) Input operand y

Figure 4.11: Normalized dynamic power consumption per input operand for the 8×8-bit multiplier provided by the synthesis software.

Table 4.1: Propagation delays of the 8×8-bit multipliers

Multiplier Topology	Delay from x [ns]	Delay from y [ns]
Radix-2 Baugh-Wooley	1.151	1.152
Radix-4 Booth-Recoded	1.091	1.096
DesignWare Library	0.951	0.935

For completeness, the same type of analysis is also applied to the multiplier topology automatically chosen during synthesis with stringent timing constraints from the Synopsys DesignWare Building Block IP library 2013.12. The normalized dynamic power consumptions of this reference topology are plotted in Fig. 4.11(a) and Fig. 4.11(b) for the ports x and y chosen for the constant coefficient operand, respectively. While the exact topology or implementation methodology chosen by the synthesis tool is not exposed to the user, the resulting power profiles are similar to the ones shown in Fig. 4.10. Thus, we conclude that this topology is most likely very similar to the considered custom implementation of a BR4 multiplier.

As the value of the constant coefficient operand can significantly reduce the dynamic power consumption of the multiplier depending on the *percentage* of the produced non-zero partial-product bits, it is worth considering whether the *position* of the partial products that are equal to zero would impact the power consumption as well. The dynamic power consumption of multipliers with a linear structure (*e.g.*, with PPR implemented as a RCA) is actually affected by the position of the non-zero partial-product bits as the switching activity of the lower levels of the PPR highly depends on the switching activity of the logic that appears early in the chain. However, in this thesis, the PPR of the considered BW2 and BR4 multipliers is based on a carry-save adder with tree-structured (m,2) compressors where all partial products enter the adder tree on the same level. For this reason, the impact on the switching activity is almost equal for all partial products. In fact, for tree-structured multipliers, the number of (*subsequent*) zero partial products has a larger impact on the dynamic power consumption compared to their position since only subsequent zero partial products silence entire subtrees.

4.4.2 Timing Analysis

When considering the timing of a multiplier where one of the coefficient operand is held constant during data processing, only the delay paths from the data operand to the output of the multiplier are relevant for the determination of the maximum operating frequency[5]. Depending on the considered multiplier topology, timing asymmetry might exist, and therefore, the multiplier may be characterized by different delays depending on which input port is connected to the constant coefficient operand and which is connected to the data operand.

---

[5] Note that proper timing measures must be taken to avoid timing violations when coefficients are changing during operation.

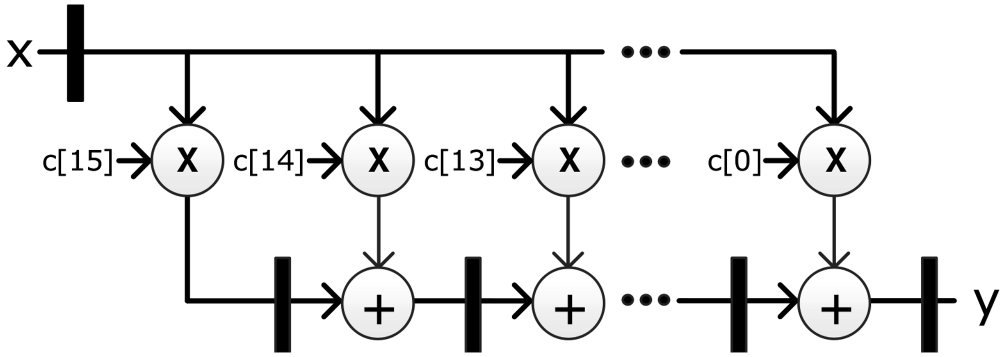


Figure 4.12: Block diagram of a direct-form FIR filter with 16 taps.

Clearly, from a timing perspective, the port that contains the critical path of the multiplier should be assigned to the constant coefficient operand to reduce any speed limitation for the data operand. However, as previously described, the potential for runtime power savings may also depend on which port is assigned to the constant coefficient operand. Hence, a timing analysis is required to verify the presence of any timing asymmetry.

To carry out the suggested timing analysis on the considered multiplier topologies, synthesis was performed with an over-constrained timing target to achieve the lowest possible propagation delay from each of the inputs to the output. STA was used to extract the worst delays through each multiplier and from each input. The extracted delays are provided in Table 4.1. The results show very similar delays through operands x and y for each of the considered multiplier topologies. This analysis was repeated while constraining the synthesis tool to independently optimize the x and y inputs and for various (larger) bit-widths; however, similar results were obtained. From this analysis it is possible to conclude that the port for the constant coefficient operand can be chosen based only on the maximum achievable power savings.

4.5 Application to an FIR Filter

In the following, the observations from the previous sections are applied to the example of a programmable FIR filter accelerator, where filter coefficients usually remain constant over a large number of cycles, while input data changes in each cycle. A direct-form FIR filter with programmable coefficients is considered and its block diagram is shown in Fig. 4.12. When designing an FIR filter, several metrics and specifications are taken into account in order to choose the optimal coefficients. However, in most cases, the impact of the choice of the filter coefficients on power consumption is not considered. Since the coefficients are set as constant operands to the multipliers that are used to construct the filter, the power consumption of the filter may strongly depend on the choice of these coefficients. Fig. 4.13 plots the dynamic power consumption of all the 8×8-bit multipliers contained in a reconfigurable 16-tap FIR filter designed with different cutoff-frequency constraints and with windowing methods. This

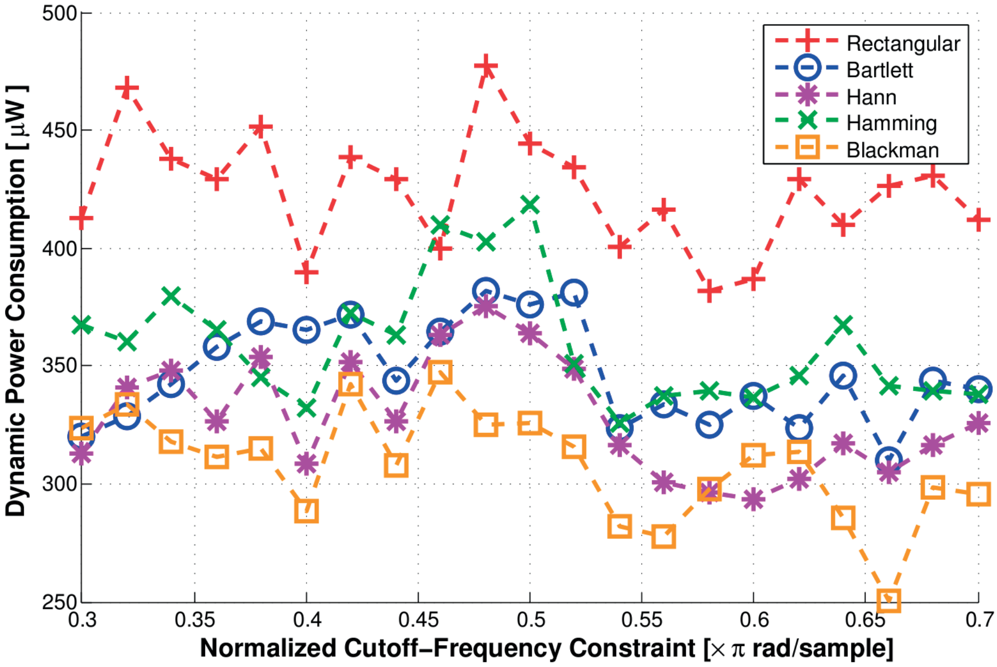


Figure 4.13: Dynamic power consumption of the 8×8-bit multipliers implemented in a reconfigurable 16-taps FIR filter designed with different cutoff-frequency constraints and windowing methods.

straightforward initial experiment illustrates how the dynamic power consumption significantly varies, even for small changes in the design constraints as well as even with the choice of the design method (windowing) used to derive the filter coefficients. This small experiment suggests that proper power-aware approximation of coefficients bares significant potential for runtime power savings.

The power-optimization technique proposed in the following, systematically utilizes the power characterization of the multipliers to perturbate the coefficients of a reference FIR filter to trade-off reduced power consumption for quality. The presented approach does not require any design overhead, since only the choice of runtime programmable coefficients is modified in a programmable FIR accelerator macro. It is worth mentioning that whenever the filter coefficients are not required to be programmable, they can be hardcoded and more area-efficient and more energy-efficient implementations of the FIR filter can be realized. In such scenarios that do not require any flexibility, the multiplications can be performed efficiently in a direct-form FIR filter by optimized shift-and-add multipliers [69]. The presented analysis does not apply to this case as the synthesis tool would optimize the produced netlist according to the hardcoded coefficient, therefore providing different hardware implementations for different sets of coefficients. When still considering coefficients that are not supposed to change during runtime, the proposed power-optimization technique does neither apply to iterative decomposed FIR filters which are implemented with one multiply-and-accumulate (MAC) unit [69]. As the focus is on programmable FIR filters as generic accelerators for

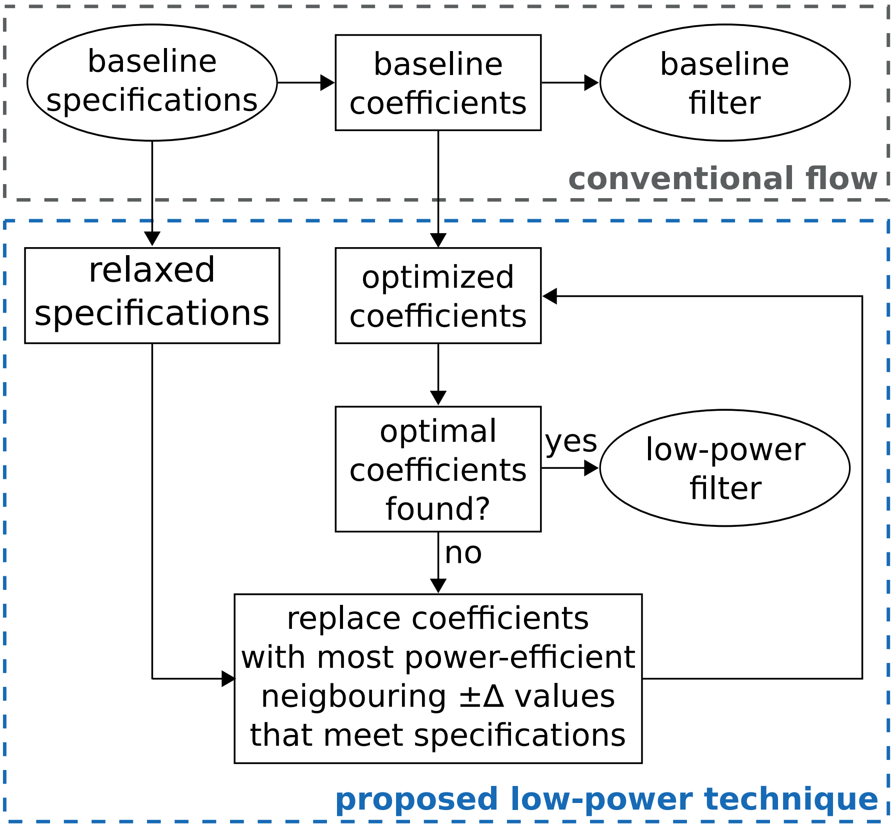

baseline specifications
baseline coefficients
baseline filter
conventional flow
relaxed specifications
optimized coefficients
optimal coefficients found?
yes
low-power filter
no
replace coefficients with most power-efficient neigbouring ±Δ values that meet specifications
proposed low-power technique

Figure 4.14: Flowchart of the conventional flow and proposed technique for the design of the filter.

various applications, the optimization algorithm of the proposed technique as well as the simulations results given by its application on four programmable FIR filters are presented in the subsections that follow.

4.5.1 Optimization Algorithm

To derive a set of coefficients that are optimized for low power consumption, the algorithm proposed in the flowchart of Fig. 4.14 is used: all the design requirements, such as passband ripple (A_p), stopband attenuation (A_s) and passband frequency edge (F_p) initially define the coefficient values of the reference filter. These initial coefficients satisfy the baseline specifications and provide an estimation of the power consumed by the multipliers using the results provided in Section 4.4.

With the coefficients of the reference filter as a starting point, the coefficients are perturbed to drive the multipliers towards lower power consumption. In particular, at each optimization step, the algorithm considers a range of values around each of the coefficients and the value that results in the lowest power consumed by the multiplier is preferred. For the examples, shown in this chapter, the considered range of values ($\pm\Delta$) is defined manually through

Table 4.2: Specifications and power savings of the low-pass FIR filters

A_s^a [dB]	F_t^a [$\pi \frac{\text{rad}}{\text{sample}}$]	Multipliers Bit-Width	Taps	S_M^b [%]	S_F^b [%]
-30	0.2	8×8	16	18.0	14.7
-40	0.2	10×10	16	24.2	19.5
-30	0.1	8×8	32	24.2	20.2
-40	0.1	10×10	32	14.6	11.7

[a] Design specifications of the baseline filter.

[b] Obtained when tolerating a 3 dB error on the stopband attenuation.

experiments and it is kept constant for all optimization steps in the algorithm for the derivation of the coefficients. While the algorithm is rather insensitive to the choice of Δ, a sufficiently large Δ value is only required to avoid local minima. However, as this minimum value highly depends on the multiplier structure, their bitwidth, the number of taps in the filters as well as on the baseline coefficients, it cannot be easily defined analytically, but it is simply derived through manual experiments to show a lower bound on the achievable gains. When the coefficients are perturbed for lower power consumption, the frequency response of the filter changes, and therefore, relaxed specifications should be defined to accept only the coefficients that still provide sufficient filter quality. Hence, the most power-efficient coefficients that meet the relaxed specifications are chosen among the considered range of values at each optimization step and this optimization is repeated until a local minimum for the power consumption is found[6]. The obtained coefficients are then programmed instead of the original reference FIR coefficients.

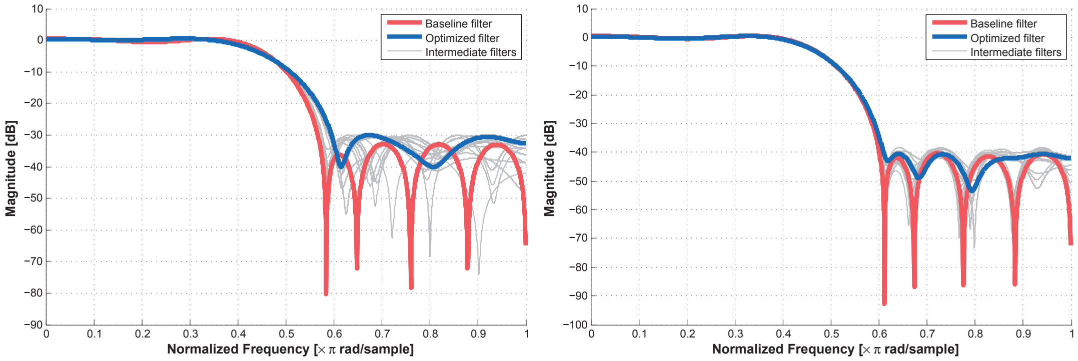
The proposed optimization technique yields a well-controlled quality degradation, since the relaxed specifications are a-priori well defined and are always met. In addition, both the frequency responses of the filter, as well as the dynamic power consumptions can easily be estimated during the filter optimization process without the need for time consuming power simulations on the whole FIR filter, since the implemented multiplier topology (used as basic building block) has been previously characterized.

4.5.2 Simulations Results

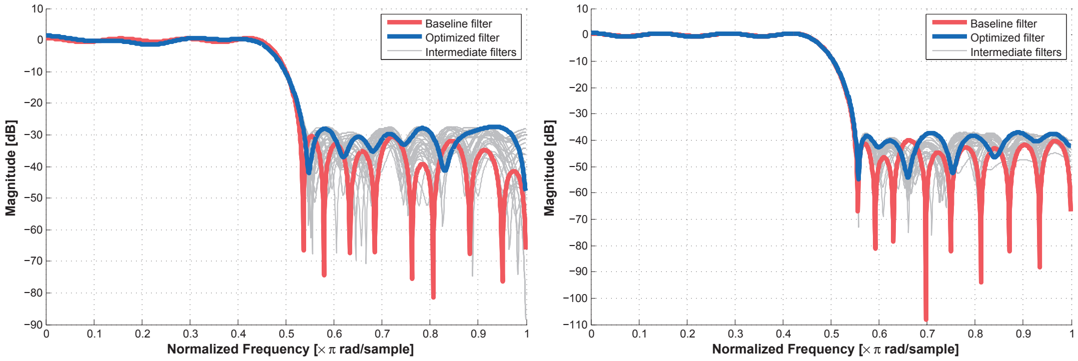
In order to evalute the efficiency of the proposed power-optimization technique, the design of four low-pass filters with different specifications on different accelerators (with different maximum filter lengths and wordlengths) is considered. In particular, two different stopband attenuations, A_s, of −30 dB and −40 dB are achieved by implementing either 8×8 or 10×10-bit

---

[6] Note that more sophisticated algorithms are possible and may lead to potentially even better results.



(a) 16-taps FIR low-pass filter with 8×8-bit multipliers (b) 16-taps FIR low-pass filter with 10×10-bit multipliers



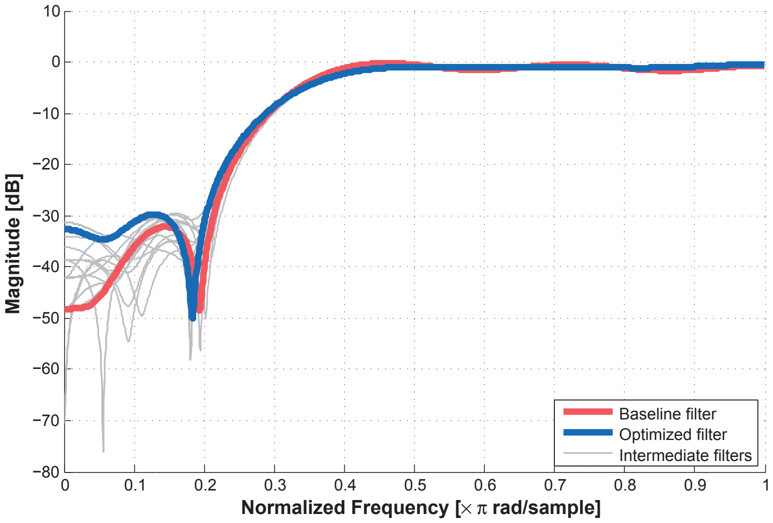
(c) 32-taps FIR low-pass filter with 8×8-bit multipliers (d) 32-taps FIR low-pass filter with 10×10-bit multipliers

Figure 4.15: Magnitude response of the initial and power-optimized FIR low-pass filters implementing various radix-4 Booth-recoded multipliers.

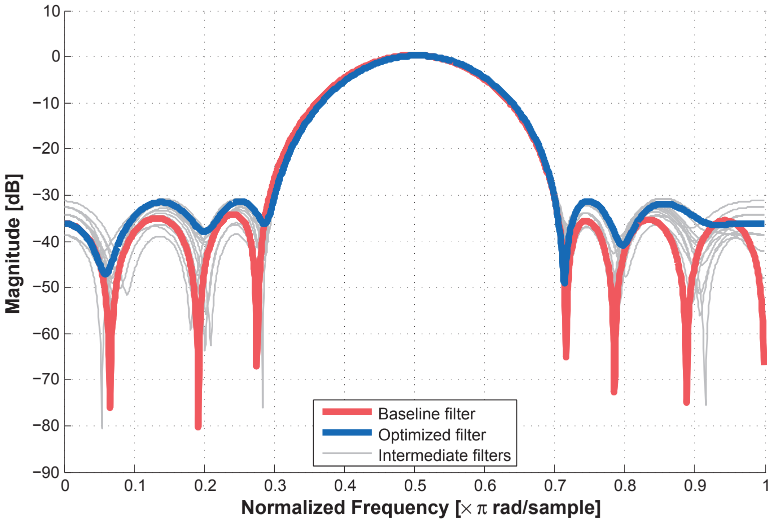
multipliers, respectively. The cut-off frequency is equal to half of the Nyquist frequency and the filters have been designed with either 16 or 32 taps to achieve 0.2 and 0.1 $\pi\frac{\text{rad}}{\text{sample}}$ transition bands (F_t), respectively. BR4 multipliers have been used in all the FIR accelerators since they provide the highest clock frequency among the two considered topologies. The design specifications of the four filters and the employed accelerators are summarized in Table 4.2.

For each of the filters, the coefficients that satisfy the baseline specifications were initially derived and the post-layout power consumption of the filter was measured. The obtained coefficients were used as initial values for the application of the proposed power-optimization technique where a 3 dB error on the stopband attenuation is tolerated for the corresponding relaxed specifications. The power consumption obtained with the perturbed coefficients is finally measured and compared to the power consumed by the baseline filter.

The magnitude responses of the baseline and the optimized filters are provided in Fig. 4.15 for all the design specifications. As expected, the perturbed coefficients result in a degraded filter response when compared with the baseline filters. However, they still satisfy the relaxed



(a) High-pass filter



(b) Band-pass filter

Figure 4.16: Magnitude response of various filters implemented with a 16-taps FIR accelerator and 8×8-bit radix-4 Booth-recoded multipliers. For each filter type, both initial and power-optimized version is provided.

specification on the stopband attenuation. Since the filter performance is checked during each optimization step, the quality degradation is always kept within the error bounds.

The resulting power savings given by the perturbation of the coefficients in all the considered filters are summarized in Table 4.2. Since the optimization algorithm considers the dynamic power consumption of the multipliers only, both the estimated dynamic power savings on the multipliers (S_M) and the effective dynamic power savings on the entire FIR filters (S_F) are reported for comparison. For a 3 dB error on the stopband attenuation, the reduction in dynamic power consumption for the FIR filters ranges from 11.7% to 20.2%. The power

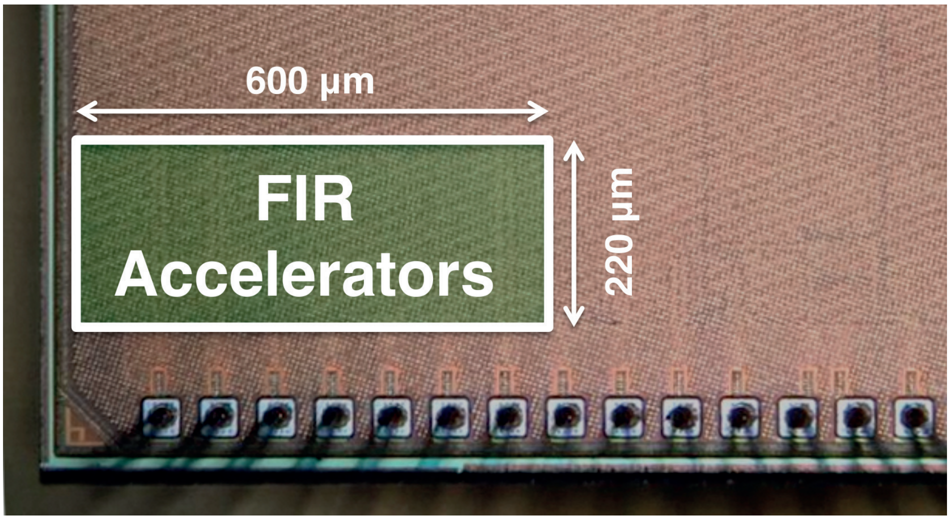


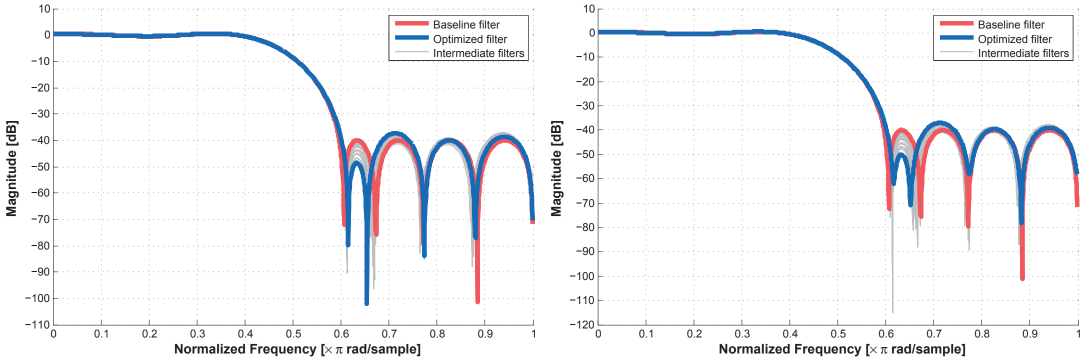
Figure 4.17: Die micrograph of the FIR accelerators.

savings in the filters are slightly smaller than the expected power reduction in the multipliers, since the power consumption of the registers, clock-distribution network, adders and buffers is not affected by the proposed technique.

The efficiency of the proposed power-optimization technique has also been evaluated considering the design of both a high-pass and a band-pass filter. For the high-pass filter, the design specifications are a stopband attenuation of −30 dB, a $0.2\,\pi\frac{\text{rad}}{\text{sample}}$ transition band and a cut-off frequency equal to 40% of the Nyquist frequency. The band-pass filter is centered at half of the Nyquist frequency and it is designed to have a stopband attenuation of −30 dB and a $0.17\,\pi\frac{\text{rad}}{\text{sample}}$ transition band on both sides of the frequency response, as well as cut-off frequencies equal to 46% and 54% of the Nyquist frequency. An FIR accelerator with 16 taps and 8×8 BR4 multipliers has been used for both filters, as for one of the previously evaluated low-pass filters. Accepting a worst-case 3 dB error relative to the stopband attenuation specification, the proposed technique allows to reduce the dynamic power consumption by 32.9% and 23.8% in the multipliers of the high-pass and the band-pass filter, respectively. The savings on the total dynamic power consumption of the programmable FIR accelerator are 25.6% and 20.3% for the high-pass and of the band-pass filter, respectively. For both case studies, the magnitude responses of the baseline and the optimized filters are shown in Fig. 4.16.

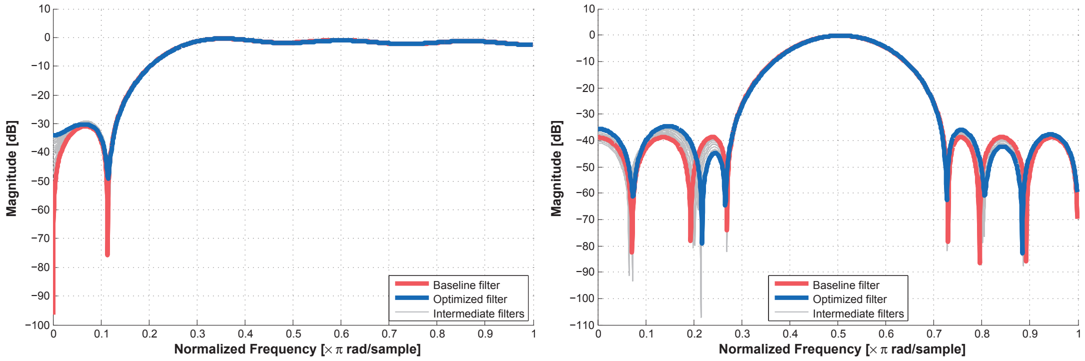
4.5.3 Test Chip and Measurements Results

Two reconfigurable FIR accelerators were fabricated as part of a 28 nm FD-SOI test chip, occupying an area of 0.132 mm^2. Since a low-power design space is considered, the accelerators are operated at 20 MHz at a near-threshold 0.6 V supply voltage. The micrograph of the fabricated circuits is shown in Fig. 4.17.



(a) Low-pass filter with 16×16-bit radix-2 Baugh-Wooley multipliers

(b) Low-pass filter with 16×16-bit radix-4 Booth-recoded multipliers



(c) High-pass filter with 16×16-bit radix-4 Booth-recoded multipliers

(d) Band-pass filter with 16×16-bit radix-4 Booth-recoded multipliers

Figure 4.18: Magnitude response of the initial and power-optimized filters implemented with the 16-taps FIR accelerators of the test chip.

The fabricated FIR accelerators were implemented in a direct form, as shown in Fig. 4.12, with 16 taps and 16×16-bit multipliers to verify the power benefits of the proposed technique on silicon and with a reasonably wide bit-width for the multipliers. The two accelerators differ by the topology used for the multipliers. The first design is based on radix-2 Baugh-Wooley multipliers and the second design on radix-4 Booth-recoded multipliers. An integrated serial interface provides communication from the external pins of the chip to the accelerators and it is used to configure the coefficients as well as all the other settings of the chip (such as the clock frequency). The data provided as input to the filters is generated by an integrated Fibonacci linear-feedback shift register (LFSR). Each filter can be independently clock-gated to separately measure their power consumption.

In order to apply the proposed technique to the fabricated accelerators, the power characterization of each of the implemented 16×16-bit multiplier topologies is required. Ideally, this characterization should be performed measuring the power consumption for each of the possible 2^{16} coefficient operands. However, this procedure would result in a very long and unpractical testing/characterization time. On the other hand, simulation results as well as the

Table 4.3: Power measurements of the FIR accelerators in the test chip

Filter Type	Multiplier Type	P_{base} $[\mu W]$	P^{a}_{opt} $[\mu W]$	S^{a}_{F} [%]
Low-Pass	Radix-2 Baugh-Wooley	335.6	224.1	33.2
Low-Pass	Radix-4 Booth-Recoded	328.7	218.2	33.6
High-Pass	Radix-2 Baugh-Wooley	360.1	258.6	28.1
High-Pass	Radix-4 Booth-Recoded	350.7	248.7	29.0
Band-Pass	Radix-2 Baugh-Wooley	385.2	309.3	19.7
Band-Pass	Radix-4 Booth-Recoded	374.1	297.3	20.5

[a] Obtained when tolerating a 3 dB error on the stopband attenuation.

structure of the circuit show that some of these operands have similar power consumption, therefore they can be divided into groups and the power consumption of only one coefficient operand per group will be measured to limit the characterization time. To this end, coefficient operands characterized by the same average number of non-zero partial-product bits are grouped together. This count of partial-product bits is simply obtained from analytical models, as the ones previously used to evaluate the non-zero partial-product bits for 8×8-bit multipliers in Fig. 4.7 and Fig. 4.8, and therefore it can be obtained with a relatively short computation time. Thus, the average number of non-zero partial-product bits for each coefficient operand has been derived for both the 16×16-bit BW2 and BR4 multipliers and their coefficient operands were divided into 32 and 9 groups, respectively. At this point, the power consumption of only one operand per group is measured and the obtained power estimate is assigned to all the other operands of the same group, allowing to dramatically reduce the characterization time.

During the power characterization, the power consumption of the reference coefficient of each group is measured by assigning the considered operand to each of the 16 taps and by continuosly operating the filter. Even though this measurement also includes the power overhead of the registers and adders of the filter, it is dominated by the power consumption of the multipliers[7] and the results can be used in the optimization algorithm of the proposed technique. A low-pass filter, a high-pass filter, and a band-pass filter are designed for each of the fabricated FIR accelerators. After having obtained the power characterization of both multipliers, the baseline coefficients are perturbed with the proposed optimization algorithm, tolerating a 3 dB worst-case error on the original stopband attenuation specification. The magnitude responses of the baseline and the optimized filters are reported in Fig. 4.18 for both FIR accelerators.

---

[7] Note that based on differences in the layout and circuit details (such as driving strength and parasitics) independent characterization of the individual multipliers may lead to even larger savings.

The measurements confirm the expected benefits of the proposed technique on both fabricated FIR accelerators. The power consumption of each of the accelerators with both the baseline (P_{base}) and optimized coefficient operands (P_{opt}) for each filter type as well as the total power savings (S_{F}) are summarized in Table 4.3. Considering the implementation of the low-pass filter as an example, the proposed technique is able to reduce the power of the baseline filter from 335.6 μW to 224.1 μW for the FIR accelerator with BW2 multipliers and from 328.7 μW to 218.2 μW for the FIR accelerator with BR4 multipliers. Similar reductions on the power consumption are also obtained for the other filter types, confirming that the presented technique can be applied succesfully for different filter characteristics. In summary, the measurements confirm the benefits of the proposed power-optimization technique and show that the measured power savings range from 19% to 33% for the considered filters when the optimized coefficients are used.

4.6 Conclusion

The application of VOS was shown to have many limitations: steep degradation of the QoS when VOS is applied to timing-critical circuits, the circuits that gracefully degrade under VOS are mainly slow circuits based on RCAs which highly limit the maximum operating frequency of the system, and the prediction of the impact on quality degradation due to timing errors is highly complex. Given these critical limitations of VOS, an approximate-computing technique that reduces the switching activity in programmable hardware has been proposed to achieve further power savings in addition to the application of conventional voltage scaling. The proposed technique is based on the observation that the power consumption of multipliers having one input port connected to a constant coefficient operand can significantly vary with the value of that coefficient. In programmable architectures where one coefficient is constant over many cycles, this can be exploited for power savings by approximating the original coefficient with a similar coefficient that yields lower power consumption. This technique can effectively be applied, for example, to reconfigurable FIR accelerators. A simple greedy algorithm is used to modify the coefficients of a baseline filter to derive a new set of coefficients that are optimized for low power consumption while allowing for some degradation of the filtering quality. By exploiting the flexibility on the algorithm level, the proposed approximate computing technique does not require any design overhead for a programmable accelerator. At the same time, it ensures the quality of the baseline filter whenever it is required, while it offers also the possibility of scaling the power consumption at runtime when energy is short and reduced accuracy is tolerated. The presented technique has been demonstrated on two FIR accelerators implemented in a 28 nm FD-SOI process technology, showing dynamic power savings of up to 33% for an accepted 3 dB degradation on the stopband attenuation.

5 Gain-Cell Embedded DRAMs: Modeling and Design-Space Exploration

The last decade of computing has been driven by data-intensive applications, which have raised the need for large-capacity memories. As a result, the silicon area of many microprocessors and system-on-chip (SoC) designs is dominated by embedded memories [70–73], especially when conventional six-transistor (6T) static random-access memory (SRAM) is used. Dynamic random-access memory (DRAM) offers a significant advantage over SRAM for large memory sizes, as it provides higher memory density by using only a one-transistor one-capacitor (1T-1C) storage cell. Furthermore, even though DRAMs need a periodic refresh to avoid loss of data, their leakage power is highly reduced as compared to SRAMs, due to the absence of direct leakage paths between the supply voltage and ground.

Among the different DRAM topologies, gain-cell embedded DRAM (GC-eDRAM) [74] is a fully logic-compatible embedded memory that does not require any dedicated process steps during manufacturing. Given both its high-density and low-latency operation, GC-eDRAM has proven to be a valid and low-power alternative to SRAM [75–80]. Accessing GC-eDRAMs is faster than conventional 1T-1C-based DRAMs as active elements are used to implement the read port of the gain cell (GC), which is the fundamental storage unit. Moreover, dual-port operation is inherently provided by GC-eDRAMs as GCs include separate write and read ports. Finally, as opposed to conventional DRAMs, the read is non-destructive in GC-eDRAMs, and therefore, the overhead of writing back the data after read is avoided.

The design space of memories is highly complex due to the presence of a large number of design variables, which have significant effects on the memory performance metrics. In fact, for GC-eDRAM the design space is even larger than the one of SRAM due to the large number of bitcell topologies and complex tradeoffs, for example between retention time and access time. For this reason, modeling tools for integrated memories are key for the rapid exploration of this design space to facilitate the choice between several design options without the need for many long and complex full design iterations. In fact, such modeling tools find applications in two main areas:

- *Memory design:* modeling tools can assist designers by predicting the performance impact of important design choices early in the development cycle, during the preliminary stage of the design phase.
- *System architecture:* architectural optimization and design-space exploration of complex SoCs heavily relies on accurate performance estimation of their embedded memories. For this reason, high-level memory design and integration of memory modeling tools with system simulators [81] is an integral part of the system development.

A number of modeling tools for conventional DRAMs have already been proposed in the literature [81–84]. Even though each of these tools provides a certain degree of flexibility, none of them can be adapted to reflect the behavior of GC-eDRAM. This restriction is mainly due to the fundamental differences in the bitcell between DRAM and GC-eDRAM and the specific features of GC-eDRAM, such as the non-destructive read and two-port operation. To overcome this limitation, GEMTOO is proposed in this chapter as the first modeling tool for GC-eDRAM, which enables both computer-aided design and modeling of future GC-eDRAMs, as well as the evaluation of existing GC-eDRAMs in an extended design space (i.e., different memory sizes or organizations).

Contributions The specific contributions of this chapter are summarized as follows:

- GEMTOO, the first modeling tool for GC-eDRAMs, is described in this chapter.
- The proposed modeling tool is based on input parameters related to technology, circuits, and memory organization of GC-eDRAMs, to allow for the exploration of a large design space.
- The physical floorplan of the GC-eDRAM is considered in the estimation of memory metrics, thereby accounting for the impact of the memory organization, as well as for the load given by the interconnects.
- An accurate timing estimation is provided by modeling the effect of the deterioration of the stored data (i.e., charge) in the GC.
- The tool is implemented in a modular structure to allow for the selection or the introduction of different modeling strategies.
- The code of GEMTOO is open source and publicly available [85].
- The timing estimation of GEMTOO is validated against transistor-level simulations of different GC-eDRAM implementations, which include both resistive and capacitive parasitics from post-layout extraction, as well as against silicon measurements of a GC-eDRAM fabricated in 28 nm CMOS process technology.

- Multiple case studies of design-space exploration are presented based on the proposed modeling tool to find the best design choices that fulfill the most critical memory requirements, such as maximum operating frequency, highest availability and bandwidth, as well as best memory density.

The rest of this chapter is organized as follows: Section 5.1 presents the related work regarding DRAM modeling. A background on the design of GC-eDRAMs is provided in Section 5.2. Section 5.3 presents the implemented model. GEMTOO is validated against post-layout simulations and silicon measurements in Section 5.4. Section 5.5 presents multiple case studies of design-space exploration based on GEMTOO modeling for different memory requirements. Section 5.6 concludes this chapter.

5.1 Related Work

Modeling tools for memories are key for the rapid evaluation of their architectural tradeoffs in a large design space. In this section, few modeling tools for DRAMs that have been proposed in literature [81–84] are described, as they model memories that are the closest comparison to GC-eDRAM.

CACTI-D [82] is probably the most well-known modeling tool for DRAM. It is based on CACTI [86], which is a popular modeling tool for SRAM, and it estimates energy consumption as well as some timing parameters. CACTI-D models the complete memory hierarchy from the SRAMs that implement the lower-level caches to the DRAMs in the higher levels of the memory hierarchy. Even though it is generally considered as the main modeling reference for DRAMs, it has a number of limitations, such as the sole use of the H-tree structure, which limits the types of memory organizations that can be evaluated, and it is not immediately compatible with GCs as it can only model 1T-1Cs bitcells.

Vogelsang [83], presented a very detailed model for the energy consumption in DRAMs. The current consumed by each operation within the memory is calculated considering the floorplan of the storage arrays as well as the location of the wires. However, this model does not include any timing estimation and cannot model embedded DRAMs as it primarily targets commodity DRAMs.

DArT [84] is a flexible modeling tool for DRAMs as it is based on the definition of components that are used to specify the memory structure. It also enhances the accuracy of a traditional RC delay model by considering the effect of overlapping operations, input transition delays, and velocity saturation, as well as by introducing a time-variant switch model. However, the tool cannot model GC-eDRAMs as it targets 1T-1C DRAMs and, to the best of the author's knowledge, the tool is not publicly available.

DRAMSpec [81] is an open-source DRAM modeling tool, where the lowest abstraction level is the DRAM storage cell that is represented only by equivalent resistance and capacitance

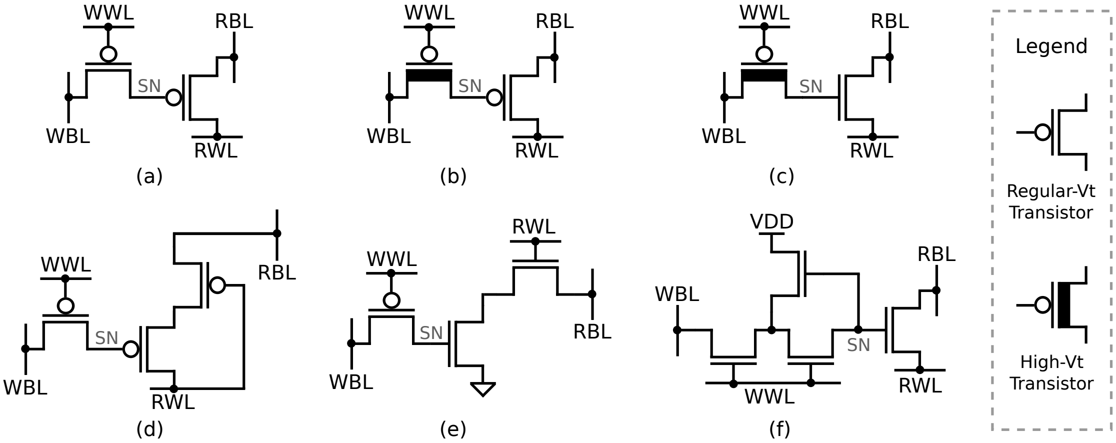


Figure 5.1: Variety of GCs reported in the literature: (a) 2T PMOS-only GC [76], (b) 2T PMOS-only GC with high-V_T access transistor [77], (c) 2T GC with PMOS high-V_T access transistor and NMOS read transistor [88], (d) 3T PMOS-only GC [78], (e) 3T GC with PMOS access transistor and NMOS read transistors [89], (f) 4T NMOS-only GC with internal feedback [79].

values. The timing and current parameters generated by DRAMSpec can be integrated into the gem5 simulator [87] to evaluate the performance of the memory within a computing system. Although the representation of the storage cell with only passive components reduces both the complexity and the execution time of the tool, DRAMSpec does not consider some key technology-related parameters such as the storage-cell retention time, which determines the memory availability. Also, DRAMSpec primarily targets off-chip 1T-1C DRAMs and cannot be used to model GC-eDRAMs.

Even though the described DRAM modeling tools provide some flexibility in both the considered technology and memory organization, they are all not able to model GC-eDRAMs. This is mainly due to the variety of GCs [74] that cannot be modeled as conventional 1T-1C storage cells. In addition, GC-eDRAMs are characterized by specific features that are not present in conventional DRAMs, such as the non-destructive read and the dual-port operation which lead to significant differenes in the peripheral circuits. For these reasons, a dedicated tool that can accurately reproduce all the features of GC-eDRAMs with the same flexibility of state-of-the-art memory modeling tools is required.

5.2 Background on GC-eDRAM

The main building blocks, memory organization, and features of GC-eDRAMs are described in this section. The GC is the fundamental storage component in GC-eDRAMs and it is discussed first, as its implementation has a significant impact on many design aspects of the memory, such as the maximum operating frequency, the memory availability, and the memory density. Moreover, the memory organization of a GC-eDRAM is described and a set of architectural transformations are defined. Specific features of GC-eDRAMs are also highlighted and a design example of a processor that implements a GC-eDRAM as data memory is described.

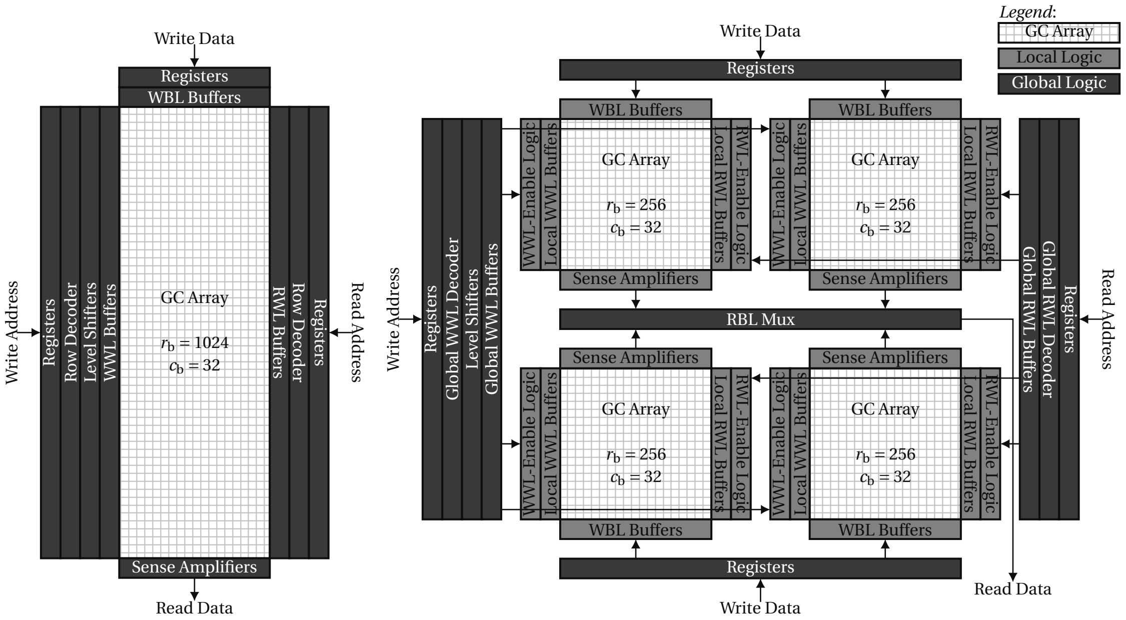
5.2.1 Gain Cell

The gain cell is the fundamental storage component in GC-eDRAMs. A variety of GC topologies have been proposed in the literature [74], and several examples are presented in Fig. 5.1. The GC topologies primarily differ in their transistor count. For example, the most compact GC is made of two transistors (2T), as shown in Fig. 5.1(a)–(c). GCs made of three transistors (3T) have also been proposed to avoid the read-bit-line saturation problem [74] (Fig. 5.1(d) and Fig. 5.1(e)), and even a 4T GC (Fig. 5.1(f)) was presented in [79, 80] to increase the retention time with internal feedback. Often, implementations with different transistor types are also possible within the same GC topology. In fact, PMOS transistors and transistors with high threshold voltages (V_T) are usually preferred as access transistors due to their low OFF-current while NMOS transistors are used as read ports as they can drive larger ON-currents to reduce the read-access delay.

Even though many different GC topologies exist, they are generally composed of a write port, a storage node (SN), and a read port. Moreover, as GCs are dual-port, they have a write bit line (WBL), a read bit line (RBL), a write word line (WWL), and a read word line (RWL). During a write operation, the WWL is enabled, the write port starts conducting, and the WBL value is written into the SN. At the beginning of a read operation, the RBL is precharged (or discharged) to a constant voltage and once the RWL is enabled, the RBL is conditionally discharged (or charged) depending on the value stored on the SN. For a better understanding of the design intricacies behind the write, storage, and read operations of GCs, a short recap of three important aspects is provided in the following paragraphs.

Voltage Boosting for the WWL The write port is generally implemented as a pass-transistor [40]; therefore the write of one of the logic values is degraded, since the access transistor is cut off before the full logic level is transfered to the storage node. This is usually overcome by overdriving the WWL to enable writing of a strong logic value into the SN. This technique is applied to any modelled, simulated, or measured GC-eDRAM in this chapter. Nevertheless, it is also worth mentioning that another solution is the implementation of the write port with a transmission-gate [90], however this increases the GC transistor count and creates an additional leakage path from the storage node (whose impact is discussed below).

Dynamic Storage As the GC is a dynamic storage cell, the stored value is degraded over time due to leakage currents, thereby requiring a periodic refresh. The minimum refresh rate primarily depends on the retention time of the GC, which is defined as the maximum amount of time between a write operation and a successful read operation. Higher leakage currents decrease the retention time and requires higher refresh rates. In addition to the process



(a) Monolithic GC-eDRAM ($r_a = 1$, $c_a = 1$).

(b) Hierarchical GC-eDRAM ($r_a = 2$, $c_a = 2$).

Figure 5.2: Different memory organizations for a 32 kb GC-eDRAM.

technology, cell-to-cell variations, and temperature, the retention time highly depends on many other design parameters, such as the type of write transistor, the stored data value, and the WBL value. For example, when considering an NMOS as the write transistor, a stored logic-1 degrades much faster than a logic-0 under worst-case biasing conditions on the WBL [79]. To avoid being limited by the low retention time of the stored logic-1, the WBL can be precharged to logic-1 whenever the GC array is not accessed for writing. This minimizes the leakage when a logic-1 is stored and the retention time is determined by the much slower decaying logic-0. More complex GC topologies have been proposed in the literature to increase the retention time and limit the refresh rate. Among them, the 4T GC [79, 80, 91] presents a retention time that is 30× longer than conventional GCs at the cost of an additional transistor.

Read-Delay Degradation GCs are generally implemented with minimum-sized transistors to minimize the footprint of the storage cell and to maximize the memory density. This choice limits the driving capability of the read transistor, which needs to discharge the relatively large capacitance of the RBL. In addition, as the SN value is degraded over time, the voltage overdrive of the read transistor is reduced, thus limiting its drive current even further. Considering the sum of these effects, the amount of time required to discharge the RBL can be significant and typically limits the maximum operating frequency in GC-eDRAMs.

5.2.2 Memory Organization in GC-eDRAMs

The organization of the different building blocks in a memory is key for both its timing and area optimization. When considering the modeling and integration of a GC-eDRAM within a design it is useful to define different levels of hierarchy from the lowest level to the highest one:

Gain Cell The fundamental 1-bit storage cell.

GC Array Multiple GCs are arranged in an array, where each row shares the WWL and the RWL and each column shares the WBL and the RBL. In this chapter, the number of rows and the number of columns within a GC array are denotated as r_b and c_b, respectively.

GC-eDRAM GC arrays are complemented with peripheral circuits to form a complete GC-eDRAM, in which the memory content can be addressed with write and read operations. Depending on the number of implemented GC arrays, additional levels of hierarchy exist within the GC-eDRAM. Thus, two cases are distinguished:

- *Monolithic GC-eDRAM*: the memory implements a single GC array together with the logic to address the storage content, as shown in Fig. 5.2(a).
- *Hierarchical GC-eDRAM*: the memory content is distributed across several GC arrays, which are organized in a grid, where the number of rows and columns is represented by r_a and c_a, respectively. This memory organization introduces a two-level hierarchy, where the bottom level is comprised of the GC array and the dedicated *local* logic, which are replicated several times in the top level of the hierarchy, where the *global* logic is shared across the whole memory. An example of a hierarchical GC-eDRAM is shown in Fig. 5.2(b).

Register-Transfer Level (RTL) The components up to and including the hierarchical GC-eDRAM are typically designed as full-custom blocks, which are then instantiated as macros at the RTL together with the memory controller and the other blocks of the system. The memory controller is a key component for the operation of the GC-eDRAMs, as it handles both the memory refresh operation and the access scheduling; however, it is outside the scope of this chapter.

Considering the above-described hierarchy, the memory organization defines how the storage is distributed across multiple GC arrays within the GC-eDRAM as well as the number of rows and columns of each GC array. The monolithic GC-eDRAM represents the simplest memory organization, as shown in Fig. 5.2(a) for a 32 kb memory size with a 32-bit word size. A single

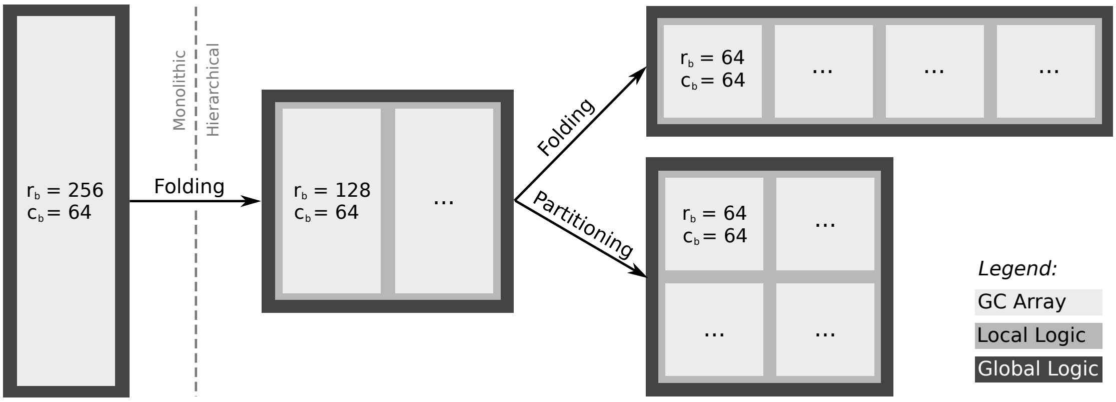


Figure 5.3: Examples of architectural transformations cutting the BLs, showing their impact on the memory organization of a 16 kb memory.

GC array is implemented together with logic that includes the registers and buffers to sample the write data and drive the long WBLs of the GC array as well as the sense amplifiers to read the data from the RBLs. As the GC-eDRAM is a dual-port memory, dedicated logic to sample, decode, and drive the long WLs of the GC array is implemented for both the write and the read address.

Even though the monolithic GC-eDRAM is a simple memory organization, it is not feasible for large memory sizes (i.e., larger than hundreds of kb). This is due to several reasons, starting with the the routing wires that would become very long and significantly degrade the memory speed. A large-size monolithic memory also leads to a very large aspect ratio, which might be unpractical to integrate in a floorplan with other components. As a solution, the hierarchical organization allows the implementation of large-size GC-eDRAMs by distributing the storage cells across several GC arrays with local logic and drivers that are connected and arranged in a floorplan to maximize speed [40], as shown in Fig. 5.2(b). Dividing the memory into GC arrays of a limited individual size also avoids long wire lengths. Therefore, the hierarchical organization improves the timing in large-size GC-eDRAMs and facilitates their physical integration into a floorplan with other components. The hierarchical organization introduces global logic and drivers that are shared across the entire GC-eDRAM and dedicated local logic and drivers that are replicated for each individual GC array. In particular, considering the example in Fig. 5.2(b), both the write and the read address are decoded by global row decoders that output global one-hot encoded WL signals used to access one of the r_b rows within the targeted GC array. Local WL-enable logic is implemented for both write and read ports to propagate the global one-hot WL signals to the GC array that is selected for access. Moreover, a global RBL mux is added to select the correct output data word during read.

5.2.3 Architectural Transformations in GC-eDRAMs

The implementation of a hierarchical GC-eDRAM is achieved by applying basic architectural transformations to an initial GC-eDRAM configuration with a monolithic GC array to divide its memory content into multiple smaller GC arrays. To this end, two transformations are defined: *partitioning* and *folding*. *Partitioning* cuts either the BLs or the WLs of the GC array into half, thereby dividing the GC array into two identical arrays. *Folding* consists of partitioning and aligning the partitioned GC arrays along the direction of the cut. Examples of these architectural transformations are shown in Fig. 5.3, where a monolithic memory is taken as starting point.

Both of the defined architectural transformations shorten the length of either the BLs or the WLs, and thus, they reduce the internal memory delays that depend on the load associated with these long wires. Besides the timing benefits, folding also helps to reduce any imbalance between the width and the height of the memory, which also facilitates its physical integration in a floorplan with other components. Moreover, the application of any architectural transformation converts a monolithic GC-eDRAM into a hierarchical GC-eDRAM, as data is distributed across multiple GC arrays. A global RBL mux, which selects the data word to be read from one of the GC arrays, is also introduced as soon as any architectural transformation that cuts the BLs is applied. Both partitioning and folding transformations can be applied repeatedly on each GC array to further optimize both the timing and the aspect ratio of the memory.

It is worth noting that when folding a GC array to shorten the BLs, the rows of the two resulting smaller GC arrays could be merged together to obtain a single GC array, where each row stores two data words instead of one. Even though this memory organization might provide less overhead in the logic as compared to keeping the GC arrays separate, it requires that any write access to a single word is preceded by a read of the entire GC-array row to modify the targeted word and to write back the updated row. Thus, merging multiple GC arrays to store more than one word within a GC-array row introduces a read-access overhead for each write operation. As this chapter focuses on GC-eDRAMs that ensure a single-cycle access, only separate GC arrays, each storing a single word per row, are considered as building blocks of hierarchical GC-eDRAM that results from folding.

5.2.4 Features of GC-eDRAMs

While many of the concepts of memory organization can be applied to different memory types, the GC-eDRAM exhibits specific features and allows for design choices that are not common to conventional DRAMs. These are summarized as follows:

Dual-Port Memory As a GC has both a write and a read port, the GC-eDRAM is a dual-port memory. Thus, it is typically equipped with dedicated address decoders for the write and read address, respectively.

Single-Cycle Refresh Given the dual-port capability, the refresh of the memory content in a GC array can be performed by reading the content of the row at a generic address n while writing back the previously-read content to the row of address $n-1$ within the same cycle. Thus, a refresh of one row occupies only a single cycle on each of the two ports. Conventional single-port DRAMs do not have this feature and require two cycles to perform the same refresh operation.

WWL Voltage Boosting voltage boosting is applied to the WWLs to overcome the threshold loss when writing to GCs implementing a pass-transistor as a write port. For this reason, level shifters are commonly placed in between the write-address decoders and the WWL buffers.

WBL Biasing The retention time of a GC highly depends on the WBL biasing condition, as previously described in Section 5.2.1. Thus, when not writing to the memory, the WBL buffers are typically disabled and their outputs are set to high impedance, while small pull-up/down transistors force the WBLs to one of the logic levels (depending on the WBL biasing condition that maximizes the retention time of the implemented GC).

5.2.5 Computing with GC-eDRAMs: Application to a 32-bit Microprocessor

An application example of a GC-eDRAM used in a computing module is described in this subsection. A 32-bit processor has been implemented with a GC-eDRAM as embedded data memory using a 28 nm FD-SOI process technology. The GC-eDRAM is part of a larger data-memory subsystem that includes a memory controller and a memory buffer implemented as a standard-cell memory (SCM) [92–97]. The described memory subsystem is a programmable and flexible module that can adapt to the needs of the application (i.e., the program that is executed on the processor).

5.2.5.1 System Architecture

The architecture of the overall system is shown in Fig. 5.4. The central processing unit (CPU) is a 32-bit, 6-stage, in-order OpenRISC [47] core that supports both fixed-point and floating-point arithmetic operations. The processor includes 2-way cache memories for both instruction and data where the size of each cache memory is 16 kb. Any change on the critical path of the processor is monitored by a tunable replica circuit (TRC) [41, 46] and compensated by the application of body biasing (BB) [98, 99] on the standard cells that implement the CPU. The BB voltages for the CPU domain are locally generated by a dedicated body-bias voltage generator.

Considering the embedded memories, instructions are stored in an SRAM of 256 kb. The data memory is implemented as a GC-eDRAM of 1 Mb and an SRAM of 32 kb. Furthermore, an additional SRAM of 32 kb is included as part of the memory built-in self-test (BIST) module for the GC-eDRAM.

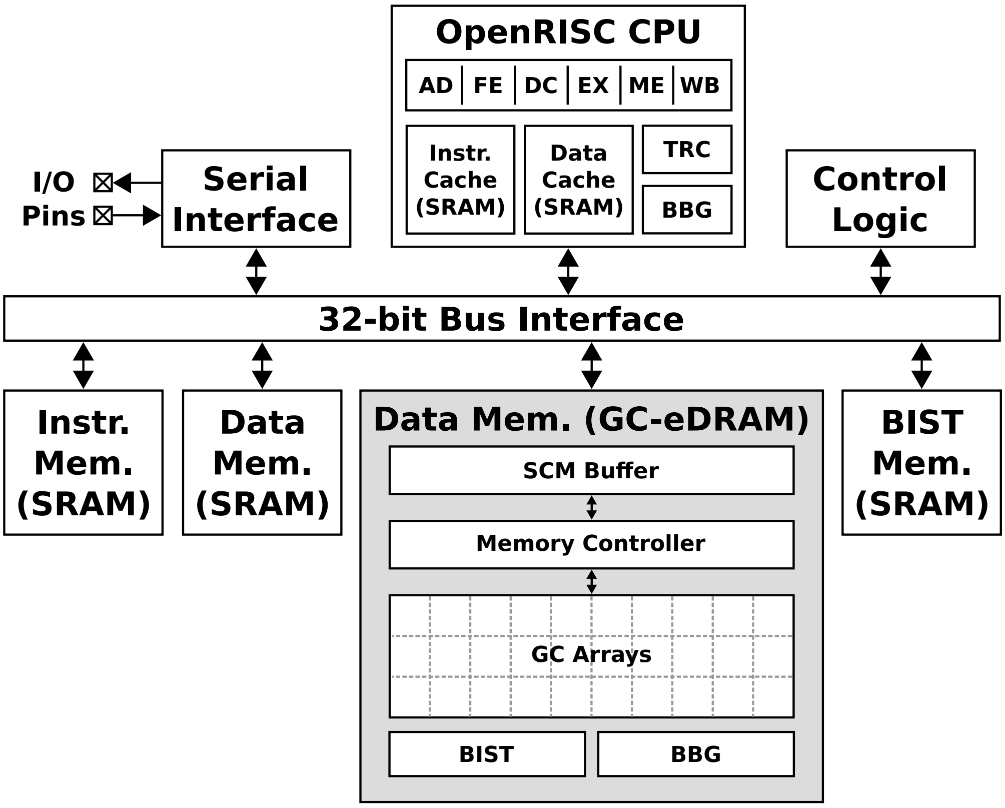


Figure 5.4: Architecture of the processor with GC-eDRAM as data memory.

A serial interface is provided to communicate with the chip through the input-output (I/O) pins. Any programmable block is configured with a control module that is based on a memory-mapped interface. A frequency-locked loop (FLL) [100] is used to internally generate a high-frequency clock signal. All the modules are connected over a 32-bit bus interface.

5.2.5.2 GC-eDRAM Subsystem

The data memory is implemented as a GC-eDRAM subsystem with a memory capacity of 1 Mb. At the circuit level, the implemented storage cell is a 3T GC with PMOS access transistor and NMOS read transistors [89], previously shown in Fig. 5.1(e). This GC topology has been selected as it is characterized by a small sub-threshold current through the write port and a sufficiently large drive current at the read port. BB can be applied to the implemented GC as the BB terminal of the PMOS access transistor is shared among all the GCs in the memory. Thus, reverse body biasing (RBB) can be applied to every access transistor of the memory for a further reduction on the leakage current of the write port. Similarly, the BB terminals of the NMOS read transistors are shorted together at the GC level and are shared across all the GCs in the memory. Thereby, the application of forward body biasing (FBB) on the read transistors of every GC can increase the read-port drive current together with the maximum operating

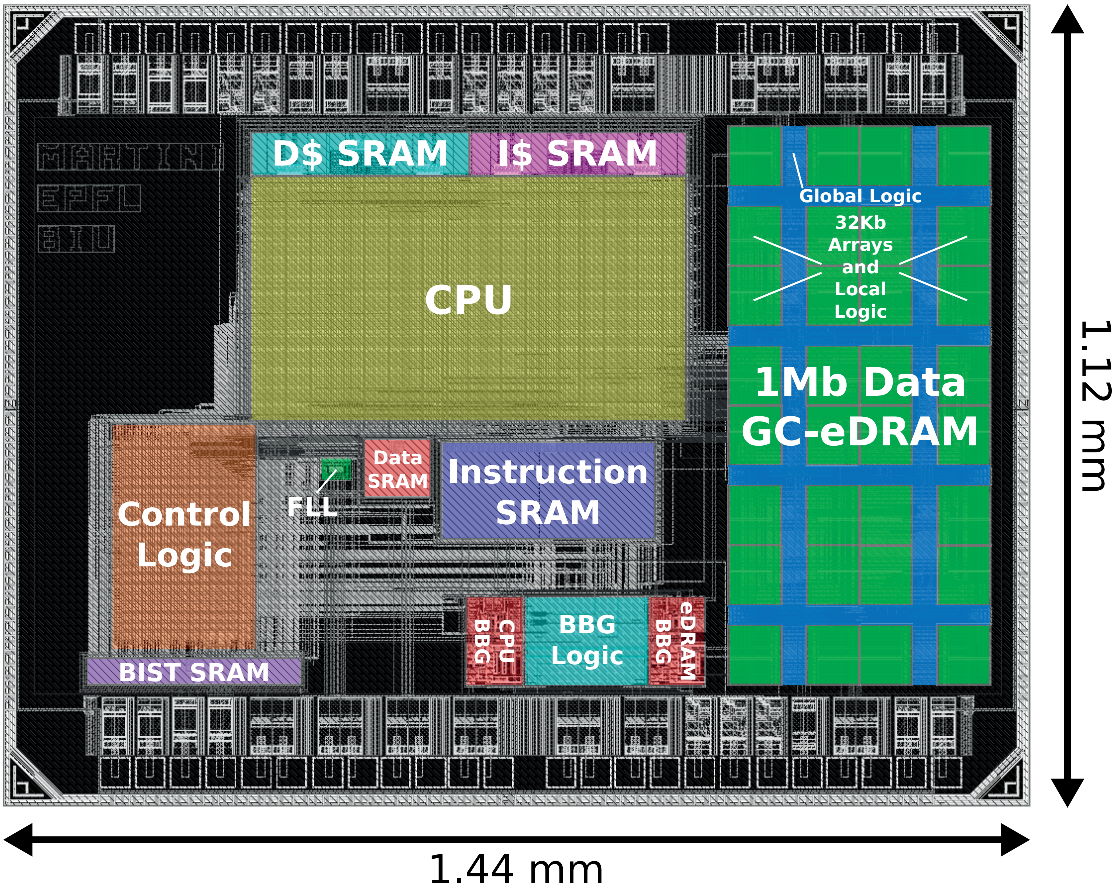


Figure 5.5: Layout floorplan of the processor with GC-eDRAM as data memory.

frequency of the memory. A dedicated body-bias voltage generator is integrated to locally generate both BB voltages required by the GC-eDRAM.

The memory content of the GC-eDRAM is distributed across 32 equal GC arrays, as depicted in Fig. 5.5. Each GC array has a memory size of 32 kb and is composed by 128 coloumns (c_b) for high memory density. In addition, each of the coloumns is terminated by a compact local latch that samples the output of the RBL sense amplifier. As the CPU handles data words of 32 bits, each row in a GC array stores four of them. Thus, any 32-bit write operation to a GC array must always be anticipated by a read access to the corresponding 128-bit wide row to modify the targeted 32-bit word and write the updated 128-bit content back to the array. The design of the GC arrays as well as the local logic is custom to ensure a high memory density and a fast operation, while the global logic of the GC-eDRAM has been designed at RTL and physically integrated in between the arrays, as shown in Fig. 5.5.

The GC-eDRAM subsystem also contains a memory controller that directly communicates with the GC-eDRAM. The primary role of the memory controller is to manage both read and write accesses to the GC-eDRAM. The refresh operation is also handled by the memory controller and all the GC arrays are refreshed in parallel. In order to increase the read-access throughput from the memory, the memory controller can trigger a read operation from one to four subsequent GC arrays in parallel. In this scenario, the read data is stored in the local latches at the output of each of the GC arrays, providing a fast access to the memory controller without the need to iteratively retrieve the data from each of the slow memory arrays. The number of GC arrays to be read in parallel is a programmable parameter that can be optimized based on the memory-access pattern of the application. The memory controller also manages the additional read access that is required when writing a 32-bit word to the 128-bit wide row of a GC array.

It is worth noting that the same clock signal is used in both the CPU and the GC-eDRAM domains, therefore the maximum operating frequency of the system might be limited by the memory arrays that may have a longer access time than the critical path of the CPU. For this reason, the memory controller allows for an arbitrary number of clock cycles for the access of the memory to reduce any potential frequency degradation.

A row buffer is included within the memory subsystem, which is implemented as an SCM [92–97]. The SCM buffer stores 128 bits, equal to the number of coloumns of a GC array. The memory content of the SCM buffer is updated by the memory controller based on the access pattern. When the requested data is found in the SCM buffer (i.e., similarly to a cache hit), the system bus will directly access the buffer for either read or write operations, therefore avoiding a slow and energy-costly accesses to the GC-eDRAM. Otherwise, when the system bus does not find the requested data in the SCM buffer (i.e., similarly to a cache miss), the memory controller updates the GC-eDRAM content if the data currently stored in the buffer has been previously modified and subsequently retrieves the requested data.

The main bottlenecks in the design of the described large memory subsystem are the optimization of the hierarchy for the GC arrays as well as the custom design of the single array. In fact, the design space of GC-eDRAMs is highly complex due to the presence of numerous design variables that significantly affect the memory performance metrics. Given this high design complexity, a fast and accurate estimation of the main GC-eDRAM metrics is key for an agile exploration of the large design space. Thus, a model that can estimate timing, bandwidth, and silicon area of GC-eDRAMs is required for the performance scaling of already existing memories as well as for the design of future GC-eDRAMs.

5.3 GEMTOO: GC-eDRAM Modeling Tool

The structure of GEMTOO, the proposed GC-eDRAM modeling tool, is shown in Fig. 5.6. Several models are included in GEMTOO to estimate the different output performance metrics based on the input parameters. These models are implemented as modular code, in which each module is based on either analytical formulas, tabulated values, or both. Analytical formulas are always preferred as they provide maximum flexibility and avoid the generation of technology/circuit specific look-up tables with time consuming simulations. Nevertheless,

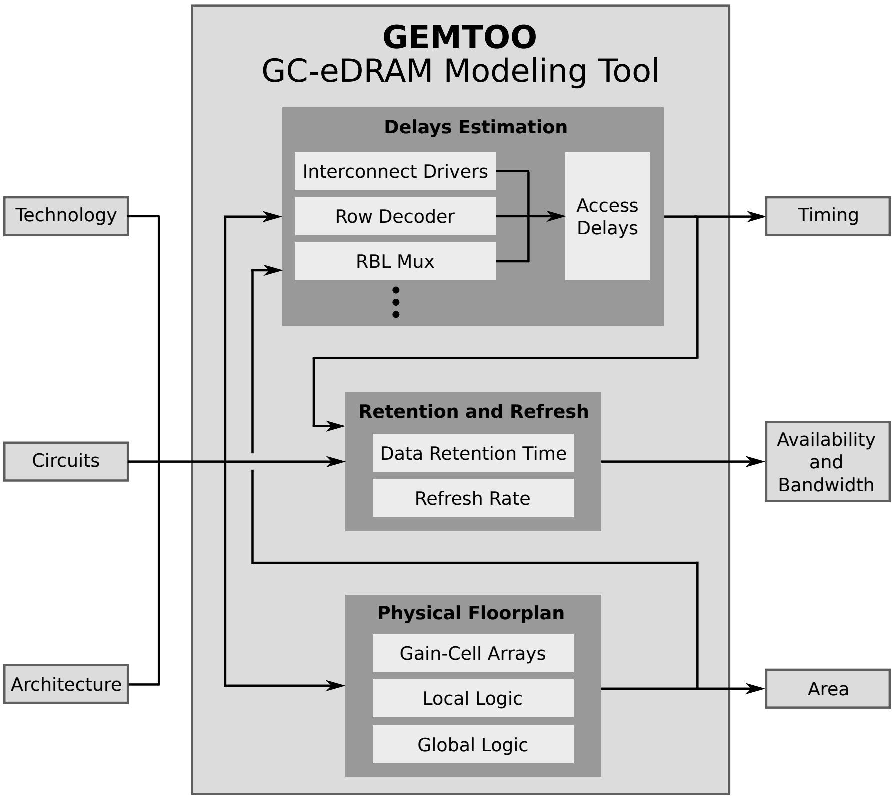


Figure 5.6: Structure of the GEMTOO modeling tool.

tabulated values based on either simulations or measurements are adopted when the accuracy of simple analytical equations is not sufficient. The modular structure of the code provides the flexibility to find a balance between these representations, as different model representations can be combined, depending on the sensitivity of the results to individual components and the accuracy requirements of the user.

The input parameters are divided into three categories: *technology*, *circuits*, and *architecture* parameters. The *technology* parameters only depend on the adopted process technology and include, for example, the capacitive and resistive parasitics of the wires. Any information related to circuit design, such as the type of GC, the drive strength of the buffers, the propagation delay of the sense amplifier, and the physical dimensions of the GC and of the circuits implemented in the logic, is specified in the *circuits* category. The *architecture* parameters describe the memory organization and include the memory size, the word size, and the architectural transformations.

The output memory performance metrics are also grouped into three main categories: timing, memory availability/bandwidth, and area of the GC-eDRAM. The following sections elaborate on how these metrics are derived.

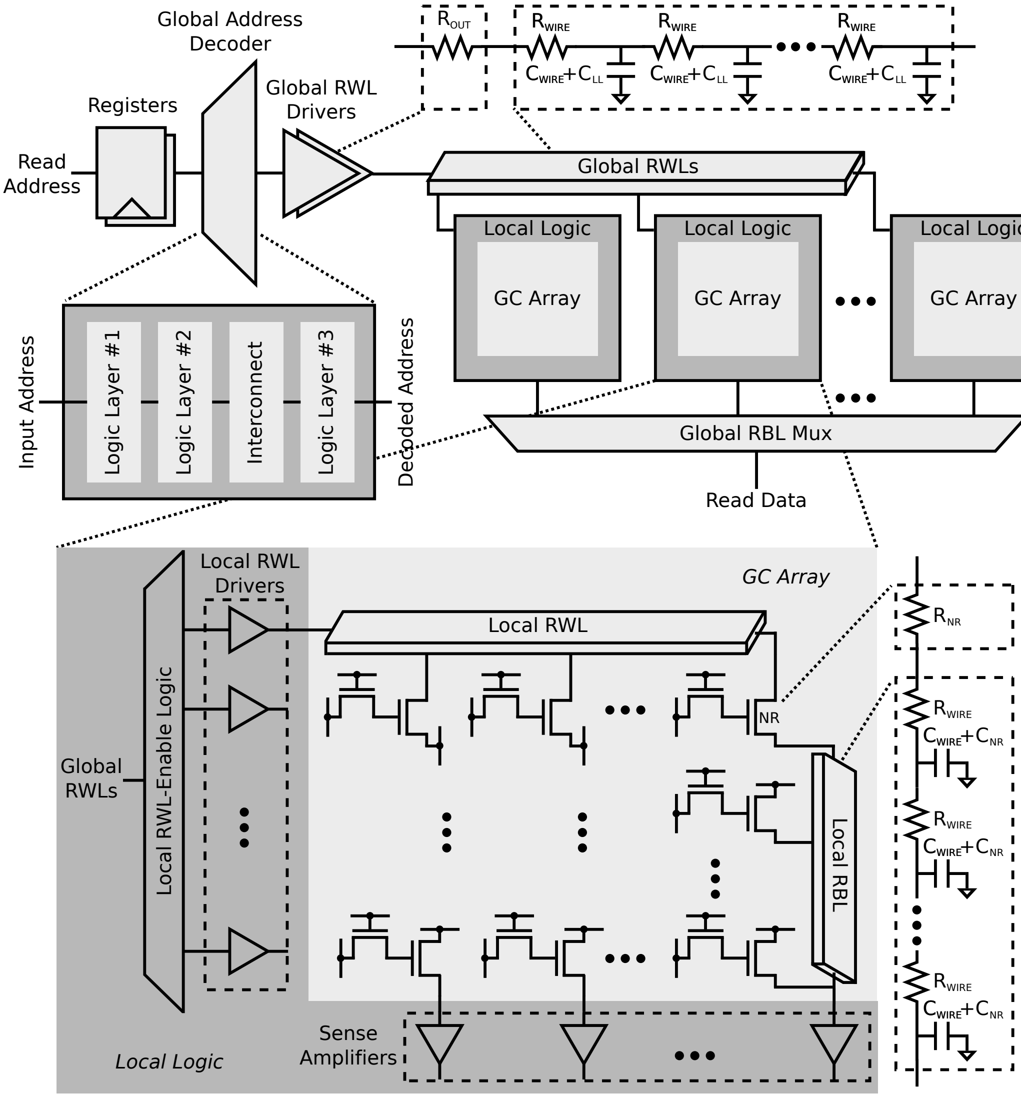


Figure 5.7: Delay modeling of the read path in a GC-eDRAM.

5.3.1 Timing

GEMTOO derives the delay of both write and read access, here defined as t_{wr} and t_{rd}, respectively, and determines which one limits the maximum operating frequency (f_{max}). The equation used to obtain t_{wr} is the following:

$$t_{\text{wr}} = \max(t_{\text{wwl}}, t_{\text{wbl}}) + t_{\text{wsn}}, \tag{5.1}$$

where t_{wwl} is the time between the sampling of the write address and the arrival of the write enable at the targeted GCs; t_{wbl} is the time between the sampling of the write data and the point when the data has settled on the WBLs of the targeted GC array; t_{wsn} is the time required to write data into the SN of a GC through its write port. Considering the read access, t_{rd} is derived by the following equation:

$$t_{\mathrm{rd}} = t_{\mathrm{rwl}} + t_{\mathrm{rbl}}, \tag{5.2}$$

where t_{rwl} is the time between the sampling of the read address and the arrival of the read enable at the targeted GCs; and t_{rbl} is the time from when a GC is enabled for read until the read data is ready at the output of the memory. When a write access occurs, the WWL path and the WBL path are excited in parallel; therefore, only the longest delay among the two paths contributes to t_{wr}, as described in (5.1). During a read access, the RBL path is excited only once the read address has propagated through the RWL path, and therefore, t_{rd} is equal to the sum of both associated path delays, as described in (5.2). Further details about the timing modeling of a GC-eDRAM are given below. To this end, the read path is used as an example that illustrates the delays that contribute to t_{rwl} and t_{rbl}. Similar considerations are used for the modeling of the write-path delay (t_{wr}).

The components contributing to the delay modeling of the read path in a GC-eDRAM are shown in Fig. 5.7. The value of t_{rwl} is the sum of many contributors, starting from the propagation delay of the registers that sample the input read address. The sampled address is decoded by the global address decoder, which provides the global one-hot RWL signals to the local RWL-enable logic of every GC-array. A single layer of logic could be used for the design of this address decoder, however, large fan-in gates would be required. This might be unpractical to implement, as such gates would hardly fit the WL pitch and their delay would be significantly degraded, especially at scaled voltages due to the need for many stacked transistors [40]. Hence, the common global one-hot address decoder is divided into multiple layers [40] to ensure a maximum fan-in of 2 for each implemented gate. The delay of the gates used in the decoder is tabulated as it can easily be estimated based on the characterization of available standard cells. In addition, GEMTOO is able to account for any interconnect driver that might be required within the decoder. In fact, the fan-out of the gates driving the next layer and the length of the interconnects increase with the number of the layer; therefore, drivers might be required to reduce the delay of signals propagating through these large loads. As an example, the global address decoder, shown in Fig. 5.7, is implemented with three layers of logic (e.g., 8-to-256 address decoder) and interconnect drivers are added between the second and third layer to drive the large fan-out of the third layer together with the long interconnect.

Within GEMTOO, the delay of any path that includes long interconnects is evaluted using a distributed-RC delay model (DDM), which is more accurate than a lumped-RC delay model [40]. The load of long wires is calculated taking into account the memory floorplan, which is affected by parameters such as the GC dimensions and the architectural transformations. To further improve the accuracy of the interconnect delay, the routing parasitics are specified as technology input parameters.

Large buffers drive the global RWLs, which reach every GC array of the memory. The delay of the signals propagating through these long interconnects is estimated with a DDM, where both the routing parasitics as well as the capacitive loads of the GC-array RWL inputs are included. Once the global RWLs have reached a GC array, they connect to the local RWL enable logic, which propagates the global RWL to the local RWL drivers only if the GC array is selected. As the local RWL-enable logic is implemented with a single layer of gates (e.g., AND gates), the delay of the implemented standard cell is tabulated and simply added to t_{rwl}. The delay of the local RWL drivers is the last contributor to t_{rwl} and is also estimated with a DDM due to the the long RWL interconnects, where the load capacitances associated with the RWL port of the GCs belonging to the selected word are also included.

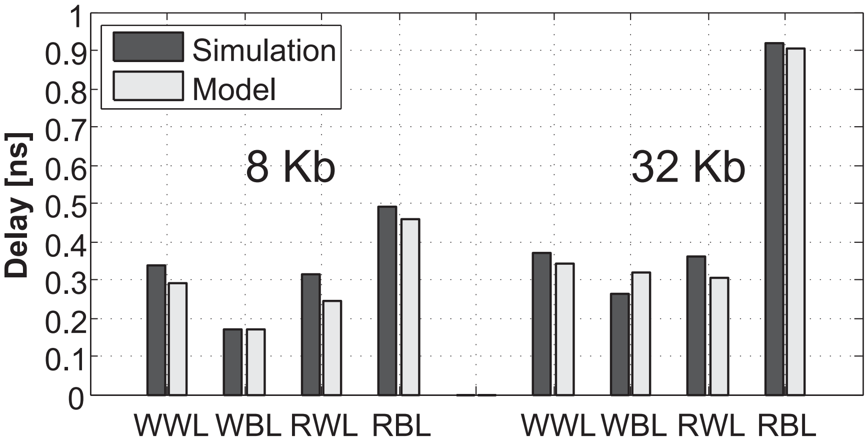
It is worth noting that the precharge of RBL is always performed at the beginning of a read access. As the RBL precharge happens in parallel with the read-address decoding and is completed in a relatively short amount of time, t_{rwl} is not affected by the RBL-precharge operation.

The first contributor to t_{rbl} is the time required by a read-enabled GC to discharge the local RBL, which is estimated by a DDM, where the capacitive loads of both the long RBL interconnect and the RBL port of the GCs connected to the same RBL are included. When computing the read-access time, the proposed tool also accounts for any degradation of the data stored in the SN of the GC due to leakage currents, which reduces the overdrive voltage of the read-access transistor and its output current. This results in an increased time required to discharge the RBL. A higher refresh rate counteracts this phenomenon, as a reduction in the time between two consequent refresh operations limits the maximum amount of degradation of the stored data, here defined as worst-case voltage degradation on SN (v_{d}). The impact of the data deterioration on timing can be evaluated by either specifying the refresh period (t_{r}) or v_{d} as input parameters. When the worst-case data degradation is specified, the tool sets the refresh rate accordingly. For example, when specifying a worst-case voltage degradation of either 20% or 30% of the original data for the 4T GC proposed in [79], the tool sets the refresh period to 74 µs and 156 µs, respectively. The corresponding output ON-resistance values of the GC for different values of data degradation are provided as inputs to the tool to account for this effect in the read-access delay with a DDM. Besides improving the accuracy in the timing estimation, this feature can be used to evaluate the impact of different refresh rates.

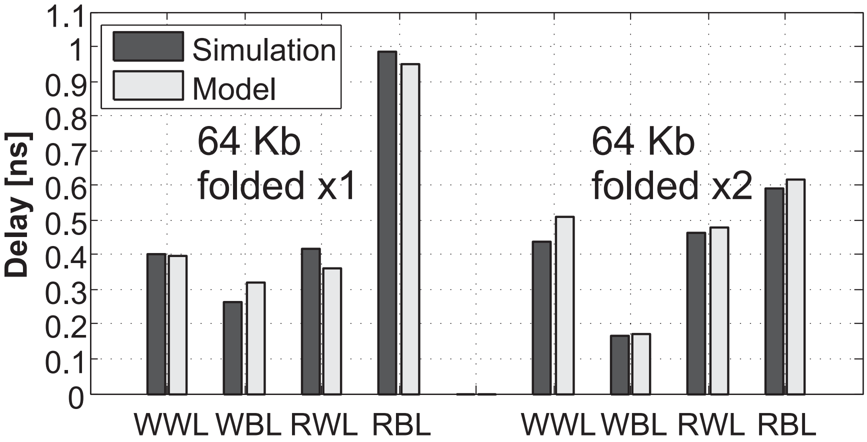
In the current model, a voltage-based sense amplifier terminates the RBLs. Its switching threshold can be defined as an input parameter for an accurate delay estimation and its propagation delay is tabulated and added to t_{rbl}. The last contributor to t_{rbl} is the the global RBL mux, which can either be implemented with tri-state buffers or with a tree of NAND-based muxes. Both models include the impact of the global RBL interconnects based on a DDM.

5.3.2 Memory Availability and Bandwidth

GC-eDRAM needs to be periodically refreshed in order to retain the stored data. However, when the refresh operation is executed, the memory cannot be accessed, resulting in limited availability. The memory availability is a key metric due to its impact on the memory band-



(a) Monolithic GC-eDRAMs with memory size of 8 kb and 32 kb.



(b) Hierarchical GC-eDRAM folded either once or twice to cut the BLs.

Figure 5.8: Comparison of the simulated and estimated delays for the timing validation of the GC-eDRAM modeling tool.

width and it depends on the amount of time required to refresh the memory content (t_s) as well as on the time between the start of two refresh periods (t_r), which represents the refresh rate. The operation to refresh the memory content consists of reading each memory word and writing it back to the same address. Therefore, t_s depends on f_max, which is estimated by the timing engine of the model as well as on the number of words in the GC array (r_b), as multiple GC arrays implemented in a hierarchical GC-eDRAM can be refreshed in parallel. Given these considerations, the memory availability (α) is derived as:

$$\alpha = 1 - \frac{t_\mathrm{s}}{t_\mathrm{r}} = 1 - \frac{r_\mathrm{b}}{f_\mathrm{max}\, t_\mathrm{r}}. \tag{5.3}$$

The derived value of α can also be included in the equation of the memory bandwidth (b):

$$b = \alpha\, f_\mathrm{max}\, c_\mathrm{b} = \left(f_\mathrm{max} - \frac{r_\mathrm{b}}{t_\mathrm{r}}\right) c_\mathrm{b}, \tag{5.4}$$

Table 5.1: Comparison between the simulated and the estimated $f_{\max}$.

Type	Folding	Size [kb]	$f_{\max}^{\text{sim}}$ [MHz]	$f_{\max}^{\text{mod}}$ [MHz]	Error [%]
Monolithic	–	8	1240	1420	15
Monolithic	–	32	780	830	6
Hierarchical	x1	64	710	760	8
Hierarchical	x2	64	950	920	4

where c_{b} is the wordsize. Note that this chapter focuses on GC arrays storing a single word per row, which ensure single-cycle access in GC-eDRAMs (see Section 5.2.3), and therefore, c_{b} represents both the number of columns within a GC array as well as the wordsize.

5.3.3 Area and Utilization

Memory density is one of the main optimization criteria for embedded memories. GEMTOO includes the estimation of the memory area as well as of other related metrics that help to identify bottlenecks and overheads. The area of the GC-eDRAM is calculated by reading the physical dimensions of its basic components as input parameters, such as the dimensions of the GC and either the width or the height of the surrounding peripheral circuits (e.g., interconnect drivers, sense amplifier, standard cells), which are considered to be pitch-matched to the WLs and BLs of the GC array. Architectural transformations are also taken into account during this process as they change the physical arrangement of the GC-eDRAM components and introduce extra global/local logic.

In addition to the physical dimensions of the memory, GEMTOO provides other relevant metrics that are key for the choice of the memory architecture. These are the memory density, which is the ratio between the memory size and the area, and the area efficiency, which is defined as the ratio between the area of the GC arrays and the total memory area. The area efficiency is a key metric for the optimization of the memory density as it quantifies the area overhead of the logic around the GCs.

5.4 Validation

The timing estimation of the proposed modeling tool is validated against transistor-level simulations of both monolithic and hierarchical GC-eDRAMs. The considered GC-eDRAMs are designed in a 28 nm FD-SOI technology and their simulations account for both resistive and capacitive parasitics taken from post-layout parasitic extractions. A typical supply voltage of 0.9 V and a boost voltage of 1.3 V for the WWL overdrive are considered for this validation.

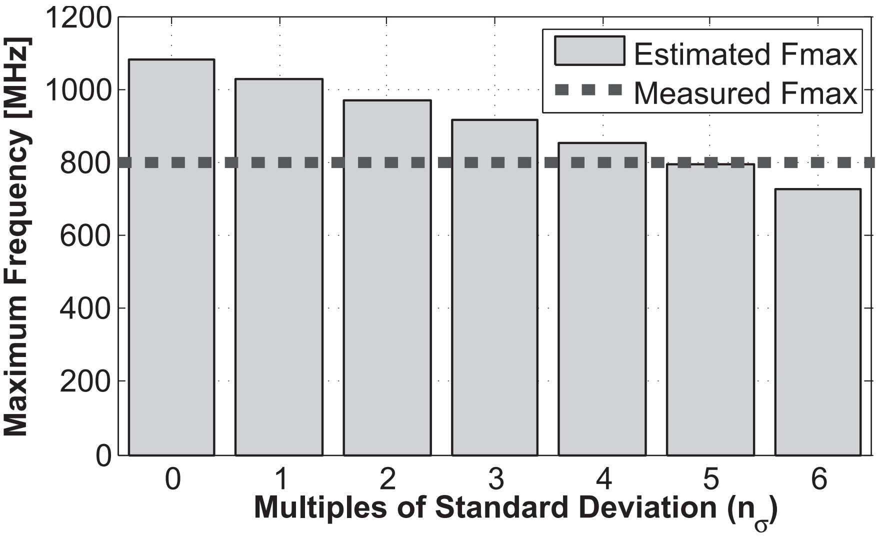


Figure 5.9: Comparison between the measured $f_{\max}$ of [80] and the $f_{\max}$ estimated with the modeling tool for different multiples of the standard deviation (n_σ).

The timing validation is performed first considering the propagation delays of each of the WL and BL paths (i.e., t_{wwl}, t_{rwl}, t_{wbl}, and t_{rbl}), as they are the main contributors to t_{wr}, t_{rd}, and therefore to the $f_{\max}$ of the GC-eDRAM. In the following analysis, t_{wsn} is not validated against simulations as its contribution to t_{wr}, and therefore to $f_{\max}$, is negligible. Nevertheless, when the model estimates the overall $f_{\max}$ of the memory, t_{wr} is taken into account for completeness.

In Fig. 5.8(a), the simulated delays of monolithic GC-eDRAMs are compared with the timing estimated by GEMTOO for two different memory sizes. The results show the timing accuracy of the modeling tool as the loads of the interconnects scale according to the memory size. A similar type of timing validation is also performed for the hierarchical GC-eDRAMs, as shown in Fig. 5.8(b). In particular, a 64 kb memory is considered, where the folding transformation for cutting the BLs is applied both once and twice. This folding transformation introduces both global-address buffering and RBL muxing, and therefore validates all aspects of a hierarchical memory. For the considered GC-eDRAMs, the estimation of the path delays provided by the modeling tool shows an average and a maximum deviation from post-layout simulations of 9% and 23%, respectively.

The timing validation of the model is also performed considering the maximum operating frequency of the GC-eDRAMs. Both the simulated and estimated $f_{\max}$ are reported in Table 5.1 as $f_{\max}^{\mathrm{sim}}$ and $f_{\max}^{\mathrm{mod}}$, respectively. Among the considered memories, results show that the maximum deviation of the estimated $f_{\max}$ is limited to 15%.

The estimation of $f_{\max}$ is also validated with silicon measurements from a 4 kb memory GC-eDRAM, fabricated in a 28 nm bulk technology [80]. This design is composed of 4T internal-feedback GCs with mixed-V_{T} transistors. As the comparison is made with a fabricated GC-eDRAM, random process variations within the GC array are considered to derive the

ON-resistance of the GC read transistor, which is an input parameter of GEMTOO. This is key for an accurate validation, since GCs are highly affected by random process variations, the GC array is composed of a large number of cells, and the read-access delay is limited by the GCs with smallest ON-current for driving the RBL. The estimated $f_{\max}$ for different multiples of the standard deviation (n_σ) is reported in Fig. 5.9 and compared with the measured $f_{\max}$. As expected, when not considering any process variation (i.e., n_σ equal to zero), the modeling tool overestimates the $f_{\max}$. However, the error on the estimated $f_{\max}$ is between +7% and -9% when process variations are considered for an n_σ equal to 4 and 6, respectively.

5.5 Design-Space Exploration

The use of GC-eDRAMs enables system architects to benefit from many advantages: high memory density, low-power operation, full logic compatibility, dual-port operation, and non-destructive read. However, the impact of the design choices for a GC-eDRAM, such as the GC topology or the memory organization, can be hard to estimate, especially in a limited amount of time, making the design optimization a complex and difficult process. By considering different memory requirements, such as the memory density, the maximum operating frequency, and the memory availability, GEMTOO can be used to rapidly perform such a design-space exploration to quickly investigate and understand different design choices. Different memory requirements are considered separately in the following subsections, in which the impact of the different design choices is considered for both monolithic and hierarchical memories and best-practice design guidelines are derived.

5.5.1 Optimization for Operating Frequency

Increasing the maximum operating frequency of a memory requires the analysis of its critical-path delay, which is either determined by t_{wr} or t_{rd}. The WWL and RWL paths have comparable delays ($t_{\mathrm{wwl}} \sim t_{\mathrm{rwl}}$), because of the similar building blocks used for their implementation, and the t_{wbl} is small due the large drive strength of the WBL driver. For this reason, the read access delay is usually more critical, as it is determined by the sum of t_{rwl} and t_{rbl}. Thus, improving $f_{\max}$ generally corresponds to design choices that aim at the reduction of the read-access delay. For effective delay optimization, the timing breakdown of the read-access delay is reported in Fig. 5.10 for different memory sizes with a monolithic GC array. The results show that, for the majority of the memory sizes, t_{rbl} generally represents the largest portion of the read delay, as it is determined by the weak drive strength of the GC read transistor. In particular, the RBL delay increases with the number of rows implemented in the GC array and reaches up to 80% for the 64 kb GC-eDRAM with 256 rows. Thus, the reduction of t_{rbl} can lead to an effective optimization of $f_{\max}$.

The deterioration of the data stored in the GC also results in a longer RBL delay since the overdrive for the read-access transistor reduces. Increasing the memory refresh rate can be an effective solution to counteract this phenomenon and to increase $f_{\max}$. However, this

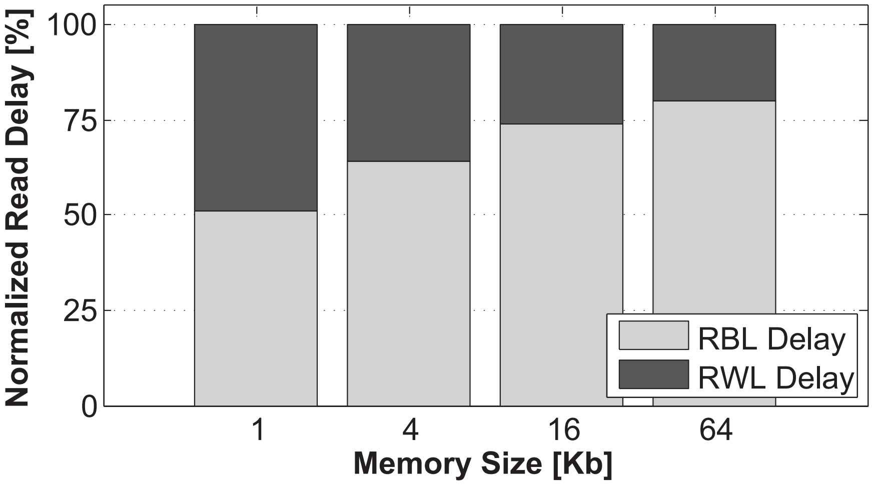


Figure 5.10: Breakdown of read delay for memories of different sizes.

measure comes at the cost of reducing the memory availability. At the circuit level, the choice of the GC topology can improve f_{max}. In the 3T GC, the RBL is discharged by two transistors in series implemented in the read port of the gain cell, while in the 2T GC, the maximum amount of current used to discharge the RBL is limited by the local buffer that drives the RWL of every GC in a GC-array row. This difference is caused by the fact that in a 3T GC the RBL is discharged directly to ground while in a 2T GC the current required to discharge the RBL is supplied by the RWL port that is driven by a local buffer. In the worst-case scenario, where every 2T GC belonging to a GC-array row is required to discharge the corresponding RBL, a large current is expected to be supplied by the shared RWL buffer. As the drive strength of the local RWL buffer is often limited by area constraints, the 2T GC is usually slower than the 3T GC in discharing the RBL. In addition, the implementation of the 2T GC is more complex due to the RBL-saturation problem [74], which is eliminated by the 3T GC.

The impact on f_{max} for different memory sizes, when either the 2T GC or the 3T GC is used, and for different refresh periods (t_{r}) is illustrated in Fig. 5.11. For this analysis, the values of t_{r} are chosen based on the worst-case degradation of the stored data (v_{d}) due to leakage deterioration at 85°C. The considered values of degradation are either 20% or 30% of the original data, which correspond to refresh periods of 0.48 µs and 3.98 µs, respectively. Considering the reported results, f_{max} is increased on average by 2× when opting for the 3T GC instead of the 2T topology. Furthermore, an average f_{max} increase of 29% is achieved by reducing the SN voltage degradation limit from 30% to 20% at the cost of a more frequent refresh.

Besides the change of the physical aspect ratio, the improvement of f_{max} is one of the main goals of architectural transformations. In fact, architectural transformations can have a significant impact on the internal delays of the memory, as illustrated in Fig. 5.12, where the delay breakdown of each of the memory paths for an 8 kb GC-eDRAM is shown. In particular, the considered memory is implemented with either a monolithic GC array, shown

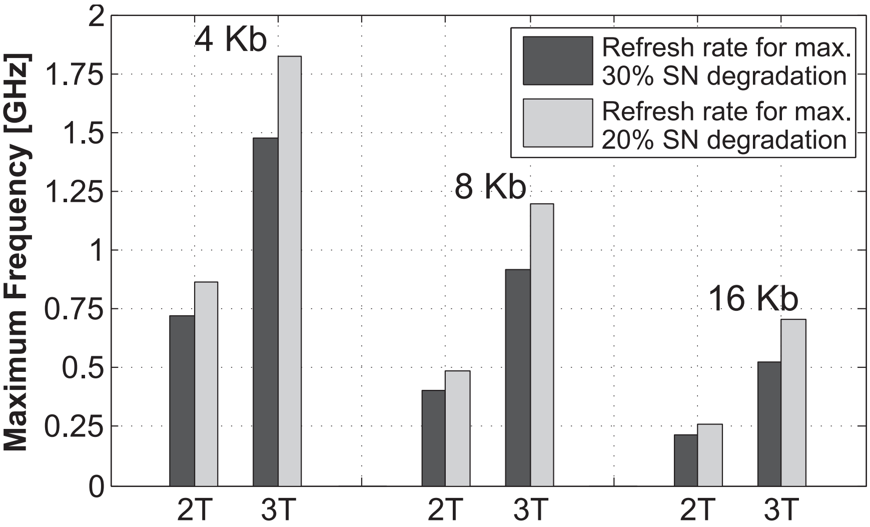


Figure 5.11: Maximum operating frequency for different GC topologies and refresh rates.

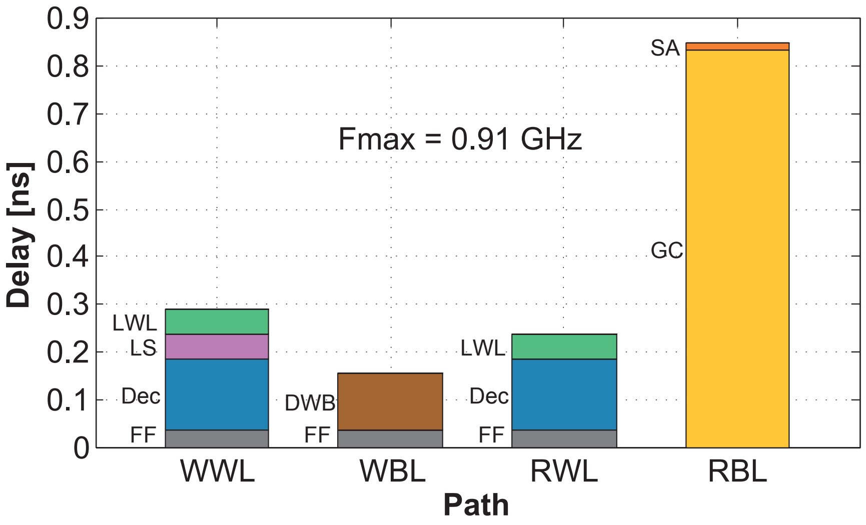
in Fig. 5.12(a), or with a hierarchical organization, where a folding transformation cuts the BLs once, as shown in Fig. 5.12(b).

The main timing benefit of both folding and partitioning when cutting the BLs is the reduction of the RBL length, which significantly shortens the RBL delay, and therefore, the read access delay. The WBL delay is also shortened as the WBLs are reduced in length. However, these architectural transformations also result in some timing overhead due to the introduction of global address buffers and RBL muxing, as shown for GWL and Mux in Fig. 5.12(b). Nevertheless, this timing overhead is limited and the large reduction of the RBL delay results in an overall improvement of $f_{\max}$ as the read access delay is on the critical path. This is confirmed by the monolithic memory of Fig. 5.12(a) which achieves an $f_{\max}$ of 0.91 GHz, while the folded implementation of Fig. 5.12(b) can be operated up to 1.35 GHz, showing an $f_{\max}$ improvement of 47% when the RBLs are cut in half.

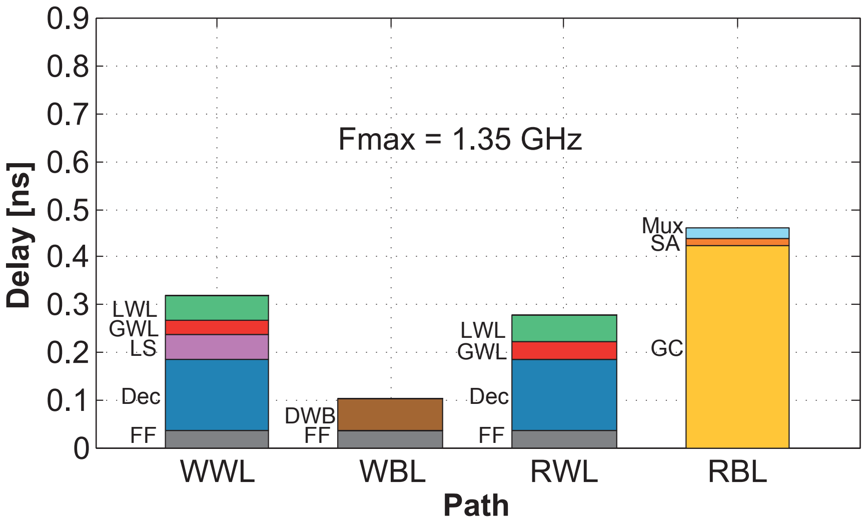
As the architectural transformations have a large impact on the timing of GC-eDRAMs, several memory organizations have been explored for different large-size memories and the options that maximize $f_{\max}$ are reported in Table 5.2. The considered design space is for GC-eDRAMs with sizes of either 128 kb or 1 Mb and a word size of either 64 or 256 bits. The presented results have been filtered to consider only memory footprints with a minimum area efficiency of 70% and with width/height aspect ratio between 0.25 and 4 to avoid awkward dimensions which are unpractical to integrate.

5.5.2 Optimization for Memory Availability and Bandwidth

Dynamic memories require periodic refresh, resulting in cycles in which the memory content cannot be accessed. The memory availability impacts the memory bandwidth and depends



(a) Monolithic implementation.



(b) Hierarchical implementation where the folding transformation cutting the BLs is applied once.

Figure 5.12: Timing breakdown for each path of a 8 kb GC-eDRAM. The considered GC-eDRAM is either implemented as monolithic or hierarchical memory. The blocks contributing to the delay of each of the paths are: flip-flop (FF), global address decoder (Dec), level shifter (LS), local WL-enable logic and WL buffer (LWL), WBL driver (DWB), GC read port (GC), sense amplifier (SA) and RBL mux (Mux).

on the refresh rate, on the $f_{\max}$ of the memory, and, as all the GC arrays within a hierarhical GC-eDRAM are refreshed in parallel, on the number of rows in each GC array, as previously described in (5.3). Even though it is clear on which parameters the memory availability depends, its maximization is not trivial as all parameters depend on each other. Finding their optimal combination is therefore a complex task. GEMTOO not only helps to understand

Table 5.2: Memory organizations that maximize f_{max}.

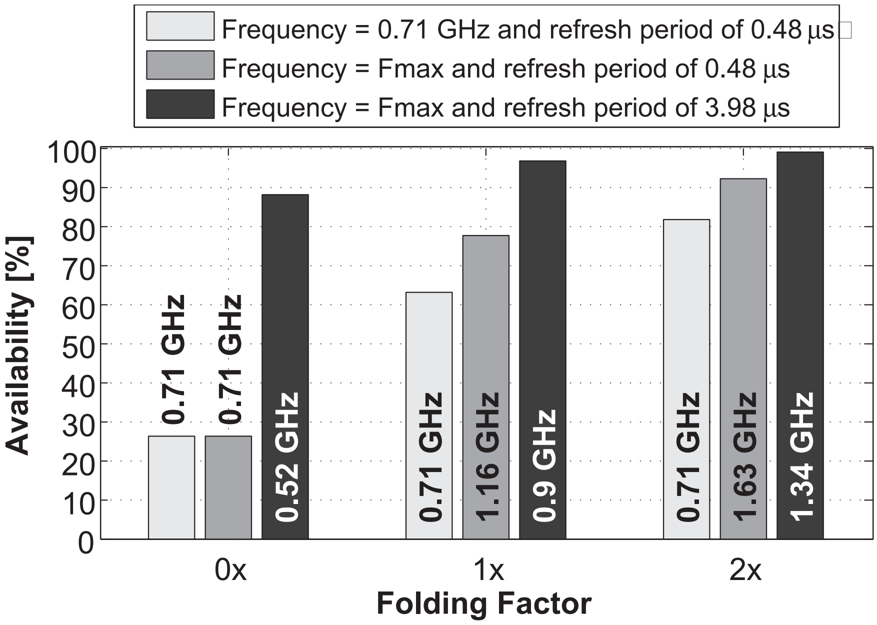
Size [kb]	Wordsize [bit]	$r_{\mathrm{a}} \times c_{\mathrm{a}}$	$r_{\mathrm{b}} \times c_{\mathrm{b}}$	f_{max} [MHz]
128	64	2×8	256×32	480
128	256	8×8	256×8	1020
1024	64	4×64	256×16	620
1024	256	16×32	128×16	710

these dependencies, but it can also be used to perform a rapid exhaustive search across the parameter space to optimize the memory availability.

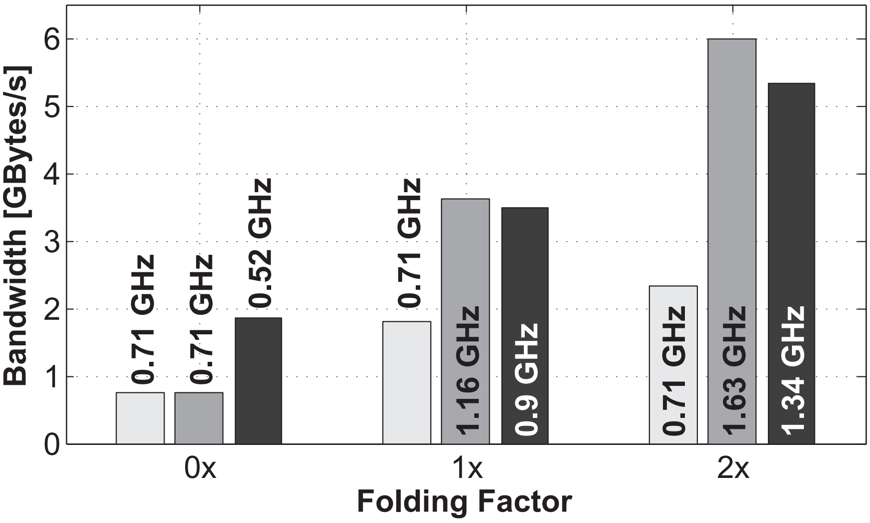
For example, the memory availability of a GC-eDRAM designed with different folding factors is shown in Fig. 5.13(a). The monolithic implementation has an f_{max} of 0.71 GHz and a memory availability of 25%, when a refresh period of 0.48 µs is specified. Folding transformations increase the memory availability as multiple GC arrays can be refreshed in parallel. The corresponding estimated memory availabilities are 62% and 81% for a folding factor of 1× and 2×, respectively, when the operating frequency remains unchanged. This assumption is relevant in cases where the operating frequency limit is not determined by the memory but by other system components.

However, folding transformations also cut the BLs of the GC array, which allows to operate at higher frequencies. In the example of Fig. 5.13(a), the single-folded memory can operate up to 1.16 GHz and the the 2× folded memory, up to 1.63 GHz, which further increases both the availability and the memory bandwidth as reported in Fig. 5.13(a) and Fig. 5.13(b), respectively.

In addition to the application of folding transformations, the refresh rate can also be adjusted to maximize the memory availability. However, the optimization of t_{r} for a higher availability is non-trivial because any change in the refresh rate has contradicting effects on the availability. On the one hand, refreshing the memory more often might limit its availability. On the other hand, an increase in the refresh rate allows for a higher f_{max} (due to less SN voltage degradation) and a higher operating frequency, which has a positive impact on availability and bandwidth. The proposed tool can be used to find the best refresh rate that maximizes the availability. Fig. 5.13(a) depicts the memory availability when specifying a refresh period of either 0.48 µs or 3.98 µs for different GC-eDRAMs. Results show that the memory availability is maximized when reducing the refresh rate, even if the memory is operated at a lower f_{max}. However, the resulting f_{max} penalty still degrades the bandwidth, as shown in Fig. 5.13(b). In fact, while the bandwidth of the monolithic GC-eDRAM benefits from a reduced refresh rate, refreshing the folded implementations less often leads to a reduced bandwidth as the f_{max} penalty is larger than the gain in availability.



(a) Memory availability.



(b) Memory bandwidth.

Figure 5.13: Impact of folding, refresh rate and operating frequency on memory availability and bandwidth.

At the circuit level, the choice of the GC topology is key when optimizing for a higher memory availability as different GCs require different refresh rates. For example, the 4T GC [79] has a longer retention time than the conventional 2T GC, and therefore, it allows for lower refresh rates, which lead to higher memory availability. The effect of different GC topologies is shown in Table 5.3, where the GC-eDRAMs based on the 4T GC show a memory availability always greater than 99%. The highest bandwidth (1.6 GB/s) is also achieved by the memory based

Table 5.3: Memory availability (α) and bandwidth (b) for different GCs and refresh rates.

GC	t_r [µs]	v_d [%]	f_{max} [MHz]	α [%]	b [GB/s]
2T	0.48	20	480	72.8	1.4
2T	3.98	30	390	95.9	1.5
2T	19.24	40	280	98.8	1.1
4T	74.6	20	420	99.7	1.6
4T	156.7	30	340	99.8	1.3
4T	397.9	40	240	99.9	0.9

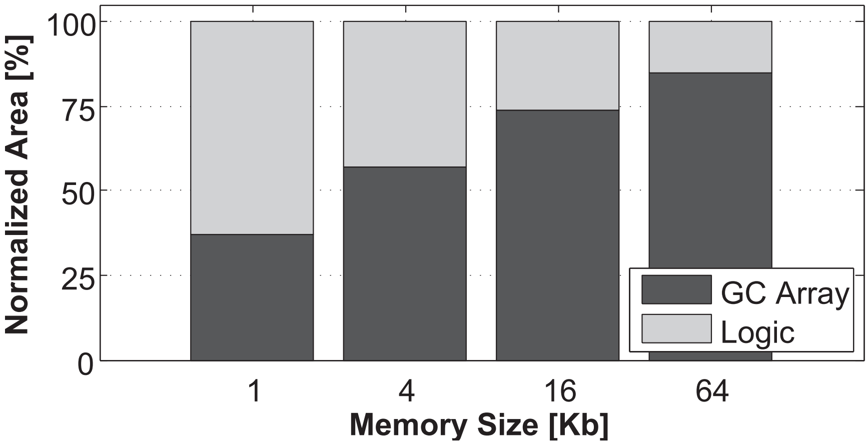


Figure 5.14: Area efficiency for single-array memories of different memory sizes.

on 4T GCs when the most frequent refresh is specified (74.6 µs), as the bandwidth is more influenced by the increase in f_{max} than by the overhead due to the more frequent refresh. It is also worth noting that f_{max} is slightly lower when the 4T GC is preferred over the 2T GC as the 4T GC has a 56% larger area, which results in longer BLs of the GC array and therefore in a slower read access.

The analysis of the memory organization that maximizes the bandwidth is extended to different memory sizes and GCs. Table 5.4 summarizes the results. As expected, larger memory sizes result in lower bandwidth due to the slower access time. The 4T-based memories always show a larger bandwidth than the memories implemented with 2T GCs. However, the difference in bandwidth among the implementations with different GCs is reduced for larger memory sizes.

Table 5.4: Memory organizations that maximize availability (α) and bandwidth (b).

Size [kb]	GC	$r_a \times c_a$	$r_b \times c_b$	f_{max} [GHz]	α [%]	b [GB/s]
128	2T	2×32	64×32	0.40	67.5	8.8
128	4T	8×8	64×32	0.38	99.7	12.2
1024	2T	16×32	64×32	0.38	65.6	8.0
1024	4T	32×16	64×32	0.34	99.7	10.8

Table 5.5: Impact of the GC topology on the area of monolithic memories.

Size [kb]	Memory Area [μm^2]			Area Reduction [%]		
	4T	3T	2T	4T	3T	2T
1	552	474	418	–	14	24
4	1523	1267	1085	–	16	28
16	4910	3994	3352	–	18	31
64	17464	14019	11621	–	19	33

5.5.3 Optimization for Memory Density

One of the main drivers for GC-eDRAMs is the need for high memory density, which mostly depends on the area efficiency, as previously described in Section 5.3.3. In fact, the area efficiency shows which of the memory blocks should be optimized for an effective increase in memory density. For this reason, the area efficiency for monolithic memories of different sizes is reported in Fig. 5.14, The figure shows that the area occupied by the GC array increases with the memory size, and in particular, the GCs use more than 50% of the total area when the memory size is larger than 4 kb. As the GC arrays, even in hierarchical memories, are most often a few kb, the area optimization of the GC is usually the key to maximize memory density.

GCs are generally characterized by their transistor count, because of its impact on the overall memory area. Considering that a 2T GC and a 3T GC are 35% and 20% smaller than a 4T GC in 28 nm FD-SOI, respectively [79], the area of monolithic memories of different sizes is reported in Table 5.5 to quantize the area savings based on the chosen GC topology. The results show the impact of area efficiency, as well. In fact, when opting for the 2T GC instead of the 4T topology, a 24% and 33% memory-area reduction is obtained for a size of 1 kb and 64 kb, respectively, where the 64 kb memory has the highest area efficiency. Even though the 2T GC provides the highest memory density, it is important to remember that the 4T GC has a significantly longer retention time, which improves the memory availability.

Table 5.6: Most area-efficient memory organizations for hierarchical memories.

Size [kb]	$r_a \times c_a$	$r_b \times c_b$	Memory Area [mm^2]			Reduction [%]		
			4T	3T	2T	4T	3T	2T
32	1×8	128×32	0.011	0.009	0.007	–	17	29
128	1×16	256×32	0.040	0.033	0.027	–	18	31
512	2×32	256×32	0.159	0.130	0.109	–	18	31
2048	4×64	256×32	0.632	0.514	0.432	–	18	31
8192	8×128	256×32	2.517	2.048	1.718	–	18	31

When focusing on monolithic memories, any architectural transformation that partitions the memory results in an area overhead, as additional logic is required. However, these transformations are mandatory for large-size memories to avoid long access times or awkard aspect ratios. In this section, GEMTOO is used to determine the most area-efficient memory organizations. The area optimization is performed for different memory sizes and the results are reported in Table 5.6. The presented results have been filtered to consider only memory floorplans with aspect ratios bounded between 0.25 and 4 to avoid very large dimension ratios, which might be impractical to integrate. Also, the number of rows per GC array is set to be less than 256, to avoid very long read access delays. The results show that distributing many GC arrays per row by folding is generally preferred, as it results in a lower area overhead as compared to partitioning. Nevertheless, partitions are also applied to increase the number of arrays per column (r_a), which fulfills the requirement on the maximum number of rows per GC array (r_b), while keeping the dimension ratio within the given constraints.

5.6 Conclusion

The integration of compact and low-power embedded memories is key for the design of energy-efficient SoCs. Among the different types of memories, GC-eDRAM is a high-density, low-leakage, and fully logic-compatible embedded memory that proved to be a valid low-power alternative to conventional SRAM. To facilitate and speed up both the design and the integration of GC-eDRAMs in computing systems, GEMTOO, the first modeling tool of GC-eDRAMs, was proposed in this chapter. The described tool enables fast scaling and optimization of GC-eDRAMs despite the large and complex design space, whose exploration typically requires several simulations as any memory metric relies on many interdependent variables. A highly accurate estimation of timing, memory availability, bandwidth, and area of GC-eDRAMs is achieved by modeling both the architecture and the device level of the memory. In particular, GEMTOO accounts for the physical representation of the memory, the impact of the memory organization, and the load of the interconnects as part of the model. At the device level, the described tool rigorously models GC-eDRAMs by considering the influence of the chosen refresh rate on both memory availability and timing, as the deterioration of the stored data significantly impacts the read access delay. The validation of the tool shows that the maximum operating frequency is estimated with a maximum deviation of 15% and 9% for different simulated memory designs and for a fabricated GC-eDRAM in 28 nm, respectively.

6 Conclusions and Outlook

Almost all design techniques for energy efficiency are ultimately limited by some form of a more or less controlled degradation of the quality of service (QoS). Thus, a thorough understanding of the circuits behaviour when operated *around* the point of first failure (PoFF), which traditionally identifies where an entire system stops working properly, is key for an effective application of design methods for energy efficiency as well as for the development of future low-energy techniques. A set of circuits, techniques and tools are proposed in this thesis to lower the energy consumption in digital very-large-scale integration (VLSI) systems when operated either in the safe and conventional exact region, *close* to the PoFF, or even in the inexact region whenever a small degradation on the QoS can be tolerated. In this final chapter, conclusions are provided for each of the tools and approaches described in the previous chapters. Furthermore, an outlook on future directions and possible improvements for the presented work is provided.

Dual-Edge-Triggered Clocking for Low-Power Operation

This thesis presented the tradeoffs involved in the choice between single-edge-triggered (SET) and dual-edge-triggered (DET) clocking for low power consumption and the power benefits of DET clocking were demonstrated on three case studies in a 40 nm CMOS process technology, showing power savings as high as 58% for a register-heavy design when compared to a corresponding SET implementation. The power reduction in the clock distribution is enabled by the proposed design flow that automatically implements DET clocking in synchronous digital designs. The key to the presented methodology is to initially design, test, and implement all components as standard SET blocks, and subsequently replace the SET clocking elements with equivalent DET gates after synthesis, before continuing to automated place-and-route (APR). With implementations of both clocking schemes at hand, the power-area tradeoff can be evaluated for each block and the best clocking strategy can be chosen for a mixed-integration of SET and DET clocking within a single system. The described methodology is one of the first works to directly address the intricacies of the physical implementation of DET designs, especially in the light of clock-gating insertion.

A DET flip-flop (DET-FF) topology that solves the inherent clock-overlap risk of DET-FFs implementing transmission gates as output multiplexer (MUX) has also been presented in this thesis. The failure risk due to clock-overlap was demonstrated on a popular transmission-gate latch-MUX (DET-TGLM) register, showing an unacceptable error-rate at near-threshold voltages in a 40 nm CMOS process technology. The proposed fully-static true-single-phase-clock (TSPC) DET-FF was shown to be fully functional at a similar operating point, under local and global process variations, and at a wide range of temperatures. In addition, the proposed cell was found to provide the best CK-to-Q delay and power-delay product among popular static DET-FFs.

Circuits and Techniques to Monitor and Trim Dynamic Timing Margins

As the critical path of a synchronous design might not always be excited, dynamic timing margins exist and they can be exploited by dynamic clock adjustment (DCA). In this context, a dynamically-adjustable clock generator (DCG) is proposed in this thesis to enable the application of the DCA approach whose benefits are demonstrated on a microprocessor fabricated in 28 nm FD-SOI, showing a maximum throughput increase of 41% or up to 15% power savings when DCA is applied. Immediate and glitch-free changes on the frequency of the clock signal generated by the DCG are ensured by internally double-sampling the input delay setting and by a careful design and controlled layout of the cells involved in timing-critical paths. The DCG has been characterized with measurements on a 28 nm FD-SOI test chip, showing an output-frequency range from 365 MHz to 1906 MHz with an average step granularity of 34 ps at 1.0 V.

In order to monitor dynamic timing margins as well as potential timing violations, a timing-monitoring sequential (TMS) capable of detecting transitions on either the high or the low phase of the clock is described in this thesis. Controlling the duty-cycle of the clock, the proposed circuit enables the detection of either timing violations or reductions on any available positive timing slack while ensuring error-free operation. The desired timing-analysis mode can be chosen at runtime and simulations show the TMS to operate at 0.9 V with either a high or a low clock phase as short as 90 ps and 140 ps, respectively. Both detection modes of the proposed TMS were verified on a fabricated test chip in a 28 nm FD-SOI process technology. Silicon measurements demonstrated that the TMS is capable to measure the maximum frequency of a digital multiplier at runtime, with an accuracy of 1%, and without incurring in any timing violation.

Exploiting Hardware Properties at the Algorithm Level for Energy-Quality Scaling

Several limitations of voltage over-scaling (VOS) have been demonstrated in this thesis: the tree structure of timing-critical circuits results in a steep degradation of the QoS as soon as the first timing errors occur, the best circuit candidates for VOS are mainly slow circuits based on ripple-carry adders (RCAs) as fundamental arithmetic operators which can degrade the maximum operating frequency of the system, and the prediction of the impact on quality degradation due to timing errors is highly complex. Given these critical limitations of VOS, an approximate-computing technique that reduces the switching activity in programmable hardware has been proposed to achieve further power savings in addition to the application of conventional voltage scaling. The proposed method relies on the observation that the power consumption of multipliers having one input port connected to a constant-coefficient operand can significantly vary with the value of that fixed operand. In architectures where coefficients are programmable but constant over many cycles, the described power property of multipliers can be exploited by replacing the original coefficients with similar coefficients that lead to lower power consumption. This technique can effectively be applied, as an example, to reconfigurable finite impulse response (FIR) accelerators. A simple greedy algorithm is used to modify the coefficients of a baseline filter to derive a new set of coefficients that are optimized for low power consumption while allowing for some degradation of the filtering quality. The proposed approximate-computing technique does not require any design overhead for a programmable accelerator as it exploits the flexibility of the algorithm level. At the same time, it ensures the quality of the baseline filter whenever it is required, while it offers also the possibility of scaling the power consumption at runtime when only a little energy is available and reduced accuracy is tolerated. The presented technique has been demonstrated on both simulated and fabricated FIR accelerators implemented in a 28 nm FD-SOI process technology, showing dynamic power savings of up to 33% for an accepted 3 dB degradation on the stopband attenuation of a filter.

Gain-Cell Embedded DRAMs: Modeling and Design-Space Exploration

In the light of the growing demand for data-intensive applications, the integration of compact and low-power embedded memories is key for the design of energy-efficient system-on-chips (SoCs). Among the different types of memories, gain-cell embedded DRAM (GC-eDRAM) is a high-density, low-leakage, and fully logic-compatible embedded memory that proved to be a valid low-power alternative to SRAM and to be faster than conventional DRAM. Furthermore, additional energy savings can be traded for a small potential loss of data in GC-eDRAMs by reducing the refresh rate below the pessimistic worst-case limit, as GC-eDRAMs gracefully degrade when operated in the inexact region. In order to facilitate and speed up both the design and the integration of GC-eDRAMs in computing systems, GEMTOO, the first modeling tool of GC-eDRAMs, was proposed in this thesis. The described tool enables fast scaling and optimization of GC-eDRAMs despite the large and complex design space, whose exploration typically requires several simulations as any memory metric relies on many interdependent variables. A highly accurate estimation of timing, memory availability, bandwidth, and area of GC-eDRAMs is achieved by modeling both the architecture and the device level of the memory. In particular, GEMTOO accounts for the physical representation of the memory, the impact of the memory organization, and the load of the interconnects as part of the model. At the device level, the described tool rigorously models GC-eDRAMs by considering the influence of the chosen refresh rate on both memory availability and timing, as the deterioration of the stored data significantly impacts the read access delay. The validation of the tool shows that the maximum operating frequency is estimated with a maximum deviation of 15% and 9% for different simulated memory designs and for a fabricated GC-eDRAM in 28 nm, respectively.

Outlook

A thorough knowledge of how circuits behave when operated at the boundary between the exact and inexact regions, where the PoFF lies, is key for the development of effective design methodologies for energy efficiency. To this end, this thesis proposed and described circuits, techniques, and tools to support the implementation of designs where the energy budget is limited. Possible improvements on the considered topics and an outlook on future design techniques are suggested in the following paragraphs.

As described in Chapter 2, the insertion of DET clocking is often relegated to small and well-confined parts of a system, mainly for the perceived complexity of its integration in a conventional design flow. The development of dedicated DET standard-cell libraries and memories as well as the integration of the DET-clocking design flow in conventional electronic design automation (EDA) tools could facilitate the adoption of such clocking scheme by a larger population of designers, leading to further developments on the topic. Moreover, as the leakage power is one of the main bottlenecks in DET clocking, circuit-level improvements in the design of low-leakage DET flip-flops could improve the achievable power savings when DET clocking is used.

The savings on both execution time and power consumption that are enabled by the DCA approach have been described in Chapter 3 for an embedded processor. The investigation of path-delay profiles should be extended to a larger population of circuit architectures (e.g., accelerators, digital signal processing (DSP) modules, memories, etc.) to potentially highlight dynamic timing margins in other design candidates that could benefit from the the application of the DCA approach.

As the calibration of the frequency settings for the DCA approach is most likely to be performed for only *few* operating points, the addition of guard bands is often required to ensure the correct functionality of the design across *any* operating point. The amount of added guard bands might be reduced by augmenting the programmable delay lines (PDLs) integrated in the DCG with replica circuits [41], to adaptively compensate the generated set of clock frequencies for dynamic variations.

The application of error-correction techniques often comes at the cost of a significant loss for the performance of the sytem (e.g., stalling of the pipeline). Thus, when considering the application of the proposed TMS, the development of further techniques that rely on the measurement of the available positive slack is key to ideally eliminate the need of error-correction techniques. At the circuit level, design optimization of the TMS is also required to reduce its large design overhead over traditional flip-flops.

In Chapter 4, an energy-quality scaling technique based on the properties of the underlying hardware has been proposed. A simple and greedy algorithm is used to modify the coefficients of a baseline filter to derive a new set of power-optimized coefficients. Even though this approach already provides significant power savings, the development of more advanced and application-tailored algorithms could further improve the energy savings. The proposed energy-quality scaling technique could also be applied to other reconfigurable and multiplication-intensive designs. For example, convolutional neural networks (CNNs) [101] are potential good candidates as they often consume a relatively large amount of energy in multiplying data by constant coefficients (i.e., the weights of the network).

GEMTOO, the modeling tool proposed in Chapter 5, estimates timing, memory availability, bandwidth, and area of GC-eDRAMs. Including the estimation of the energy consumption to the provided performance metrics would be an important next step to further the exploration of the memory design space. In addition, the memory availability and bandwidth of GC-eDRAMs, can be increased by reducing the refresh rate below the critical limit, therefore operating the memory in the inexact region. Thus, the estimation of these memory metrics provided by GEMTOO could be used in a larger framework that evaluates with statistical analysis how memory errors impact the quality of real-life embedded applications, for the optimization of GC-eDRAMs operating in the inexact region. Research in this direction is already ongoing.

To conclude, given the results of the work described in this thesis, two main research directions are identified among the future and most promising techniques for energy efficiency. The fundamental limit that differentiates the envisioned directions coincides with the choice on the error tolerance.

Adaptive and Safe Techniques for Energy Efficiency Any excess in energy consumption can potentially be eliminated by the dynamic adaption to the operating conditions of a system. This approach typically requires to operate very close to the PoFF with a high risk of errors. Even though the correct operation of the system can be ensured with error-correction mechanisms, these techniques often come with undesirable and large overheads (e.g., throughput loss due to pipeline stalling). As an alternative, while dynamically adapting to the operating conditions, low-energy techniques for error-free systems should preferrably *anticipate* the potential generation of errors, for example through slack monitoring rather than setup-violation monitoring, to prevent any costly correction mechanism.

Stochastic Computing The traditional von Neumann approach relies on the fundamental choice that both computational errors as well as any stochastic phenomenon at either the circuit or the device level should be suppressed by deterministic computing platforms [102]. The elimination of such thorough and conservative control on the error can improve the state of the art of energy-efficient designs through the realization of statistical computing platforms that rely on stochastic components. For example, GC-eDRAMs are potential good candidates among stochastic storage components, due to their energy efficiency, low silicon footprint, and graceful degradation when operated in the inexact region. As one of the main challenges, an accurate prediction of the impact of errors is a complex task, even if this is only based on statistics. In this context, a cross-layer analysis and optimization from the computational algorithms to the fundamental storage devices of stochastic memories is key for an effective reduction of the energy consumed per operation.

A Power Tradeoffs in Dual-Edge-Triggered Clocking

In general, DET clocking is more power efficient than conventional SET clocking if the following inequality is met:

$$D^{\text{tot}} < S^{\text{tot}}, \tag{A.1}$$

where D^{tot} and S^{tot} are the total power consumptions of the same design implemented with DET and SET clocking, respectively. Assuming a synchronous digital block compiled exclusively of standard-cell based logic, the total power consumption can be written as:

$$S^{\text{tot}} = S^{\text{tot}}_{\text{tre}} + S^{\text{tot}}_{\text{reg}} + S^{\text{tot}}_{\text{log}}, \tag{A.2}$$

where $S^{\text{tot}}_{\text{tre}}$, $S^{\text{tot}}_{\text{reg}}$, and $S^{\text{tot}}_{\text{log}}$ are the power consumption of the clock tree, the registers, and the logic gates, respectively. The total power consumption of these digital gates can be further divided into two major categories: static (leakage) power consumption and dynamic power consumption that depends on the switching frequency. Considering this distinction, (A.2) can be reformulated as:

$$\begin{aligned} S^{\text{tot}} &= S^{\text{lkg}}_{\text{tre}} + f_{\text{set}}\, K_{\text{tre}}\, E^{\text{dyn}}_{\text{tre}} \\ &+ S^{\text{lkg}}_{\text{reg}} + f_{\text{set}}\, K_{\text{reg}}\, E^{\text{dyn}}_{\text{reg}} \\ &+ S^{\text{lkg}}_{\text{log}} + f_{\text{set}}\, K_{\text{log}}\, E^{\text{dyn}}_{\text{log}}, \end{aligned} \tag{A.3}$$

where f_{set} is the SET clock frequency while K_{tre}, K_{reg}, and K_{log} are the activity factors of the clock tree, registers, and logic, respectively and they depend on the considered design, as well as on the clock-gating efficiency. The remaining components are $S^{\text{lkg}}_{\text{tre}}/S^{\text{lkg}}_{\text{reg}}/S^{\text{lkg}}_{\text{log}}$ and $E^{\text{dyn}}_{\text{tre}}/E^{\text{dyn}}_{\text{reg}}/E^{\text{dyn}}_{\text{log}}$ which are the clock tree/registers/logic leakage power consumption and dynamic energy of a single transition, respectively.

Since SET flip-flops (SET-FFs) will be replaced by DET-FFs, the internal power of the registers might significantly vary depending on the transistor-level design of the implemented DET sequentials, while the output load remains mostly unchanged. It is worth to explicitly distinguish the internal from the switching energy. To this end, $E_{\text{reg}}^{\text{dyn}}$ can be reformulated as:

$$E_{\text{reg}}^{\text{dyn}} = E_{\text{reg}}^{\text{int}} + E_{\text{reg}}^{\text{sw}}, \tag{A.4}$$

where $E_{\text{reg}}^{\text{int}}$ and $E_{\text{reg}}^{\text{sw}}$ are the registers internal and switching energy of a single transition, respectively. Substituting (A.4) into (A.3), we obtain:

$$\begin{array}{rcl} S^{\text{tot}} & = & S_{\text{tre}}^{\text{lkg}} + f_{\text{set}}\, K_{\text{tre}}\, E_{\text{tre}}^{\text{dyn}} \\ & + & S_{\text{reg}}^{\text{lkg}} + f_{\text{set}}\, K_{\text{reg}}\, (E_{\text{reg}}^{\text{int}} + E_{\text{reg}}^{\text{sw}}) \\ & + & S_{\text{log}}^{\text{lkg}} + f_{\text{set}}\, K_{\text{log}}\, E_{\text{log}}^{\text{dyn}}. \end{array} \tag{A.5}$$

These equations are valid for any digital system, therefore they can be similary written to define the total power consumption of a design implemented with DET clocking. For this case, it is possible to define the DET total power consumption as the sum of contributors that are proportional to each contribution to the sum in (A.5):

$$\begin{array}{rcl} D^{\text{tot}} & = & \Omega_{\text{tre}}\, S_{\text{tre}}^{\text{lkg}} + \Phi_{\text{tre}}\, \frac{1}{2}\, f_{\text{set}}\, K_{\text{tre}}\, E_{\text{tre}}^{\text{dyn}} \\ & + & \alpha_{\text{reg}}\, S_{\text{reg}}^{\text{lkg}} + f_{\text{set}}\, K_{\text{reg}}\, (\beta_{\text{reg}}\, E_{\text{reg}}^{\text{int}} + \gamma_{\text{reg}}\, E_{\text{reg}}^{\text{sw}}) \\ & + & \Omega_{\text{log}}\, S_{\text{log}}^{\text{lkg}} + \Phi_{\text{log}}\, f_{\text{set}}\, K_{\text{log}}\, E_{\text{log}}^{\text{dyn}}, \end{array} \tag{A.6}$$

where the dynamic energy of clock buffers is multiplied by $\frac{1}{2}$ to represent the impact of the correspondingly reduced DET clock frequency. The scaling factors Ω_{tre} and Ω_{log} are used to define the DET leakage power as as proportional quantity of the SET leakage power for the clock tree and the logic, respectively. Similarly, Φ_{tre} and Φ_{log} are the scaling factors of the dynamic energy for the clock tree and the logic, respectively. Even if these scaling factors depend on how the SET and DET designs are implemented through the standard digital design flow, in a first approximation they can be considered to be equal to one since both the clock tree and the logic are expected to be very similar in both designs. Considering the registers, α_{reg} is the scaling factor between the SET and the DET registers leakage power while the scaling factors for the internal and the switching energy are β_{reg} and γ_{reg}, respectively. While α_{reg} and β_{reg} solely and highly depend on the power consumption of the implemented DET-FFs as compared to the SET-FFs, γ_{reg} can also be approximated to one since, in a first approximation, we can assume that the fan-out of the registers will remain the same for both the SET and the DET designs. The values of α_{reg} and β_{reg} are obtained considering the average performance of the SET and DET registers with transistor-level simulations. In particular, β_{reg} results from the comparison between the internal energy consumed by the SET-FFs in a SET clock period (due to both rising and falling edge on the clock port) and the average internal energy consumed by the DET-FFs in the presence of either a rising or a falling clock edge.

With the definitions of the total power consumption of both SET and DET implementations in (A.5) and (A.6), respectively, it is possible to substitute their values in (A.1) to obtain:

$$\begin{gathered} f\{(1-\tfrac{1}{2}\Phi_{\text{tre}})K_{\text{tre}}E_{\text{tre}}^{\text{dyn}} + [(1-\beta_{\text{reg}})E_{\text{reg}}^{\text{int}} + (1-\gamma_{\text{reg}})E_{\text{reg}}^{\text{sw}}]K_{\text{reg}} + (1-\Phi_{\text{log}})K_{\text{log}}E_{\text{log}}^{\text{dyn}}\} > \\ (\Omega_{\text{tre}}-1)S_{\text{tre}}^{\text{lkg}} + (\alpha_{\text{reg}}-1)S_{\text{reg}}^{\text{lkg}} + (\Omega_{\text{log}}-1)S_{\text{log}}^{\text{lkg}}, \end{gathered} \tag{A.7}$$

where the dynamic power savings of DET clocking and the relative increase in leakage power are expressed on the left and right sides of (A.7), respectively.

Bibliography

[1] "The Route to a Trillion Devices," ARM, Tech. Rep., 2017. [Online]. Available: https://community.arm.com/iot/b/blog/posts/white-paper-the-route-to-a-trillion-devices

[2] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge, "Near-Threshold Computing: Reclaiming Moore's Law Through Energy Efficient Integrated Circuits," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 253–266, Feb 2010.

[3] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, Jan 2008.

[4] A. Teman, L. Pergament, O. Cohen, and A. Fish, "A 250 mV 8 kb 40 nm Ultra-Low Power 9T Supply Feedback SRAM (SF-SRAM)," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 11, pp. 2713–2726, Nov 2011.

[5] D. Ernst, N. S. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner, and T. Mudge, "Razor: a low-power pipeline based on circuit-level timing speculation," in *Proceedings. 36th Annual IEEE/ACM International Symposium on Microarchitecture, 2003. MICRO-36.*, Dec 2003, pp. 7–18.

[6] S. Das, C. Tokunaga, S. Pant, W. H. Ma, S. Kalaiselvan, K. Lai, D. M. Bull, and D. T. Blaauw, "RazorII: In Situ Error Detection and Correction for PVT and SER Tolerance," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 1, pp. 32–48, Jan 2009.

[7] I. Kwon, S. Kim, D. Fick, M. Kim, Y. P. Chen, and D. Sylvester, "Razor-Lite: A Light-Weight Register for Error Detection by Observing Virtual Supply Rails," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 9, pp. 2054–2066, Sept 2014.

[8] Y. Zhang, M. Khayatzadeh, K. Yang, M. Saligane, N. Pinckney, M. Alioto, D. Blaauw, and D. Sylvester, "iRazor: Current-Based Error Detection and Correction Scheme for PVT Variation in 40-nm ARM Cortex-R4 Processor," *IEEE Journal of Solid-State Circuits*, vol. PP, no. 99, pp. 1–13, 2017.

[9] K. A. Bowman, J. W. Tschanz, S. L. L. Lu, P. A. Aseron, M. M. Khellah, A. Raychowdhury, B. M. Geuskens, C. Tokunaga, C. B. Wilkerson, T. Karnik, and V. K. De, "A 45 nm Resilient

Microprocessor Core for Dynamic Variation Tolerance,” *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 194–208, Jan 2011.

[10] R. N. Tadros, W. Hua, M. T. Moreira, N. L. V. Calazans, and P. A. Beerel, “A Low-Power Low-Area Error-Detecting Latch for Resilient Architectures in 28-nm FDSOI,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no. 9, pp. 858–862, Sept 2016.

[11] R. Hegde and N. R. Shanbhag, “A voltage overscaled low-power digital filter IC,” *IEEE Journal of Solid-State Circuits*, vol. 39, no. 2, pp. 388–391, Feb 2004.

[12] B. Shim, S. R. Sridhara, and N. R. Shanbhag, “Reliable low-power digital signal processing via reduced precision redundancy,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 5, pp. 497–510, May 2004.

[13] L. Wang and N. Shanbhag, “Low-power filtering via adaptive error-cancellation,” *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 575–583, Feb 2003.

[14] P. N. Whatmough, S. Das, D. M. Bull, and I. Darwazeh, “Circuit-Level Timing Error Tolerance for Low-Power DSP Filters and Transforms,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 6, pp. 989–999, June 2013.

[15] D. Rossi, A. Pullini, I. Loi, M. Gautschi, F. K. Gurkaynak, A. Teman, J. Constantin, A. Burg, I. Miro-Panades, E. Beigné, F. Clermidy, F. Abouzeid, P. Flatresse, and L. Benini, “193 MOPS/mW @ 162 MOPS, 0.32V to 1.15V voltage range multi-core accelerator for energy efficient parallel and sequential digital processing,” in *2016 IEEE Symposium in Low-Power and High-Speed Chips (COOL CHIPS XIX)*, April 2016, pp. 1–3.

[16] H. Kawaguchi and T. Sakurai, “A reduced clock-swing flip-flop (RCSFF) for 63% power reduction,” *IEEE Journal of Solid-State Circuits*, vol. 33, no. 5, pp. 807–811, May 1998.

[17] V. G. Oklobdzija, V. M. Stojanovic, D. M. Markovic, and N. M. Nedovic, *Digital system clocking: high-performance and low-power aspects*. John Wiley & Sons, 2005.

[18] N. Nedovic and V. G. Oklobdzija, “Dual-edge triggered storage elements and clocking strategy for low-power systems,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 5, pp. 577–590, May 2005.

[19] P. Zhao, T. K. Darwish, and M. A. Bayoumi, “High-performance and low-power conditional discharge flip-flop,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 5, pp. 477–484, May 2004.

[20] S. Paik, J. Kung, and Y. Shin, “Exploring the opportunity of optimizing sequencing elements in ASIC designs,” in *2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Aug 2011, pp. 1–4.

[21] M. Alioto, E. Consoli, and G. Palumbo, "DET FF topologies: A detailed investigation in the energy-delay-area domain," in *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, May 2011, pp. 563–566.

[22] ——, "Analysis and Comparison in the Energy-Delay-Area Domain of Nanometer CMOS Flip-Flops: Part I – Methodology and Design Strategies," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 5, pp. 725–736, May 2011.

[23] ——, "Analysis and Comparison in the Energy-Delay-Area Domain of Nanometer CMOS Flip-Flops: Part II – Results and Figures of Merit," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 5, pp. 737–750, May 2011.

[24] R. P. Llopis and M. Sachdev, "Low power, testable dual edge triggered flip-flops," in *Low Power Electronics and Design, 1996., International Symposium on*, Aug 1996, pp. 341–345.

[25] A. Gago, R. Escano, and J. A. Hidalgo, "Reduced implementation of D-type DET flip-flops," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 3, pp. 400–402, Mar 1993.

[26] J. Tschanz, S. Narendra, Z. Chen, S. Borkar, M. Sachdev, and V. De, "Comparative delay and energy of single edge-triggered and dual edge-triggered pulsed flip-flops for high-performance microprocessors," in *Low Power Electronics and Design, International Symposium on, 2001.*, 2001, pp. 147–152.

[27] N. Nedovic, W. W. Walker, V. G. Oklobdzija, and M. Aleksic, "A low power symmetrically pulsed dual edge-triggered flip-flop," in *Solid-State Circuits Conference, 2002. ESSCIRC 2002. Proceedings of the 28th European*, Sept 2002, pp. 399–402.

[28] A. Bonetti, A. Teman, and A. Burg, "An overlap-contention free true-single-phase clock dual-edge-triggered flip-flop," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2015, pp. 1850–1853.

[29] M. Alioto, E. Consoli, and G. Palumbo, "Clock distribution in clock domains with Dual-Edge-Triggered Flip-Flops to improve energy-efficiency," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, May 2010, pp. 321–324.

[30] N. Nedovic, W. W. Walker, and V. G. Oklobdzija, "A test circuit for measurement of clocked storage element characteristics," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 8, pp. 1294–1304, Aug 2004.

[31] D. Doswald, B. Schreier, S. Oetiker, J. Hafliger, P. Blessing, N. Felber, and W. Fichtner, "A 30 Frames/s Megapixel Real-Time CMOS Image Processor," in *2000 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.00CH37056)*, Feb 2000, pp. 232–233.

[32] S. Lapshev and S. M. R. Hasan, "New Low Glitch and Low Power DET Flip-Flops Using Multiple C-Elements," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 10, pp. 1673–1681, Oct 2016.

[33] R. P. Llopis, “Electronic circuit with dual edge triggered flip-flop,” Oct. 24 2000, uS Patent 6,137,331.

[34] J. W. Tschanz, D. Somasekhar, and V. K. De, “Gating for dual edge-triggered clocking,” Sep. 19 2006, uS Patent 7,109,776.

[35] H. Kaeslin, *Digital integrated circuit design: from VLSI architectures to CMOS fabrication.* Cambridge University Press, 2008.

[36] Y. Kim, W. Jung, I. Lee, Q. Dong, M. Henry, D. Sylvester, and D. Blaauw, “27.8 A static contention-free single-phase-clocked 24T flip-flop in 45nm for low-power applications,” in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2014, pp. 466–467.

[37] F. Stas and D. Bol, “A 0.4-V 0.66-fJ/Cycle Retentive True-Single-Phase-Clock 18T Flip-Flop in 28-nm Fully-Depleted SOI CMOS,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 3, pp. 935–945, March 2018.

[38] M. Afghahi, “A robust single phase clocking for low power, high-speed VLSI applications,” *IEEE Journal of Solid-State Circuits*, vol. 31, no. 2, pp. 247–254, Feb 1996.

[39] J.-S. Wang, “A new true-single-phase-clocked double-edge-triggered flip-flop for low-power VLSI designs,” in *Circuits and Systems, 1997. ISCAS ’97., Proceedings of 1997 IEEE International Symposium on*, vol. 3, Jun 1997, pp. 1896–1899 vol.3.

[40] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits.* Prentice Hall, 2002.

[41] M. Cho, S. T. Kim, C. Tokunaga, C. Augustine, J. P. Kulkarni, K. Ravichandran, J. W. Tschanz, M. M. Khellah, and V. De, “Postsilicon Voltage Guard-Band Reduction in a 22 nm Graphics Execution Core Using Adaptive Voltage Scaling and Dynamic Power Gating,” *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, pp. 50–63, Jan 2017.

[42] W. Jin, S. Kim, W. He, Z. Mao, and M. Seok, “In Situ Error Detection Techniques in Ultralow Voltage Pipelines: Analysis and Optimizations,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 3, pp. 1032–1043, March 2017.

[43] J. Constantin, L. Wang, G. Karakonstantis, A. Chattopadhyay, and A. Burg, “Exploiting dynamic timing margins in microprocessors for frequency-over-scaling with instruction-based clock adjustment,” in *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2015, pp. 381–386.

[44] J. Constantin, A. Bonetti, A. Teman, C. Müller, L. Schmid, and A. Burg, “DynOR: A 32-bit microprocessor in 28 nm FD-SOI with cycle-by-cycle dynamic clock adjustment,” in *ESSCIRC Conference 2016: 42nd European Solid-State Circuits Conference*, Sept 2016, pp. 261–264.

[45] F. Botman, D. Bol, J. Legat, and K. Roy, “Data-Dependent Operation Speed-Up Through Automatically Inserted Signal Transition Detectors for Ultralow Voltage Logic Circuits,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 12, pp. 2561–2570, Dec 2014.

[46] D. Bol, J. D. Vos, C. Hocquet, F. Botman, F. Durvaux, S. Boyd, D. Flandre, and J. Legat, “SleepWalker: A 25-MHz 0.4-V Sub-mm^{2}7-μW/MHzMicrocontroller in 65-nm LP/GP CMOS for Low-Carbon Wireless Sensor Nodes,” *IEEE Journal of Solid-State Circuits*, vol. 48, no. 1, pp. 20–32, Jan 2013.

[47] OpenRISC Community, “OpenRISC Architecture,” 2018, http://openrisc.io/architecture.

[48] J. H.-F. Constantin, “Microarchitectural Low-Power Design Techniques for Embedded Microprocessors,” Ph.D. dissertation, Lausanne, 2016.

[49] S. Kim and M. Seok, “Variation-Tolerant, Ultra-Low-Voltage Microprocessor With a Low-Overhead, Within-a-Cycle In-Situ Timing-Error Detection and Correction Technique,” *IEEE Journal of Solid-State Circuits*, vol. 50, no. 6, pp. 1478–1490, June 2015.

[50] E. Beigné, A. Valentian, I. Miro-Panades, R. Wilson, P. Flatresse, F. Abouzeid, T. Benoist, C. Bernard, S. Bernard, O. Billoint, S. Clerc, B. Giraud, A. Grover, J. L. Coz, J. P. Noel, O. Thomas, and Y. Thonnart, “A 460 MHz at 397 mV, 2.6 GHz at 1.3 V, 32 bits VLIW DSP Embedding $\mathrm{F_{MAX}}$ Tracking,” *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 125–136, Jan 2015.

[51] M. Alioto, *Enabling the Internet of Things*. Springer, 2017.

[52] D. Blaauw, D. Sylvester, P. Dutta, Y. Lee, I. Lee, S. Bang, Y. Kim, G. Kim, P. Pannuto, Y. S. Kuo, D. Yoon, W. Jung, Z. Foo, Y. P. Chen, S. Oh, S. Jeong, and M. Choi, “IoT design space challenges: Circuits and systems,” in *2014 Symposium on VLSI Technology (VLSI-Technology): Digest of Technical Papers*, June 2014, pp. 1–2.

[53] D. Rossi, I. Loi, A. Pullini, and L. Benini, “Ultra-Low-Power Digital Architectures for the Internet of Things,” in *Enabling the Internet of Things*. Springer, 2017, pp. 69–93.

[54] Y. Huan, N. Ma, J. Mao, S. Blixt, Z. Lu, Z. Zou, and L. R. Zheng, “A 101.4 GOPS/W Reconfigurable and Scalable Control-Centric Embedded Processor for Domain-Specific Applications,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 12, pp. 2245–2256, Dec 2016.

[55] D. Rossi, A. Pullini, I. Loi, M. Gautschi, F. K. Gurkaynak, A. Teman, J. Constantin, A. Burg, I. Miro-Panades, E. Beigné, F. Clermidy, F. Abouzeid, P. Flatresse, and L. Benini, “193 MOPS/mW @ 162 MOPS, 0.32V to 1.15V voltage range multi-core accelerator for energy efficient parallel and sequential digital processing,” in *2016 IEEE Symposium in Low-Power and High-Speed Chips (COOL CHIPS XIX)*, April 2016, pp. 1–3.

[56] F. Conti, C. Pilkington, A. Marongiu, and L. Benini, “He-P2012: Architectural heterogeneity exploration on a scalable many-core platform,” in *2014 IEEE 25th International Conference on Application-Specific Systems, Architectures and Processors*, June 2014, pp. 114–120.

[57] F. Sheikh, M. Miller, B. Richards, D. Marković, and B. Nikolić, “A 1-190MSample/s 8-64 tap energy-efficient reconfigurable FIR filter for multi-mode wireless communication,” in *VLSI Circuits (VLSIC), 2010 IEEE Symposium on*, June 2010, pp. 207–208.

[58] J. Chen, C. H. Chang, F. Feng, W. Ding, and J. Ding, “Novel Design Algorithm for Low Complexity Programmable FIR Filters Based on Extended Double Base Number System,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 1, pp. 224–233, Jan 2015.

[59] S. Y. Park and P. K. Meher, “Efficient FPGA and ASIC Realizations of a DA-Based Reconfigurable FIR Digital Filter,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 61, no. 7, pp. 511–515, July 2014.

[60] P. Kulkarni, P. Gupta, and M. Ercegovac, “Trading Accuracy for Power with an Underdesigned Multiplier Architecture,” in *VLSI Design (VLSI Design), 2011 24th International Conference on*, Jan 2011, pp. 346–351.

[61] K. Bhardwaj, P. S. Mane, and J. Henkel, “Power- and area-efficient Approximate Wallace Tree Multiplier for error-resilient systems,” in *Quality Electronic Design (ISQED), 2014 15th International Symposium on*, March 2014, pp. 263–269.

[62] C. H. Lin and I. C. Lin, “High accuracy approximate multiplier with error correction,” in *2013 IEEE 31st International Conference on Computer Design (ICCD)*, Oct 2013, pp. 33–38.

[63] C. Liu, J. Han, and F. Lombardi, “A low-power, high-performance approximate multiplier with configurable partial error recovery,” in *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2014, pp. 1–4.

[64] C. Neau, K. Muhammad, and K. Roy, “Low complexity FIR filters using factorization of perturbed coefficients,” in *Design, Automation and Test in Europe, 2001. Conference and Exhibition 2001. Proceedings*, 2001, pp. 268–272.

[65] S. Hong, S. Kim, M. C. Papaefthymiou, and W. E. Stark, “Low power parallel multiplier design for DSP applications through coefficient optimization,” in *ASIC/SOC Conference, 1999. Proceedings. Twelfth Annual IEEE International*, 1999, pp. 286–290.

[66] M. Mehendale, S. D. Sherlekar, and G. Venkatesh, “Low-power realization of FIR filters on programmable DSPs,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 6, no. 4, pp. 546–553, Dec 1998.

[67] B. Parhami, *Computer Arithmetic*. Oxford University Press, 1999, vol. 20, no. 00.

[68] R. Zimmermann, "VHDL library of arithmetic units," in *Proc. First Int. Forum on Design Languages (FDL'98), Lausanne, Switzerland.* Citeseer, 1998, pp. 267–272.

[69] Y. Huang, A. Kapoor, R. Rutten, and J. Pineda de Gyvez, "A 13 Bits 4.096GHz 45Nm CMOS Digital Decimation Filter Chain with Carry-Save Format Numbers," *Microprocess. Microsyst.*, vol. 39, no. 8, pp. 869–878, Nov. 2015. [Online]. Available: http://dx.doi.org/10.1016/j.micpro.2014.11.003

[70] C. Berry, J. Warnock, J. Isakson, J. Badar, B. Bell, F. Malgioglio, G. Mayer, D. Hamid, J. Surprise, D. Wolpert, O. Geva, B. Huott, L. Sigal, S. Carey, R. Rizzolo, R. Nigaglioni, M. Cichanowski, D. Chidambarrao, C. Jacobi, A. Saporito, A. O'neill, R. Sonnelitter, C. Zoellin, M. Wood, and J. Neves, "IBM z14: 14nm Microprocessor for the Next-Generation Mainframe," in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb 2018, pp. 36–38.

[71] M. Gautschi, P. D. Schiavone, A. Traber, I. Loi, A. Pullini, D. Rossi, E. Flamand, F. K. Gürkaynak, and L. Benini, "Near-Threshold RISC-V Core With DSP Extensions for Scalable IoT Endpoint Devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2700–2713, Oct 2017.

[72] Y. H. Chen, T. Krishna, J. S. Emer, and V. Sze, "Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, pp. 127–138, Jan 2017.

[73] M. Tikekar, V. Sze, and A. Chandrakasan, "A fully-integrated energy-efficient H.265/HEVC decoder with eDRAM for wearable devices," in *2017 Symposium on VLSI Circuits*, June 2017, pp. C230–C231.

[74] P. Meinerzhagen, A. Teman, R. Giterman, N. Edri, A. Burg, and A. Fish, *Gain-Cell Embedded DRAMs for Low-Power VLSI Systems-on-Chip.* Springer, 2017.

[75] M. Ichihashi, H. Toda, Y. Itoh, and K. Ishibashi, "0.5 V asymmetric three-Tr. cell (ATC) DRAM using 90nm generic CMOS logic process," in *Digest of Technical Papers. 2005 Symposium on VLSI Circuits, 2005.*, June 2005, pp. 366–369.

[76] D. Somasekhar, Y. Ye, P. Aseron, S. L. Lu, M. M. Khellah, J. Howard, G. Ruhl, T. Karnik, S. Borkar, V. K. De, and A. Keshavarzi, "2 GHz 2 Mb 2T Gain Cell Memory Macro With 128 GBytes/sec Bandwidth in a 65 nm Logic Process Technology," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 1, pp. 174–185, Jan 2009.

[77] Y. Lee, M. T. Chen, J. Park, D. Sylvester, and D. Blaauw, "A 5.42nW/kB retention power logic-compatible embedded DRAM with 2T dual-Vt gain cell for low power sensing applications," in *2010 IEEE Asian Solid-State Circuits Conference*, Nov 2010, pp. 1–4.

[78] K. C. Chun, P. Jain, J. H. Lee, and C. H. Kim, "A 3T Gain Cell Embedded DRAM Utilizing Preferential Boosting for High Density and Low Power On-Die Caches," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 6, pp. 1495–1505, June 2011.

[79] R. Giterman, A. Fish, A. Burg, and A. Teman, “A 4-Transistor nMOS-Only Logic-Compatible Gain-Cell Embedded DRAM With Over 1.6-ms Retention Time at 700 mV in 28-nm FD-SOI,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. PP, no. 99, pp. 1–12, 2017.

[80] R. Giterman, A. Fish, N. Geuli, E. Mentovich, A. Burg, and A. Teman, “An 800-MHz Mixed-VT 4T IFGC Embedded DRAM in 28-nm CMOS Bulk Process for Approximate Storage Applications,” *IEEE Journal of Solid-State Circuits*, pp. 1–13, 2018.

[81] O. Naji, C. Weis, M. Jung, N. Wehn, and A. Hansson, “A high-level DRAM timing, power and area exploration tool,” in *2015 International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS)*, July 2015, pp. 149–156.

[82] S. Thoziyoor, J. H. Ahn, M. Monchiero, J. B. Brockman, and N. P. Jouppi, “A Comprehensive Memory Modeling Tool and Its Application to the Design and Analysis of Future Memory Hierarchies,” in *2008 International Symposium on Computer Architecture*, June 2008, pp. 51–62.

[83] T. Vogelsang, “Understanding the Energy Consumption of Dynamic Random Access Memories,” in *Proceedings of the 2010 43rd Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO ’43. Washington, DC, USA: IEEE Computer Society, 2010, pp. 363–374. [Online]. Available: http://dx.doi.org/10.1109/MICRO.2010.42

[84] H. C. Shih, P. W. Luo, J. C. Yeh, S. Y. Lin, D. M. Kwai, S. L. Lu, A. Schaefer, and C. W. Wu, “DArT: A Component-Based DRAM Area, Power, and Timing Modeling Tool,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 9, pp. 1356–1369, Sept 2014.

[85] A. Bonetti, “GEMTOO: A Gain-Cell Embedded DRAM Modeling Tool,” 2018. [Online]. Available: https://tclweb.epfl.ch/gemtoo/index.html

[86] S. J. E. Wilton and N. P. Jouppi, “CACTI: an enhanced cache access and cycle time model,” *IEEE Journal of Solid-State Circuits*, vol. 31, no. 5, pp. 677–688, May 1996.

[87] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, “The Gem5 Simulator,” *SIGARCH Comput. Archit. News*, vol. 39, no. 2, pp. 1–7, Aug. 2011. [Online]. Available: http://doi.acm.org/10.1145/2024716.2024718

[88] P. Meinerzhagen, A. Teman, R. Giterman, A. Burg, and A. Fish, “Exploration of Sub-VT and Near-VT 2T Gain-Cell Memories for Ultra-Low Power Applications under Technology Scaling,” *Journal of Low Power Electronics and Applications*, vol. 3, no. 2, pp. 54–72, 2013. [Online]. Available: http://www.mdpi.com/2079-9268/3/2/54

[89] M. U. Khalid, P. Meinerzhagen, and A. Burg, “Replica bit-line technique for embedded multilevel gain-cell DRAM,” in *10th IEEE International NEWCAS Conference*, June 2012, pp. 77–80.

[90] R. Giterman, A. Teman, P. Meinerzhagen, L. Atias, A. Burg, and A. Fish, “Single-Supply 3T Gain-Cell for Low-Voltage Low-Power Applications,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 1, pp. 358–362, Jan 2016.

[91] R. Giterman, A. Fish, N. Geuli, E. Mentovich, A. Burg, and A. Teman, “An 800 Mhz mixed-VT 4T gain-cell embedded DRAM in 28 nm CMOS bulk process for approximate computing applications,” in *ESSCIRC 2017 - 43rd IEEE European Solid State Circuits Conference*, Sept 2017, pp. 308–311.

[92] P. Meinerzhagen, C. Roth, and A. Burg, “Towards generic low-power area-efficient standard cell based memory architectures,” in *2010 53rd IEEE International Midwest Symposium on Circuits and Systems*, Aug 2010, pp. 129–132.

[93] P. Meinerzhagen, S. M. Y. Sherazi, A. Burg, and J. N. Rodrigues, “Benchmarking of Standard-Cell Based Memories in the Sub-V_T Domain in 65-nm CMOS Technology,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 2, pp. 173–182, June 2011.

[94] P. Meinerzhagen, O. Andersson, B. Mohammadi, Y. Sherazi, A. Burg, and J. N. Rodrigues, “A 500 fW/bit 14 fJ/bit-access 4kb standard-cell based sub-VT memory in 65nm CMOS,” in *2012 Proceedings of the ESSCIRC (ESSCIRC)*, Sept 2012, pp. 321–324.

[95] P. Meinerzhagen, A. Bonetti, G. Karakonstantis, C. Roth, F. Giirkaynak, and A. Burg, “Refresh-free dynamic standard-cell based memories: Application to a QC-LDPC decoder,” in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2015, pp. 1426–1429.

[96] A. Teman, D. Rossi, P. Meinerzhagen, L. Benini, and A. Burg, “Controlled placement of standard cell memory arrays for high density and low power in 28nm FD-SOI,” in *The 20th Asia and South Pacific Design Automation Conference*, Jan 2015, pp. 81–86.

[97] ——, “Power, Area, and Performance Optimization of Standard Cell Memory Arrays Through Controlled Placement,” *ACM Trans. Des. Autom. Electron. Syst.*, vol. 21, no. 4, pp. 59:1–59:25, May 2016. [Online]. Available: http://doi.acm.org/10.1145/2890498

[98] J. Tschanz, N. S. Kim, S. Dighe, J. Howard, G. Ruhl, S. Vangal, S. Narendra, Y. Hoskote, H. Wilson, C. Lam, M. Shuman, C. Tokunaga, D. Somasekhar, S. Tang, D. Finan, T. Karnik, N. Borkar, N. Kurd, and V. De, “Adaptive Frequency and Biasing Techniques for Tolerance to Dynamic Temperature-Voltage Variations and Aging,” in *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, Feb 2007, pp. 292–604.

[99] S. Clerc, M. Saligane, F. Abouzeid, M. Cochet, J. M. Daveau, C. Bottoni, D. Bol, J. De-Vos, D. Zamora, B. Coeffic, D. Soussan, D. Croain, M. Naceur, P. Schamberger, P. Roche, and D. Sylvester, “A 0.33V/-40°C process/temperature closed-loop compensation SoC embedding all-digital clock multiplier and DC-DC converter exploiting FDSOI 28nm back-gate biasing,” in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, Feb 2015, pp. 1–3.

[100] I. Miro-Panades, E. Beigné, Y. Thonnart, L. Alacoque, P. Vivet, S. Lesecq, D. Puschini, A. Molnos, F. Thabet, B. Tain, K. B. Chehida, S. Engels, R. Wilson, and D. Fuin, “A Fine-Grain Variation-Aware Dynamic Vdd-Hopping AVFS Architecture on a 32 nm GALS MPSoC,” vol. 49, no. 7, pp. 1475–1486, Jul 2014.

[101] M. Verhelst and B. Moons, “Embedded Deep Neural Network Processing: Algorithmic and Processor Techniques Bring Deep Learning to IoT and Edge Devices,” *IEEE Solid-State Circuits Magazine*, vol. 9, no. 4, pp. 55–65, Fall 2017.

[102] N. Shanbhag, Y. Kim, A. Singer, B. Murmann, N. Verma, J. Rabaey, and D. Blaauw, “Systems on Nanoscale Information fabriCs,” in *GOMACTech Conference*, March 2018.

Acronyms

1T-1C	one-transistor one-capacitor
6T	six-transistor
ADC	analog-to-digital converter
APR	automated place-and-route
BB	body biasing
BC	best case
BDM	backward-detection mode
BIST	memory built-in self-test
BR4	radix-4 Booth-recoded
BW2	radix-2 Baugh-Wooley
CNN	convolutional neural network
CPU	central processing unit
CTS	clock-tree synthesis
DCA	dynamic clock adjustment
DCG	dynamically-adjustable clock generator
DCRO	digitally-controlled ring oscillator
DDM	distributed-RC delay model
DET	dual-edge-triggered
DET-C2LM	C^2MOS latch-MUX
DET-CDFF	conditional discharge flip-flop
DET-CFF	C-element flip-flop
DET-CG	DET clock gate
DET-FF	DET flip-flop
DET-ISLM	isolated-storage latch-MUX
DET-SPGFF	symmetric pulse-generator flip-flop

DET-TGLM	transmission-gate latch-MUX
DFF	D flip-flop
DRAM	dynamic random-access memory
DSP	digital signal processing
DSPR	DET/SET power ratio
DVFS	dynamic voltage and frequency scaling
EDA	electronic design automation
EDAC	error detection and correction
EDS	error-detection sequential
EMI	electromagnetic interference
FA	full adder
FBB	forward body biasing
FDM	forward-detection mode
FIR	finite impulse response
FLL	frequency-locked loop
FPM	fast-path measurement
GC	gain cell
GC-eDRAM	gain-cell embedded DRAM
HA	half adder
I/O	input-output
IC	integrated circuit
IoT	Internet of Things
LFSR	linear-feedback shift register
LSB	least-significant bit
LUT	lookup table
MAC	multiply-and-accumulate
MC	Monte Carlo
MSB	most-significant bit
MUX	multiplexer
NCO	negative clock-overlap

PCO	positive clock-overlap
PDL	programmable delay line
PDP	power-delay product
PoFF	point of first failure
PPG	partial-products generator
PPR	partial-products reducer
PVT	process, voltage and temperature
QoS	quality of service
RAM	random-access memory
RBB	reverse body biasing
RCA	ripple-carry adder
RTL	register-transfer level
SC	standard cell
SCM	standard-cell memory
SDC	Synopsys design constraints
SDET-TSPCFF	static dual-edge-triggered flip-flop with true-single-phase clock
SET	single-edge-triggered
SET-CG	SET clock gate
SET-FF	SET flip-flop
SoC	system-on-chip
SRAM	static random-access memory
STA	static timing analysis
SVM	setup-violation monitoring
TC	typical case
TG	transmission gate
TMS	timing-monitoring sequential
TRC	tunable replica circuit
TSM	timing-slack monitoring
TSPC	true-single-phase-clock
ULP	ultra-low power

VLSI	very-large-scale integration
VMA	vector-merging adder
VOS	voltage over-scaling
WC	worst case

List of Figures

1.1 Modules embedded in a node for the Internet of Things (IoT). The modules highlighted in green are targeted for energy efficiency by the circuits, techniques, and tools proposed in this thesis. 2

1.2 Contributions of this thesis classified by targeted operating region and addressed circuit topology. 4

2.1 Schematic of the DET-TGLM flip-flop used in this work. 17

2.2 Schematic of the DET clock gate (DET-CG) [33] used in this work, characteristic waveforms and clock-tree synthesis (CTS) directives. 18

2.3 Flow chart for the proposed DET implementation within the digital standard design flow. 19

2.4 Example of Synopsys design constraints (SDC) commands for defining a virtual reference clock and interface constraints. 21

2.5 Waveforms showing edge combinations due to DET clock gating and example design constraints to ensure proper STA. 24

2.6 Breakdown of the total power consumption for the SET implementation of each case study. 25

2.7 Equivalent SET maximum clock frequency achieved at different technology-process corners under a specific set of design constraints for each of the three case studies. 26

2.8 Power breakdown for each of the case studies, considering the contribution from the clock tree, registers, combinatorial logic, and the total power consumption. Each bar plot is normalized to the power consumption under SET clocking and it is further divided into internal, switching, and leakage power. 27

2.9 Dependency of the DET/SET total power ratio (DSPR) on the equivalent SET frequency that is defined in Section 2.1.4.1. The operating points of the presented simulations are marked with a cross, while the points where SET and DET power consumptions are estimated to be equal have been marked with a circle. 29

2.10 Schematic of the DET-TGLM. 33

2.11 Voltage waveforms of the signals affected by the clock overlap in the DET-TGLM. The failure is represented with continuous lines while the segmented lines show the case when the circuit overcomes the hazard. 34

2.12 Distribution of the minimum voltage value reached by the storage node SNN during the clock overlap for 10,000 Monte Carlo (MC) runs. 35

2.13 Schematic of the proposed SDET-TSPCFF. 36

2.14 Top branch of the SDET-TSPCFF. 37

2.15 Monte Carlo family plots of the SDET-TSPCFF for 10,000 runs. 38

3.1 The effect of path shaping and the application of the dynamic clock adjustment (DCA) technique [44]. 44

3.2 Architecture of a microprocessor supporting dynamic clock adjustment (DCA) [44]. 45

3.3 Schematic of the dynamically-adjustable clock generator (DCG), layout of the programmable delay line (PDL) and internal timing. 46

3.4 Die micrograph of the test chip in 28 nm FD-SOI process technology implementing the dynamically-adjustable clock generator (DCG) and the microprocessor supporting the dynamic clock adjustment (DCA) technique [44]. 47

3.5 Measurements of the output clock periods produced by the DCG for every delay setting of the DCA PDL while the static PDL is set to its shortest delay and for a voltage supply of 1.2 V. 48

3.6 Circuit schematic of the timing-monitoring sequential (TMS). 52

3.7 Examples of error detection in both forward-detection mode (FDM) and backward-detection mode (BDM). 55

3.8 Timing-monitoring modes enabled by the TMS. 56

4.1 Structure of a signed $n \times n$-bit radix-2 Baugh-Wooley multiplier. 63

4.2 Structure of a generic partial-products reducer implemented as a carry-save adder with (m,2) compressors [68]. 63

4.3 Structure of a signed $n \times n$-bit radix-4 Booth-recoded multiplier. 64

4.4 reliability of the 32 output bits of a 16×16-bit radix-4 Booth-recoded (BR4) multiplier when overclocking is applied. 66

4.5 Reliability rate of the 32 output bits of a 16×16-bit BR4 multiplier when overclocking is applied. 67

4.6 Number of the generated non-zero partial-product bits in a signed 4×4-bit radix-2 Baugh-Wooley multiplier for all input combinations. 70

4.7 Average number of the generated non-zero partial-product bits per input operand in the 8×8-bit radix-2 Baugh-Wooley multiplier. 71

4.8 Average number of the non-zero partial-product bits in an 8×8-bit radix-4 Booth-recoded multiplier. 72

4.9 Normalized dynamic power consumption per input operand x in the 8×8-bit radix-2 Baugh-Wooley multiplier. 73

4.10 Normalized dynamic power consumption in an 8×8-bit radix-4 Booth-recoded multiplier. 74

4.11 Normalized dynamic power consumption per input operand for the 8×8-bit multiplier provided by the synthesis software. 76

4.12 Block diagram of a direct-form FIR filter with 16 taps. 78

4.13 Dynamic power consumption of the 8×8-bit multipliers implemented in a reconfigurable 16-taps FIR filter designed with different cutoff-frequency constraints and windowing methods. 79

4.14 Flowchart of the conventional flow and proposed technique for the design of the filter. 80

4.15 Magnitude response of the initial and power-optimized FIR low-pass filters implementing various radix-4 Booth-recoded multipliers. 82

4.16 Magnitude response of various filters implemented with a 16-taps FIR accelerator and 8×8-bit radix-4 Booth-recoded multipliers. For each filter type, both initial and power-optimized version is provided. 83

4.17 Die micrograph of the FIR accelerators. 84

4.18 Magnitude response of the initial and power-optimized filters implemented with the 16-taps FIR accelerators of the test chip. 85

5.1 Variety of gain cells (GCs) reported in the literature: (a) 2T PMOS-only GC [76], (b) 2T PMOS-only GC with high-V_T access transistor [77], (c) 2T GC with PMOS high-V_T access transistor and NMOS read transistor [88], (d) 3T PMOS-only GC [78], (e) 3T GC with PMOS access transistor and NMOS read transistors [89], (f) 4T NMOS-only GC with internal feedback [79]. 92

5.2 Different memory organizations for a 32 kb GC-eDRAM. 94

5.3 Examples of architectural transformations cutting the BLs, showing their impact on the memory organization of a 16 kb memory. 96

5.4 Architecture of the processor with GC-eDRAM as data memory. 99

5.5 Layout floorplan of the processor with GC-eDRAM as data memory. 100

5.6 Structure of the GEMTOO modeling tool. 102

5.7 Delay modeling of the read path in a GC-eDRAM. 103

5.8 Comparison of the simulated and estimated delays for the timing validation of the GC-eDRAM modeling tool. 106

5.9 Comparison between the measured $f_{\max}$ of [80] and the $f_{\max}$ estimated with the modeling tool for different multiples of the standard deviation (n_σ). 108

5.10 Breakdown of read delay for memories of different sizes. 110

5.11 Maximum operating frequency for different GC topologies and refresh rates. . . 111

5.12 Timing breakdown for each path of a 8 kb GC-eDRAM. The considered GC-eDRAM is either implemented as monolithic or hierarchical memory. The blocks contributing to the delay of each of the paths are: flip-flop (FF), global address decoder (Dec), level shifter (LS), local WL-enable logic and WL buffer (LWL), WBL driver (DWB), GC read port (GC), sense amplifier (SA) and RBL mux (Mux). 112

5.13 Impact of folding, refresh rate and operating frequency on memory availability and bandwidth. 114

5.14 Area efficiency for single-array memories of different memory sizes. 115

List of Tables

2.1 Power Breakdown Details of Case Study B 28

2.2 Area Overhead of the Standard Cells (SCs) 30

2.3 Summary of the compared DET-FFs[1] 39

3.1 Truth table of the internal error signals in the TMS 53

3.2 Comparison between a baseline flip-flop and the TMS 57

4.1 Propagation delays of the 8×8-bit multipliers 77

4.2 Specifications and power savings of the low-pass FIR filters 81

4.3 Power measurements of the FIR accelerators in the test chip 86

5.1 Comparison between the simulated and the estimated f_{max}. 107

5.2 Memory organizations that maximize f_{max}. 113

5.3 Memory availability (α) and bandwidth (b) for different GCs and refresh rates. . 115

5.4 Memory organizations that maximize availability (α) and bandwidth (b). 116

5.5 Impact of the GC topology on the area of monolithic memories. 116

5.6 Most area-efficient memory organizations for hierarchical memories. 117

Curriculum Vitae

Name: **Andrea BONETTI**
Date of Birth: 11.05.1987
Nationality: Italian
Address: EPFL, STI-IEL-TCL
Station 11
CH-1015 Lausanne
Switzerland
E-Mail: andrea.bonetti@epfl.ch
andreabonetti@gmail.com

Education

06.2014 – 01.2019	**École Polytechnique Fédérale de Lausanne**, Lausanne (VD), CH Ph.D Degree in Electrical Engineering
10.2009 – 04.2012	**Politecnico di Milano**, Milano, IT Master's Degree in Electronic Engineering
09.2006 – 09.2009	**Politecnico di Milano**, Milano, IT Bachelor's Degree in Biomedical Engineering

Professional Experience

06.2014 – 01.2019	**École Polytechnique Fédérale de Lausanne**, Lausanne (VD), CH Doctoral Assistant at Telecommunications Circuits Laboratory
05.2016 – 07.2016	**Intel Corporation**, Portland (OR), US Design Intern at Circuit Research Laboratory
11.2012 – 05.2014	**AMS AG**, Rapperswil (SG), CH Design Engineer

List of Publications

Journal Papers

A. Bonetti, R. Golman, R. Giterman, A. Teman, and A. Burg, "Gain-Cell Embedded DRAMs: Modeling and Design Space", *Under revision*, 2018.

R. Giterman, A. Bonetti, A. Teman, and A. Burg, "GC-eDRAM with Body-Bias Compensated Readout and Error Detection in 28 nm FD-SOI", *Under revision*, 2018. *

P. Giard, A. Balatsoukas-Stimming, C. Müller, A. Bonetti, C. Thibeault, W. J. Gross, P. Flatresse, and A. Burg, "PolarBear: A 28-nm FD-SOI ASIC for Decoding of Polar Codes", *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, December 2017. *

A. Bonetti, A. Teman, P. Flatresse and A. Burg, "Multipliers-Driven Perturbation of Coefficients for Low-Power Operation in Reconfigurable FIR Filters". *IEEE Transactions on Circuits and Systems I: Regular Papers*, September 2017.

A. Bonetti, N. Preyss, A. Teman and A. Burg, "Automated Integration of Dual-Edge Clocking for Low- Power Operation in Nanometer Nodes", *ACM Transactions on Design Automation of Electronic Systems*, May 2017.

S. Brenna, A. Bonetti, A. Bonfanti, and A. Lacaita, "An Efficient Tool for the Assisted Design of SAR ADCs Capacitive DACs", *Integration, the VLSI Journal (Elsevier)*, March 2016. *

Conference Papers

M. Widmer, A. Bonetti, and A. Burg, "FPGA-Based Emulation of Embedded DRAMs for Statistical Error Resilience Evaluation of Approximate Computing Systems", *Under revision*, 2018. *

E. Vicario Bravo, A. Bonetti, and A. Burg, "Data-Retention-Time Measurement of Gain-Cell eDRAMs across the Design and Variations Space", *Under revision*, 2018. *

A. Bonetti, J. Constantin, A. Teman and A. Burg, "A Timing-Monitoring Sequential for Forward and Backward Error-Detection in 28 nm FD-SOI", *IEEE International Symposium on Circuits & Systems (ISCAS)*, May 2018.

J. Constantin, A. Bonetti, A. Teman, Christoph Müller, Lorenz Schmid and A. Burg, “DynOR: A 32-bit Microprocessor in 28nm FD-SOI with Cycle-By-Cycle Dynamic Clock Adjustment”, *European Solid-State Circuits Conference (ESSCIRC)*, September 2016.

S. Brenna, L. Bettini, A. Bonetti, A. Bonfanti, and A. Lacaita, “Fundamental Power Limits of SAR and $\Delta\Sigma$ Analog-to-Digital Converters”, *IEEE Nordic Circuits and Systems Conference (NORCAS)*, October 2015. *

A. Bonetti, A. Teman and A. Burg, “An Overlap-Contention Free True-Single-Phase Clock Dual-Edge- Triggered Flip-Flop”, *IEEE International Symposium on Circuits & Systems (ISCAS)*, May 2015.

P. Meinerzhagen, A. Bonetti, G. Karakonstantis, C. Roth, F. K. Gürkaynak, and A. Burg, “Refresh-Free Dynamic Standard-Cell Based Memories: Application to a QC-LDPC Decoder”, *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2015. *

S. Brenna, A. Bonetti, A. Bonfanti, and A. Lacaita, “A Tool for the Assisted Design of Charge Redistribution SAR ADCs”, *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, March 2015. *

S. Brenna, A. Bonetti, A. Bonfanti, and A. Lacaita, “A Simulation and Modeling Environment for the Analysis and Design of Charge Redistribution DACs used in SAR ADCs”, *IEEE International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, July 2014. *

Patents

A. Bonetti, J. P. Kulkarni, C. Tokunaga, M. Cho, P. A. Meinerzhagen, and M. M. Khellah (Applicant: Intel Corporation), “Voltage Level Shifter Monitor with Tunable Voltage Level Shifter Replica Circuit”, *U.S. Patent Application 20180191347*, filed on December 29, 2016. *

* The contents of these publications do not form a part of this thesis.

Ce document a été imprimé au Centre d’impression EPFL,
imprimerie climatiquement neutre, certifiée myClimate.