Energy Efficiency Enhancement of Sub-threshold Digital CMOS

Modeling, Technology Selection, and Architectural Exploration

by

Omer Can Akgun

Thesis for the degree of Doctor of Philosophy

Acknowledgments

At last I have a chance for presenting people who had a positive effect on the work presented in this thesis. This page has been written at the last minute, as usual.

First, I would like to thank my thesis advisor Prof. Yusuf Leblebici for allowing me to carry my thesis work at his laboratory and giving me the chance to work at EPFL. I would like to thank my thesis co-advisor Prof. Joachim Rodrigues for all his support during the last year of my thesis preparation. It has been a great pleasure working with him and I hope to continue it for many years to come.

I would like to thank the members of my thesis jury, Prof. Jens Sparsø, Prof. Tor Sverre Lande, Prof. Christian Piguet and Prof. Adrian Ionescu for investing their time to proofread my thesis and evaluate the research.

I would like to thank current and previous members of LSM for the nice time we spent, all the fruitful discussions and their support. Amongst them, my office mate Milos Stanisavljevic, Alain Vachoux, Alexandre Schmid, Torsten Mähne, Stephane Badel, Ilhan Hatirnaz, Armin Tajalli, Thomas Liechti, Yuksel Temiz, Frank Gurkaynak, Christophe Guillaume-Gentil.

I would like to thank the big Turkish community at Lausanne. Amongst them, our oldest and wisest Ozge, Engin, Zafer, Mustafa, Tolga, Ozden, Derin, Yuksel, Aydin, Mithat, Ayfer, Bilge, Nilay, Eren, Isik, Altug. I am grateful to all of them for all their great friendship and making Lausanne a fun place to be.

I would like to thank many friends back in Turkey. I love and miss you all, I will be there very soon.

I would like to extend my sincere thanks to Marie Halm, who provided immense help with the administrative processes as well as much needed moral support during my studies.

My family - core and extended - have been always there for me and supported all my decisions whether right or wrong. I am deeply grateful to you all.

Last but not least, I would like to thank Gitze for all the love and support she has given me, for holding my hand when I needed it the most and making this thesis a reality.

Abstract

Power density and energy dissipation of digital IC's has become one of the main concerns during the recent years. With the increased usage of battery powered devices, ubiquitous computing, and increase in implantable biomedical applications, enhancing energy efficiency of digital systems is one of the key research areas in digital IC design. For applications with a low demand on throughput, sub-threshold digital operation is one of the promising techniques for ultra-low energy operation. Moreover, global energy minimum operating point of a digital static CMOS circuit, if exists, is in the sub-threshold regime; thus, realizing global minimum energy operation.

This doctoral dissertation presents different energy efficiency enhancement methods for sub-threshold digital CMOS circuits. First, a high level sub-threshold energy model is developed for rapid characterization of digital circuits. Model accuracy is validated with measurements of a circuit that was fabricated in 0.18 μm process, and with simulations for smaller feature sized technologies. Second, this model is applied to compare the energy efficiency of synchronous and asynchronous circuits. It is shown that with a suitable external completion detection mechanism, energy efficiency of asynchronous circuits in the sub-threshold regime is better than the synchronous counterparts. Process selection to minimize energy dissipation is investigated. Moreover, it is shown that with the correct choice of process options, migrating to a smaller feature sized technology increases the energy efficiency of a circuit. Architectural modifications such as parallelism, pipelining, and folding are also explored and applied to reduce energy dissipation.

Finally, a current sensing completion detection system is proposed and implemented for sub-threshold asynchronous circuits. Design flows for de-synchronization of synchronous circuits are presented for various cases with detailed explanations of sub-blocks of the completion detection system. As a proof of concept example, a self-timed cardiac event detector is implemented in a 65 nm CMOS process.

Keywords sub-threshold circuits, static CMOS, asynchronous circuits, current sensing completion detection, energy efficiency, energy model, hardware optimization.

Zusammenfassung

Die Leistungsdichte und der Energieverbrauch von digitalen integrierten Schaltungen (ICs) ist in letzten Jahren zu einem Designkriterium geworden. Durch einen kontinuierlich steigenden Bedarf an batteriebetriebenen Applikationen, und mit der Zunahme von batteriebetriebenen biomedizinischen Implantaten, ist eine Verbesserung der Energieeffizienz in ICs ein populäres Forschungsgebiet geworden.

Der digitale Unterschwellspannungsbetrieb (Subthresholdbetrieb) ist eine adäquate Technik um den Energieverbrauch von Schaltungen mit niedrigem Datendurchsatz zu reduzieren. Des weiteren und falls existierend, liegt der globale energieminimale Arbeitspunkt einer statischen CMOS Schaltung in Unterschwellspannungsbereich; somit wird der energieminimale Betrieb verwirklicht.

Diese Doktorarbeit präsentiert verschiedene Verbesserungsmethoden der Energieeffizienz für digitale Subthreshold CMOS Schaltungen. Ein high-level Subthreshold Energiemodell wurde für die schnelle Charakterisierung von digitalen Schaltungen entwickelt. Die Genauigkeit des Modells wird durch Messungen an einem Referenz IC (180 nm CMOS) validiert. Des weiteren wird dieses Modell verwendet, um die Energieeffizienz von synchronen und asynchronen Schaltungen zu vergleichen.

Es wird demonstriert, dass mit einer externen analogen Schaltung die Vollendung von Operationen annotiert, die Energieeffizienz der asynchronen Schaltungen im Subthresholdbetrieb höher ist als die der synchronen Schaltungen. Optimale Prozessauswahl für den minimalen Energieverbrauch wird untersucht, und es wird gezeigt, dass neuere Technologien mit geeigneten Prozessoptionen die Energieeffizienz der Schaltung erhöht. Die Architekturänderungen wie Parallelismus, Pipelining und Folding werden analysiert um Energieverbrauch zu reduzieren.

Schliesslich wird eine stromüberwachende Schaltung entwickelt und für asynchrone Subtreshold Schaltungen implementiert. Der Entwurfsfluss zur Desynchronisierung der synchronen Schaltungen wird mit ausführlichen Erklärungen über die Unterblöcke des Ausführungsentdeckungsystems präsentiert. Als ein Beispiel für den Konzeptnachweis wird ein selbst getakteter Detektor zur Erfassung von kardialer Aktivität in 65 nm CMOS Prozess implementiert.

Stichwörter Subtreshold Schaltungen, statische CMOS, asynchrone Schaltungen, stromerkennendes Ausführungsentdeckungsystem, die Energieeffizienz, das Energiemodell, die Hardwareoptimierung.

Contents

List of Figures

List of Tables

List of Listings

List of Acronyms

AC	Alternating Current.
AFSM	Asynchronous Finite State Machine.
ASIC	Application Specific Integrated Circuit.
CMOS	Complementary Metal Oxide Semiconductor.
CPF	Common Power Format.
CPU	Central Processing Unit.
CSCD	Current Sensing Completion Detection.
DSP	Digital Signal Processing.
DTMOS	Dynamic Threshold MOS.
ECG	Electrocardiogram.
EDA	Electronic Design Automation.
EDP	Energy Delay Product.
EGM	Electrogram.
EKV	Enz-Krummenacher-Vittoz.
EMV	Energy Minimum Voltage.
FFT	Fast Fourier Transform.
FSM	Finite State Machine.
FT	Fourier Transform.
GLRT	Generalized Likelihood Ratio Test.
HDL	Hardware Description Language.
HVT	High Threshold Voltage.
IC	Integrated Circuit.
LL	Low Leakage.
LVT	Low Threshold Voltage.
MIM	Metal-Insulator-Metal.
MOS	Metal Oxide Semiconductor.
MOSCAP	MOS Capacitor.
MOSFET	Metal Oxide Semiconductor Field Effect Transistor.
MTCMOS	Multiple Threshold CMOS.
PDF	Probability Density Function.
PVT	Process Voltage Temperature.
RDF	Random Dopant Fluctuation.
RTL	Register Transfer Level.

SNM	Static Noise Margin.
SNR	Signal-to-Noise Ratio.
SRAM	Static Random Access Memory.
STG	Signal Transition Graph.
SVT	Standard Threshold Voltage.
VT	Threshold Voltage.
VTC	Voltage Transfer Characteristic.
WT	Wavelet Transform.

Chapter 1

Introduction

Power density and energy dissipation of digital IC's has become a significant design constraint during the recent years. Power density is a more important concern for high performance microprocessor design because of the large number of transistors on a single die and the increasing clock frequencies. The power density limit of a processor is set by the thermal design of the system, and the reliability of operation under high temperature conditions. On the other hand, energy dissipation is a more important concern for mobile systems where the long battery life is desirable.

As portable battery-powered devices such as cell phones, PDAs and portable computers become more complex and ubiquitous; the demand for increased battery life and high performance pushes the designers to develop new circuit techniques to maintain high performance and long operational times. Significant reduction in energy dissipation is possible by lowering the operating voltage of the circuits. Furthermore, by relaxing the constraints of classical strong-inversion operation of MOSFETs, and by accepting that transistors may be operated well below their threshold, ultra-low energy operation in the sub-threshold regime, e.g., with power supply voltages of 100-400 mV, is possible.

In biomedical applications, especially in *in vivo* implanted electronic systems, battery size, hence available energy is extremely limited. For this reason minimizing energy dissipation is one of the main topics of research for biomedical implantable devices. Another application area of ultra-low energy operation is self-sustained systems where reducing the energy dissipation to a minimum is crucial. Furthermore, in the recently emerging field of micro-sensor networks, energy dissipation is one of the main constraints. The micro-sensor nodes are either supplied with a very small battery or have integrated energy scavenging circuitry. For example, a 1 cm^3 lithium

battery is able to continuously supply $10\,\mu W$ of power for five years [8]. This low power supply capability requires the optimization of the circuits for lowest energy dissipation possible.

This thesis deals with the exploration of energy efficiency enhancement techniques of sub-threshold (sub-V_T) digital CMOS circuits. Energy reduction techniques applicable in different levels of the design process of digital circuits are investigated. Based on the findings of our explorations, an asynchronous cardiac event detector, as an example of an ultra-low energy biomedical application, is implemented.

1.1 Thesis Contributions

There are multiple contributions in this thesis for reducing energy dissipation of sub-V_T digital CMOS circuits. These are:

- **A High Level Sub-V_T Energy Model**
  A high level energy model is developed for quick characterization and optimization of sub-V_T static CMOS digital circuits. The model is developed based on asynchronous specifications, and later extended for synchronous operation. The proposed model reduces the simulation time significantly when compared to transistor level simulations, thus, allowing designers to characterize and optimize their designs in the RTL level without requiring time and resource intensive low level simulations.

- **Optimum Process Selection for Energy Minimization**
  Different feature sized processes, i.e., 180, 130, 90 and 65 nm processes are explored for their energy efficiency. Different process options are investigated whenever available for low energy operation. It is shown that unlike the previously published work, migrating to smaller feature sized technologies is beneficial for enhancing energy efficiency.

- **Energy Reduction By Architectural Improvement**
  Energy efficiency enhancement based on architectural techniques and modifications is explored. Benefits and limits of simple architectural modifications such as pipelining and parallelism are shown both analytically and through simulation results. A DSP related method, i.e., folding, is employed for reducing the energy dissipation for very low speed applications.

- **Asynchronous Sub-V_T Operation**
  Energy dissipation reduction by asynchronous operation is shown by analyti-

cal model derivation, numerical simulations and application to real-world circuit examples. Current sensing completion detection for sub-threshold regime is proposed and implemented. An EDA flow for de-synchronization of a cardiac event detector is developed and employed for implementing an asynchronous version of the event detector.

1.2 Thesis Overview

The overview of the thesis is as follows. A brief overview of low-power and asynchronous operations are given in Chapter 2. High level energy model for sub-V_T operation is presented in Chapter 3. Furthermore, the analysis flow for the application of the model is explained. In Chapter 4 asynchronous and synchronous operations are compared in terms of their energy efficiency in the sub-threshold regime. Basics of digital cardiac event detection as well as the architecture of the wavelet based R-wave cardiac event detector are explained in Chapter 5. Process selection methodology for energy minimization is presented in Chapter 6. Moreover, optimum process selection is applied to the R-wave event detector for energy minimization. Hardware optimization techniques are examined and applied to the R-wave event detector in Chapter 7. Chapter 8 is dedicated to the current sensing completion detection system. In this chapter details of the system as well as trade-offs during the implementation are given. In Chapter 9 completion detection system is slightly modified and used to implement an asynchronous version of the cardiac event detector. Finally, Chapter 10 presents a summary of the research and draws conclusions.

Chapter 2

Digital Sub-threshold and Asynchronous Operation

In this chapter, sub-threshold digital circuit design and asynchronous operation are introduced. First, basic low power design methodologies are reviewed. Afterwards, sub-threshold operation is explained followed by a survey of sub-threshold circuit implementations in the literature. In the last part of this chapter an introduction to asynchronous circuit design is made, and various asynchronous design approaches as well as recent implementation examples are presented.

2.1 Low Power Digital Design

In this section first we present the distinction between power and energy and their importance for different applications. Afterwards, we give a brief overview of low power/energy design methods.

2.1.1 Energy and Power

Power consumption has become one of the greatest challenges as semiconductor devices are scaled. Scaling and being able to produce chips that contain hundreds of millions of devices and operate at multi GHz frequencies bring challenges related to power distribution and heat removal. For example, modern microprocessors use hundreds of pins and multiple interconnect layers for just power delivery [9].

Power consumption of static CMOS circuits arises from two main mechanisms: *Static power* which is the result of leakage paths between the supply rails, and *dynamic power* which is due to the switching of capacitive loads. During switching, for a brief

time, both NMOS and PMOS transistors are turned on at the same time. This causes a *short-circuit* current to flow between the power rails. Usually short-circuit current account to a small percent of the dynamic power (5-10%) so generally included in the dynamic power consumption. Thus, total power consumption of a digital block is specified as

$$P_{total} = P_{dynamic} + P_{leakage} + P_{short-circuit}, \tag{2.1}$$

and

$$P_{total} = \alpha C V_{DD}^2 f + V_{DD} I_{leakage} + \alpha V_{DD} Q_{short-circuit} f, \tag{2.2}$$

where α is the activity factor, C is the total switched capacitance, V_{DD} is the supply voltage, f is the operating frequency, $I_{leakage}$ is the leakage current of the circuit, and finally $Q_{short-circuit}$ is the total short-circuit charge per transition. It is immediately observable from equation (2.2) that reducing the supply voltage decreases the power consumption.

Energy is the proper metric for battery powered systems as the energy stored in a battery is not infinite in capacity; once a battery is empty, it either needs to be replaced or recharged, that is, energy is dissipated. Total energy dissipation over a clock period T is related to power consumption as

$$E_T = \int_0^T P_{total} dt, \tag{2.3}$$

and

$$E_T = \alpha C V_{DD}^2 + V_{DD} I_{leakage} T + \alpha V_{DD} Q_{short-circuit}. \tag{2.4}$$

By examining equation (2.4), it is seen that only leakage energy is dependent on the operating speed, and like power consumption, it is possible to reduce energy dissipation by lowering the supply voltage. Although reducing the supply voltage is the most effective energy reduction method, it results in reducing the operating frequency. For applications where the operating frequency as well as the power consumption are important, different low-power design methodologies, which are briefly overviewed next, have been proposed by researchers.

2.1.2 Low Power Design Methodologies

In the literature there are various low power design methodologies that try to reduce total power consumption:

- **Clock Gating**
  In synchronous applications a significant amount of power, i.e., $15 - 45\%$, is consumed by the clock tree [10]. Furthermore, the switching clock signal causes a lot of unnecessary switching activity. In [11] clock signal that is fed into the idle modules is stopped to reduce total power consumption. The clock signal to different modules is gated by a control signal to reduce the dynamic power consumption of the clock tree and the idle modules.

- **Multiple Supply Voltages**
  Dynamic power is directly proportional to the supply voltage. Hence, reducing the supply voltage reduces dynamic power consumption. However, operating speed of the circuit decreases due to lower operating voltage. Therefore, to be able to reduce power consumption and not to sacrifice operating speed, multiple supply voltage domains are created. Clusters with critical delay are supplied with a higher voltage not to reduce the operating speed, whereas non-critical clusters are supplied with a lower supply value to reduce the power consumption [12]. Use of multiple supply values result in supply generation and routing overhead as well as the power and area overhead due to level-converters.

- **Dynamic Voltage Frequency Scaling**
  In this method both voltage and frequency are changed dynamically based on the operating conditions and speed requirements. When high speed operation is required, the supply voltage is increased as well as the clocking frequency of the system resulting in higher power consumption. For cases where the power consumption is of primary concern, the supply voltage is reduced together with the clock frequency, resulting in lower power consumption.

- **Multi-Threshold**
  In the Multi-Threshold CMOS (MTCMOS) technique a high-threshold footer and/or header transistor is used to disconnect a circuit from the power rails during standby [13]. MTCMOS technique reduces the standby leakage power but incurs a speed penalty due to the sleep transistors.

- **Dual-Threshold**
  Dual-threshold technique employs both high- and low-threshold transistors. Low-threshold transistors are used on the critical paths while high-threshold ones are used for the rest of the circuit. This approach effectively reduces the leakage power consumption both in active and standby modes as the leakage

current in equation (2.2) is reduced. Both dual-threshold and multi-threshold techniques may be applied together to further reduce power consumption.

Although the presented techniques reduce the power consumption, they do not necessarily result in lower energy dissipation. It is seen from equation (2.4) that dynamic energy, i.e., $\alpha C V_{DD}^2$, is independent of the operation frequency and reduces quadratically with supply voltage scaling. For systems where energy dynamic energy dissipation dominates the total energy, reducing the supply voltage result in significant energy savings. However, supply voltage scaling increases the critical path delay of the circuit, reducing the maximum clocking frequency.

For systems where low energy dissipation and long battery life are the primary concerns, energy minimization is the main target. It was first shown in [14] and [15] that global energy minimum operating point of a circuit, if exists, occurs in the sub-threshold regime. Thus, in this thesis, we explore digital circuits that operate in the sub-threshold regime for energy minimization and enhanced energy efficiency.

2.2 Sub-threshold Operation

2.2.1 MOS Sub-threshold Operation

In digital applications an ideal field effect transistor (FET) has a large on current (I_{on}) when the gate overdrive, i.e., $V_{GS} - V_T$, is high and zero off current (I_{off}) when the gate overdrive is zero. Though in reality, there is always current flowing in the FET even for zero overdrive voltage. This off current causes standby power consumption overhead as well as failure in dynamic logic circuits and memories. Especially with lower threshold voltages with each successive technology, the effects of non-zero off current are more pronounced.

Sub-threshold drain current is the current flowing between the drain and the source of a metal-oxide-semiconductor field-effect transistor (MOSFET) when the transistor is operating in the sub-threshold regime. MOS transistors operate in the sub-threshold regime when the gate-to-source voltage of the transistor is lower than the threshold voltage. This region of operation is also referred as weak inversion region [15]. In sub-threshold region of operation, the current conduction is due to diffusion, unlike the above-threshold operation where the current conduction mechanism is majority carrier drift [16, 17]. CMOS digital circuits operate in sub-threshold regime when the supply voltage is lower than the threshold voltage (V_T) of the transistors.

In this regime sub-threshold leakage currents are used for charging and discharging of the node capacitances.

In sub-threshold operation, the channel of the MOS transistor is weakly inverted, i. e., no part of the channel is inverted moderately or strongly (above-threshold), and the currents flow by diffusion [17]. The drain current of an n-channel MOS transistor operating in this regime is given by [15]

$$I_{DS} = I_S \exp \frac{V_{GS} - V_T}{nU_T} \left(1 - \exp \frac{-V_{DS}}{U_T}\right), \tag{2.5}$$

where n is a process dependent term called slope factor and is typically in the range of $1.3 - 1.5$ for modern CMOS processes. The value of n depends on the depletion region characteristics of the transistor, i. e., $n = 1 + C_d/C_{ox}$. V_{GS} and V_{DS} are the gate-to-source and drain-to-source voltages, respectively. The parameter I_S is the specific current which is given by,

$$I_S = 2n\mu C_{ox} U_T^2 \frac{W}{L}, \tag{2.6}$$

where μ is the mobility of carriers, C_{ox} is the gate oxide capacitance per unit area, U_T is the thermal voltage whose value is $26mV$ at 300K and $\frac{W}{L}$ is the aspect ratio of the transistor. By setting $V_{GS} = 0$ for $V_{DS} >= 4U_T$, saturation off current is specified as

$$I_0 = I_S \exp \frac{-V_T}{nU_T}. \tag{2.7}$$

Thus, equation (2.5) is written as:

$$I_{DS} = I_0 \exp \frac{V_{GS}}{nU_T} \left(1 - \exp \frac{-V_{DS}}{U_T}\right). \tag{2.8}$$

Due to the second term in equation (2.8), the drain current is 0 when $V_{DS} = 0$ but reaches its maximum value and saturates with V_{DS} values higher than a few U_T. Based on equations (2.7) and (2.8), the drain current of a MOS transistor in sub-threshold region shows exponential dependence on the gate-to-source, drain-to-source voltages, slope factor and the operating temperature.

Figure 2.1 shows the drain current of a MOS transistor for changing gate-to-source voltage. The sub-threshold, moderate inversion and strong inversion (above-threshold) regions are marked on the figure. At low values of V_{GS} (weak inversion), the drain current varies exponentially with the gate to source voltage. Digital CMOS

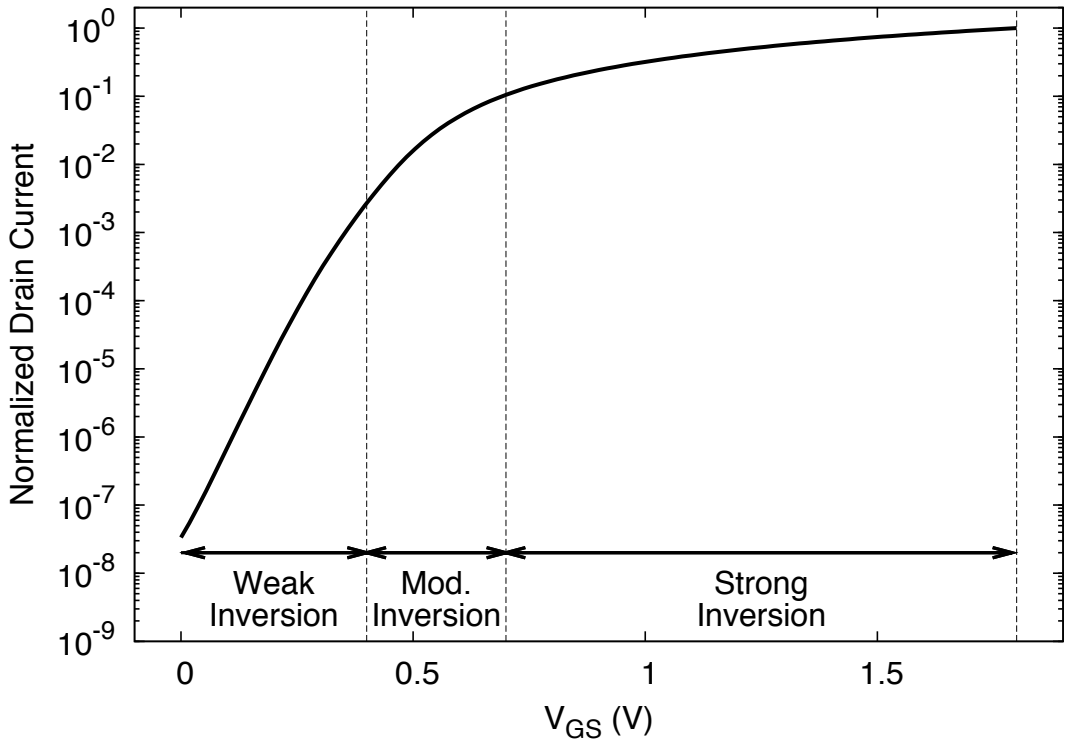


Figure 2.1: A standard CMOS 0.18 μm process NMOS transistor drain current, I_{DS}, versus changing gate-to-source voltage, V_{GS}. The drain is connected to the nominal supply voltage ($V_{DD} = 1.8V$) for the simulations. The operating regions of the NMOS transistor are marked.

operation in this regime is explored in this thesis for minimizing energy dissipation and enhancing energy efficiency.

2.2.2 Digital Sub-threshold Operation

Although it was first shown by Swanson as early as 1972 that CMOS digital operation may be realized with ultra-low supply voltages [18], sub-threshold digital design did not get high attention until recently. Sub-threshold MOS operation was applied to the design of analog circuits [19–22].

Sub-threshold digital operation is similar to above threshold operation except the ultra-low supply voltage, which has been reduced below the threshold voltage of the MOS transistors. For example, Figure 2.2 shows the sub-threshold voltage transfer curves of a static CMOS inverter designed in a standard 0.18 μm process. Threshold voltages of standard PMOS and NMOS transistors are -0.46 and 0.31 V, respectively.

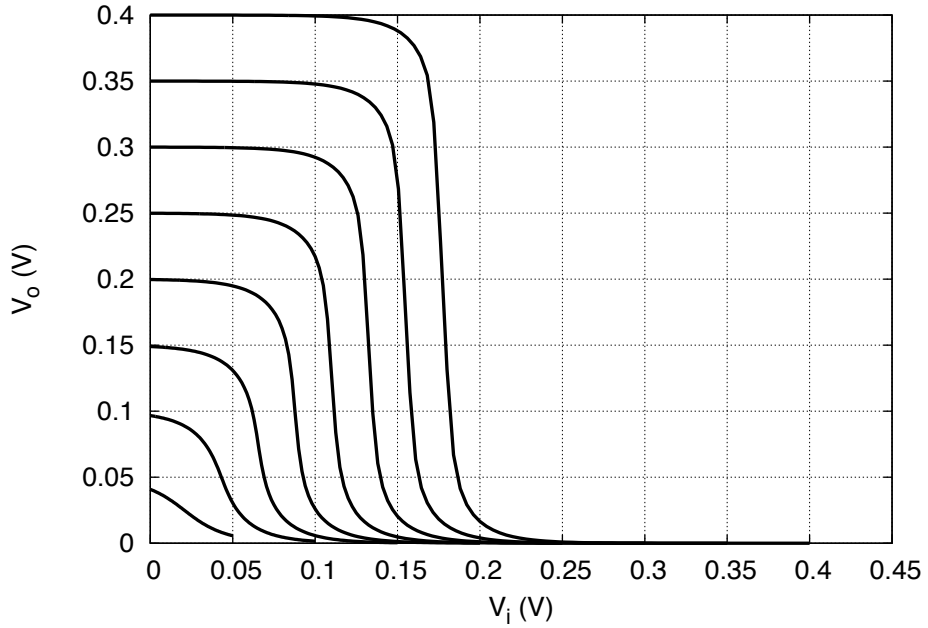


Figure 2.2: Voltage transfer characteristics of a static CMOS inverter gate. Process used is a standard 0.18 μm CMOS process. Different curves represent different transfer characteristics for changing supply voltage values in the sub-threshold regime.

Limits of supply voltage scaling are given in [18] and [15]. The limit is found to be $4U_T$ and is based on the the minimum operating voltage for obtaining an absolute gain of more than 1 from a simple inverter, and for guaranteeing bi-stability with sufficient voltage swing.

In [23] Soeleman investigated sub-threshold operation of CMOS and pseudo-NMOS logic families. The only comparison made in the paper was about the energy dissipation reduction with respect to operation at the nominal voltage. Simulations were run on simple gates and in a 0.35 μm process and energy savings up to two orders of magnitude were shown. Later in 2000 dynamic threshold MOS (DTMOS) operation in the sub-threshold regime was investigated in [24]. In this paper it was shown that DTMOS technique allowed faster operation when compared to static CMOS implementation.

In 2002, Kao presented a 175 mV digital circuit employing adaptive supply voltage and body bias architecture [25]. This research aimed to develop a theoretical model for predicting the optimal threshold voltage - supply voltage pair to minimize power consumption. Kim presented a dynamic threshold voltage SRAM in [26]. Body biasing was used to control the threshold voltage of each cache line trying to minimize

leakage energy in this paper. Wang investigated optimal supply and threshold voltage pairs in [27]. In this work energy and performance contours were used to determine the optimal supply scaling of supply and threshold voltages for low-performance applications.

Another real-world circuit implementation employing sub-threshold logic was presented in [28]. Kim et al. implemented a delayed least mean square adaptive filter working in the sub-threshold regime. The target application of the filter was hearing aids, for which ultra-low power operation is crucial. It was shown that the filter was able to process voice signals with a throughput of 22 kSamples at 400mV supply voltage.

Global minimum energy operation in the sub-threshold domain was first noted in [14] and [15]. In [15] sub-threshold operation was investigated using the EKV model [29]. In [15,30–33] the occurrence of minimum energy point in the sub-threshold regime was shown.

Device optimization for ultra-low power digital operation in the sub-threshold regime was investigated in several works. In [34] Paul et al. proposed different device designs by modifying the process technology. They showed that by modifying the devices for sub-threshold operation, delay and power-delay-product (PDP) of the inverter chains improved by 44% and 51%, respectively. In [35] work on device, circuit and architecture were employed together to further reduce power consumption of sub-threshold digital circuits. Hanson et al. examined the affects of process scaling and device degradation on sub-threshold operation in [36]. They predicted 60% reduction in I_{on}/I_{off} while moving from 90 nm process node to 32 nm.

Transistors that belong to the memory arrays often account for the majority of the transistor count in a processor or system-on-a-chip [37]. To be able to implement a whole system for energy minimum operation memory sub-systems working in the sub-threshold regime are required. In [38] sub-threshold static random access memory (SRAM) was investigated for feasibility across process technologies. It was shown that SRAM sub-threshold operation is possible but due to the intra-die variations SRAM cells are vulnerable to process variations. It was concluded that the benefits of sub-threshold operation for the SRAM blocks in terms of power consumption diminish as process technologies scale. In [39] Chen et al. presented a sub-threshold SRAM fabricated on a 130 nm CMOS process. The presented SRAM operates down to 216 mV at 28 kHz and at 310 mV for a 1 MHz clock rate. Kulkarni further reduced the operating voltage of an SRAM down to 160 mV employing Schmitt Triggers for increasing the static noise margin (SNM) of the SRAM cells [40]. In [41] standard 6

transistor (T) SRAM cell was modified to alleviate write failure, degraded read SNM and bitline leakage problems. The resulting 10T SRAM was shown to operate below 380 mV in a 65 nm process technology.

In [42] and [43] effects of the process variation on the minimum energy operation were investigated. It was shown that sub-threshold logic was affected by random dopant fluctuations (RDF) more than above-threshold operation. It was noted that special care is needed for robust design of sub-threshold circuits.

For system level sub-threshold power management, Calhoun proposed ultra-dynamic voltage scaling for sub-threshold operation [44]. In the proposed system local voltage dithering is employed for minimizing the energy dissipation of the circuit under varying performance requirements. Another example of system level sub-threshold power management was presented in [45]. In this paper an example circuit with embedded minimum energy tracking DC-DC converter loop was presented.

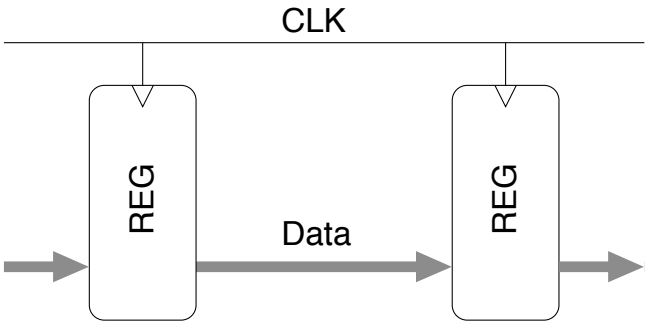
Circuits operating at these extreme low supply voltages work at much lower speeds, for example the FFT processor presented in [46] works with a maximum clock frequency of 10 kHz with a power supply of 350 mV. Even so, their extreme low power consumption results in excellent power delay product (PDP) values, making such circuits very interesting candidates for ultra-low power applications which do not have very high processing requirements.

2.3 Asynchronous Operation

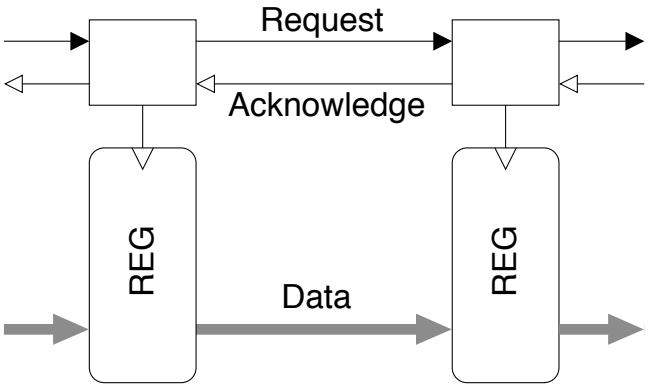
In an asynchronous circuit the parts that make up the circuit are largely autonomous. They are not governed by a clock circuit or global clock signal, but instead need only wait for the signals that indicate completion of instructions and operations. These signals are specified by simple data transfer protocols. The data through the stages propagate by means of handshake signals that signal propagation of the data, see Figure 2.3a. This digital logic design is contrasted with a synchronous circuit which operates according to clock timing signals (Figure 2.3b).

Advantages of employing asynchronous circuits are as follows:

- **No clock skew**
  Clock skew is the time difference in arrival times of the clocking signal to different parts of the circuit. As the clock rate of synchronous circuits increase, less variation in timing can be tolerated for correct operation. For example, if clock propagates slower than the path from one register to another, data may



(a) Synchronous operation.



(b) Asynchronous operation.

Figure 2.3: Difference between synchronous and asynchronous implementations.

be latched by both registers, destroying the integrity of the latched data. Asynchronous circuits are not affected by the clock skew problems due to the absence of a global clocking signal.

- **Average-case performance**
  Synchronous circuits must be operated at or below the speed of the longest path in the combinational logic. During the operation, the circuit may have settled to its final value before the next clock signal. This results in worst-case performance for the circuit. In asynchronous circuits with completion detection the average-case performance can be achieved. For circuits where the worst-case and average-case delays are significantly different, substantial improvements in the operating speed can be realized [47].

- **Lower power consumption**
  In synchronous circuits each register dissipates energy during each clock cycle, regardless of the change at the input of the circuit and/or the state change [48]. In asynchronous circuits only the parts of the circuit that are processing data are activated, reducing the total dynamic power consumption. Although in synchronous designs clock gating can be applied for reducing the register activity, very fine grain clock gating, that is achievable only by asynchronous design, is not possible. Another reason for theoretical lower power consumption of asynchronous circuits is the lack of a clocking network. In digital designs clocking network power consumption constitute a significant portion of the total power budget [49, 50]. Owing to the lack of a clock distribution network, hence repeaters and signal shapers, asynchronous designs have the possibility to consume less power when compared to their synchronous counterparts.

  One example of low-power asynchronous circuits is an IFIR filter bank for a digital hearing aid [51]. In [52] a detailed study of asynchronous circuits for low power and energy efficient applications can be found.

- **Design reuse - modularity**
  When compared to synchronous circuits which operate with a global clocking signal, asynchronous circuits implement distributed local control blocks. This control blocks rely on the use of handshaking signals. Thus it is a matter of simply connecting the handshaking signals of different blocks for composing a bigger system utilizing already implemented digital blocks [53].

- **Reduced electromagnetic emission**
  In an asynchronous system different parts of the system may be activated at arbitrary times. This results in current peaks in the supply network spread over time, reducing the current peak amplitudes, hence lower electromagnetic emission.

There are several reasons why asynchronous circuits are not as common as synchronous designs. Firstly, asynchronous circuits are more difficult to design then their synchronous counterparts. Synchronous designers are not concerned with what is going on between the latches/registers of a design as long as the data at the input of the memory elements is stable before the next clock signal. In contrast, asynchronous designs need to be free of logic hazards in all the levels of abstraction [47] and switching activity must be properly ordered not to cause wrong data propagation. Secondly,

asynchronous design methodologies are not fully supported by commercial Electronic Design Automation (EDA) tools, necessitating custom modifications to EDA tools for asynchronous design implementation.

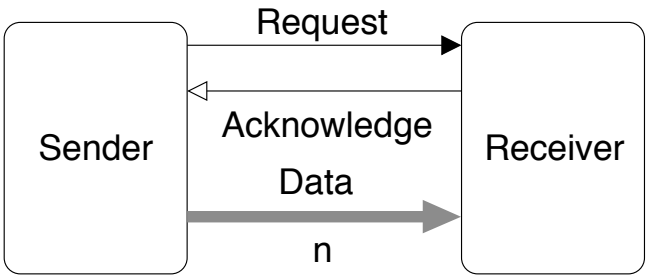
Functional redundancy may be employed to explicitly model computation flows without using a global clocking signal. This guarantees the correct switching order of the logic circuitry. Using logic circuits to ensure correct circuit behavior is usually costly and impractical [54]. Therefore during the design of asynchronous circuits some timing assumptions are made. Most common timing assumptions/models are as follows:

- **Bundled-delay (BD)** This model is similar to that of the synchronous circuits where a matched delay to the critical path of a purely combinational logic island is employed. It is to be guaranteed that delay of all the paths in the combinational island are smaller than that of the matched delay. While implementing a BD circuit, same gate library as the combinational logic may be used. This model is the closest to synchronous circuits and the easiest to implement because it is possible to implement BD circuits by using well-known commercial EDA tools.

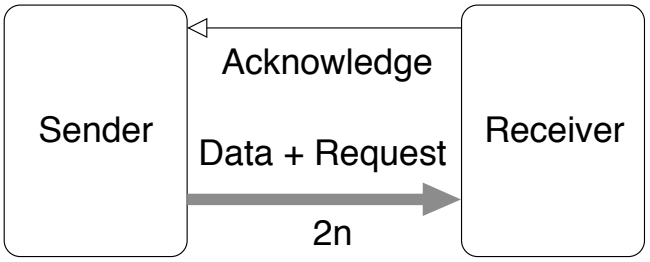
- **Delay-insensitive (DI)**
  Delay-insensitive circuits work correctly under the assumption that gate and wire delays are *unknown* but *finite*. DI model allows the most robust asynchronous circuit implementation but the class of DI implementations are very limited in practice [55].

- **Quasi-delay-insensitive (QDI)**
  In this model it is assumed that the skew between the critical wires, i.e., forks, is less than the gate delay. This assumption is called the isochronic fork and can be applied to a group of gates and wires on a small area of the chip.

Multiple design flows from the academic research groups and the industry exist for implementing different classes of asynchronous circuits. Recent design flows that are employed for real world designs are; De-synchronization [56], Null Convention Logic (NCL) employed by Theseus Logic [57] and bundled data circuits (Figure 2.4a) implemented using Haste, a proprietary language of the Handshake Technology. In this language parallel structures may be implemented with ease unlike Verilog or VHDL implementations. Also in Handshake implementation, de-synchronization may be implemented in the design flow. Except the NCL flow, other flows rely on matched



(a) Single rail - bundled data implementation.



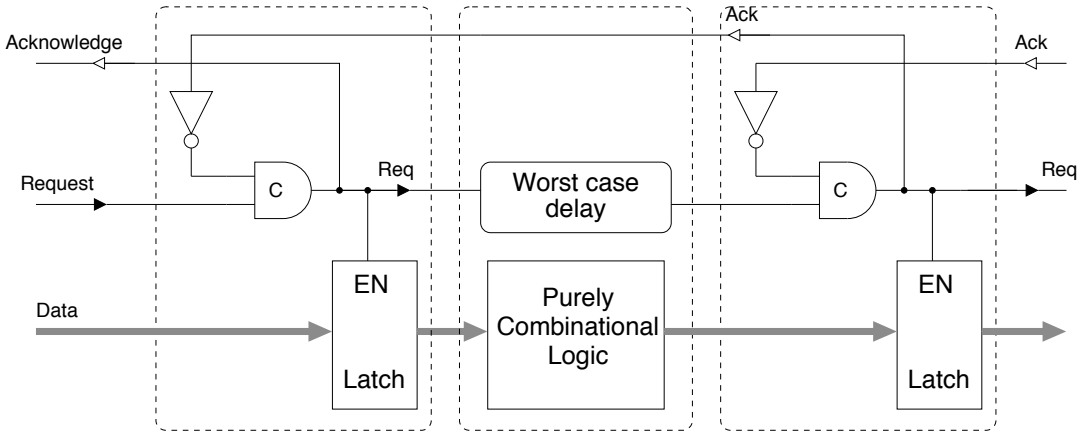
(b) Dual rail implementation.

Figure 2.4: Bundled data and dual rail asynchronous circuit implementations.

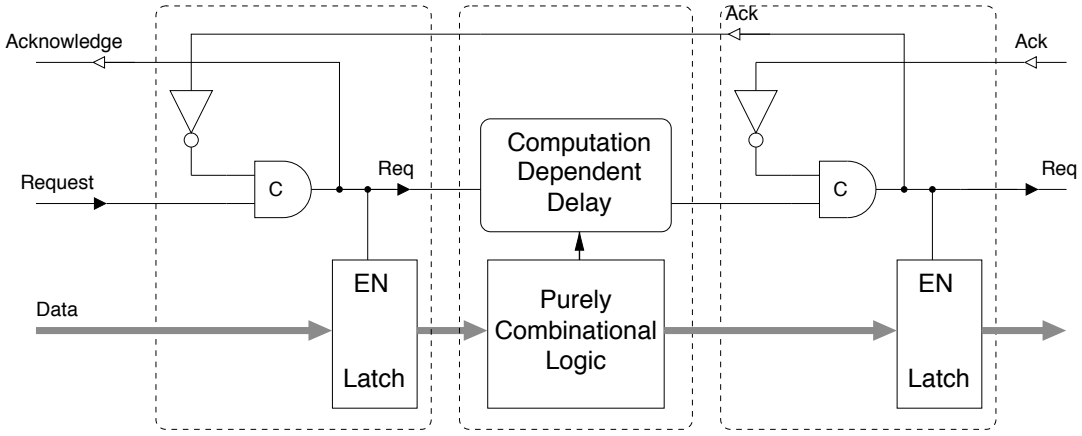
delay implementations, thus not realizing average case performance of asynchronous circuits. For the average-case performance to be realized, means of completion detection of data processing are needed.

Completion detection in asynchronous circuits can be implemented using different approaches. One such approach is using dual rail signaling. The request signal is encoded inside the data signals in a dual-rail protocol. In a dual-rail implementation each data bit is transferred using two wires as shown in Figure 2.4b. Therefore for an n-bit wide data bus for the bundled-data case, $2n + 1$ wires need to be implemented for the dual rail case. Also in the dual rail case the number of logic gates increase because of the inherent redundancy in the protocol.

Another approach for implementing completion detection is sensing the current consumption of the circuits and generating a handshaking signal when the sensed circuit completes the computation. There are multiple examples of current sensing completion detection in the literature [58–61]. These methods rely on bipolar transistors, and/or resistors with high values, both of which are not always available in a



(a) Fixed delay implementation



(b) Computation based delay implementation

Figure 2.5: 4-phase bundled data pipeline (After [1]).

standard process, or come as a process option with additional cost. The requirements on the bipolar transistors and resistors in these solutions set practical limits for the detection of current values in the μA-to-mA range.

As an example, a basic 4-phase, bundled data asynchronous pipeline is shown in Figure 2.5. In Figure 2.5(a) the most common implementation (fixed delay) is shown. In this implementation the *Request* signal is delayed by an amount equal to the worst case delay of the purely combinational logic stage. This implementation is a similar approach to the synchronous operation, where unnecessary delay, which is fixed regardless of the logic computation time, is introduced. Another version of the 4-phase bundled data pipeline is shown in Figure 2.5(b), where the delay mimics the computation completion time of the logic block as much as possible. In the last

part of this thesis we focus our attention on the implementation of a variable delay element, i.e., current sensing completion detection, for reducing the overall energy dissipation.

An extensive evaluation of asynchronous designs and their properties are beyond the scope of this thesis. More detailed information can be found in references [1, 47, 48, 53, 62].

Chapter 3

A High Level Sub-threshold Energy Model

In this chapter an energy model for digital systems working in the sub-threshold regime is presented. The energy dissipation model is comparable to other models that have been published previously [15,30,32]. In [15] Vittoz investigated and proved the energy-minimum operation property of sub-threshold logic. In the model developed, an expression for the energy-minimum operating voltage was not derived and energy-minimum operating point was shown by numerically inverting the duty factor for minimum energy numerically. In [30] occurrence of the energy-minimum operating voltage was shown again but the energy-minimum operating voltage equation was solved by curve fitting. In [32] Calhoun solved the sub-threshold energy-minimum operating voltage analytically. In Calhoun's model average switched capacitance and average leakage current were specified as parameters and they were extracted from SPICE level simulation results. However, all the previous models focused only on synchronous sub-threshold operation. As will be shown in this chapter, we developed our model for asynchronous operation and later extended to include synchronous operation as well. Furthermore, the developed energy model gives accurate results without requiring computation and time intensive SPICE simulations, making the model usable for all phases of the design flow. Another benefit of the model is that it allows the designers to be able to characterize the sub-threshold energy efficiency of different circuit implementations efficiently.

3.1 Asynchronous Energy Model

In an asynchronous system, the operation of the system is both dictated by the switching and delay properties of a system, and the external request and acknowledge signals. During the development of the model for simplifying the mathematical operations and derivation of the equations, following assumptions are made:

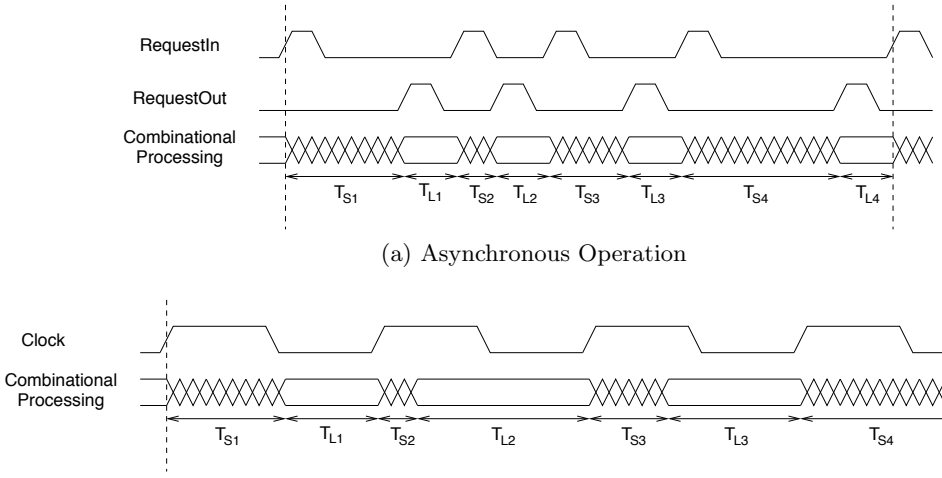
Assumption-1: As soon as the asynchronous block finishes processing the current data, a new data input may be applied, i.e., there is no idle time between data inputs to the asynchronous block.

Assumption-2: The energy dissipation and processing delays of the circuit per computation are randomly distributed. This assumption is guaranteed by applying a randomly distributed data set to the input of the circuit.

Assumption-3: Synchronous circuits work at their maximum speed, i.e., clocked at a speed equal to their critical path delay.

Assumptions 1 and 3 guarantee that synchronous and asynchronous operations are compared fairly in terms of energy dissipation. Together they guarantee that circuits run at their maximum speeds possible, hence dissipating minimum leakage energy, while working at the lowest energy-minimum operating voltage possible. Assumption-2 is used to simplify the statistical properties of the energy model. As long as the energy dissipation and processing delay of the circuit are randomly distributed with a mean, the model may be applied to any circuit that is operated with an arbitrary data set. While deriving the energy dissipation model, assumptions made will be emphasized wherever required.

The conceptual operation of an asynchronous block is shown in Figure 3.1a. In this example, the circuit is observed for an arbitrary time frame T, where four distinct sets of input data are processed. The time spans where the circuit is purely *leaking* (waiting for handshake completion) are denoted with T_{Li} and the time spans where the circuit is both *switching* (processing input data) and *leaking* are denoted with T_{Si}. In this diagram we assume that as soon as the *RequestOut* signal is lowered, *RequestIn* goes high (Assumption-1) and the stage begins the *switching* phase; so all the purely leaking time spans (T_{Li}s) are fixed, equal, and represent the asynchronous communication overhead.



(a) Asynchronous Operation

(b) Synchronous Operation

Figure 3.1: Timing diagram showing the same logic block operating in (a) asynchronous and (b) synchronous modes.

To calculate the energy dissipation in an arbitrarily long time frame T, we monitor the operation of the circuit. During this time frame, we assume that the number of energy dissipating computations is N. In any static CMOS digital system the total energy dissipation is given by

$$E_T = E_{dynamic} + E_{leakage} + E_{short-circuit}, \tag{3.1}$$

where $E_{dynamic}$ is the dynamic energy dissipation due to the switching of the capacitances, $E_{leakage}$ is the leakage energy dissipation of the circuit during the whole time the circuit is supplied by an energy source, and $E_{short-circuit}$ is the short-circuit energy dissipation due to the short-circuit current flowing from the supply to the ground during switching. In our analysis we neglect the contribution of the short-circuit component, as it is known to contribute only a small portion of the overall energy dissipation [15].

Elaborating the general energy dissipation equation (3.1), the dynamic energy dissipation during the ith time span is expressed as

$$E_{dynamic_i} = e_i C_{tot} V_{DD}^2, \tag{3.2}$$

where e_i is a scaling parameter that defines the switching property of the circuit for a specific input data transition, and C_{tot} is the maximum possible switched capacitance

of the circuit. The switching energy scaling parameter e_i is in the range $[0, 1]$. Without loss of generality we express e_i as a single value in a random process e (Assumption-2). By defining e as a random process, we may define a mean μ_e and using this mean the average dynamic energy for N computations is expressed as

$$E_{dynamic} = N\mu_e C_{tot} V_{DD}^2. \tag{3.3}$$

In (3.2) and (3.3), the total capacitance C_{tot} may be normalized in terms of the total inverter capacitance using a capacitance scaling factor $k_{cap-logic}$ as $C_{tot} = k_{cap-logic} C_{inv}$ where C_{inv} is the switched capacitance of an inverter.

Assuming even during the *switching* time, most of the cells in the circuit are *leaking*, the leakage energy dissipation during the observation period T is defined as

$$E_{leak} = k_{leak} I_0 V_{DD} T, \tag{3.4}$$

where k_{leak} is the average leakage scaling factor of the circuit, and I_0 is the leakage current of a single inverter. Total average leakage current of the circuit is calculated as $k_{leak} I_0$ from equation (3.4). The average leakage parameter k_{leak} is obtained from the synthesis results by accumulating the individual average leakage currents of the digital gates, where the average leakage current is the mean of the leakage current for all the combinations of input vectors applied to the logic gate, and normalizing the result to the average leakage current of a single inverter. Combining equations (3.1), (3.3) and (3.4), total energy dissipation during the monitoring time frame T is defined as

$$E_T = N\mu_e k_{cap-logic} C_{inv} V_{DD}^2 + k_{leak} I_0 V_{DD} T. \tag{3.5}$$

From Figure 3.1a, the total time spent during switching is

$$T_S = \sum_{i=1}^{S} T_{s_i}, \tag{3.6}$$

and based on the switching/timing statistics of the circuit we are analyzing, any switching activity (T_{s_i}) in Figure 3.1a may be defined as

$$T_{s_i} = d_i k_{crit} T_{sw\_inv}, \tag{3.7}$$

where d_i is a scaling parameter that defines the delay properties of the circuit processing the current data, k_{crit} is a coefficient that defines the critical path delay of

the circuit in terms of the inverter delay, and T_{sw_inv} is the delay of an inverter. The scaling parameter d_i may take any value in the range $[0, 1]$. Like e, if d is modeled as a random process with the mean μ_d (Assumption-2), the total time spent during switching is calculated approximately as

$$T_S = N\mu_d k_{crit} T_{sw\_inv}. \tag{3.8}$$

During the observation frame T, N switchings and N handshakes take place, so T is expressed as

$$\begin{aligned} T &= T_S + T_L \\ &= N\mu_d k_{crit} T_{sw\_inv} + N k_{com_{oh}} k_{crit} T_{sw\_inv}, \end{aligned} \tag{3.9}$$

where $k_{com_{oh}}$ is a parameter defining the overhead caused by the asynchronous communication in terms of the critical path delay of the purely combinational logic block. The delay of an inverter working in the sub-threshold regime is given in [15] as

$$T_{sw\_inv} = \frac{C_{inv} V_{DD}}{I_0 e^{V_{DD}/(nU_t)}}. \tag{3.10}$$

By introducing equation (3.10) into equation (3.9), we get the total observation time as

$$T = N k_{crit} \frac{C_{inv} V_{DD}}{I_0 e^{V_{DD}/(nU_t)}} (\mu_d + k_{com_{oh}}). \tag{3.11}$$

Introducing equation (3.11) into equation (3.5), the final total energy dissipation equation for N switchings is specified as

$$E_T = N C_{inv} V_{DD}^2 \left[\mu_e k_{cap-logic} + k_{crit} k_{leak} (\mu_d + k_{com_{oh}}) e^{-V_{DD}/(nU_t)} \right]. \tag{3.12}$$

By setting $N = 1$ in equation (3.12), average energy dissipation per operation may be found. The optimal operating voltage for minimum energy operation is found by taking the derivative of (3.12) with respect to V_{DD}, equating the result to 0, and solving for V_{DD}. Thus, the energy-minimum operating voltage is given in (3.13) as

$$V_{opt-async} = 2nU_t - nU_t W_{-1} \left[-\frac{2e^2 k_{cap-logic} \mu_e}{k_{crit} k_{leak} (k_{com_{oh}} + \mu_d)} \right], \tag{3.13}$$

where W_{-1} is the -1 branch of the LambertW function. LambertW function is examined thoroughly in [63] and briefly explained in the Appendix A. All the k-parameters in equations (3.12) and (3.13) are found from the synthesis results of the digital cir-

cuit, and the μ-parameters are extracted from switch level (synthesized Verilog) simulation results. Hence, the total simulation time for characterizing the sub-threshold performance of the circuit is reduced greatly, compared to SPICE-level simulations.

3.2 Synchronous Energy Model

A similar modeling approach is performed to model a synchronous system operating as shown in Figure 3.1b. By assuming only one set of data is processed during a clock period ($N = 1$) and the clock period is equal to the critical path of the logic circuit, i.e., ($T = k_{crit}T_{sw\_inv}$); the energy per operation is derived from equations (3.11) and (3.12) as

$$E_T = C_{inv}V_{DD}^2\left[\mu_e k_{cap-logic} + k_{crit}k_{leak}e^{-V_{DD}/(nU_t)}\right]. \tag{3.14}$$

As in the asynchronous case, by taking the derivative of (3.14), equating the result to 0 and solving for V_{DD}, we get the optimum voltage that realizes the minimum energy operation as

$$V_{opt-sync} = 2nU_t - nU_t W_{-1}\left[-\frac{2e^2 k_{cap-logic}\mu_e}{k_{crit}k_{leak}}\right]. \tag{3.15}$$

So far, development of the energy model assumed that the circuit under consideration operates at the maximum frequency that is imposed by the operating voltage, and hence operating with minimum leakage energy possible at that voltage. Usually this is not the case in real world applications, where the frequency of operation is dictated by external systems or circuitry. For such a case, equation (3.12) cannot be used to calculate the energy dissipation of a circuit. A model that is not constraining the leakage time by the maximum operating frequency needs to be developed. For externally speed constrained systems which work below the speed that is achievable at the energy-minimum operating point, we slightly modify equation (3.5) for synchronous operation as

$$E_T = \mu_e k_{cap-logic}C_{inv}V_{DD}^2 + k_{leak}I_0 V_{DD}T_{CLK}. \tag{3.16}$$

3.3 Model Implementation Flow

The application flow of the model to designs is presented in Figure 3.2. For the application of the model, some pre-characterization of the process and the standard cell libraries are required. First, the standard cell library is characterized for calculating the average leakage factor of each cell normalized to a chosen inverter implementa-

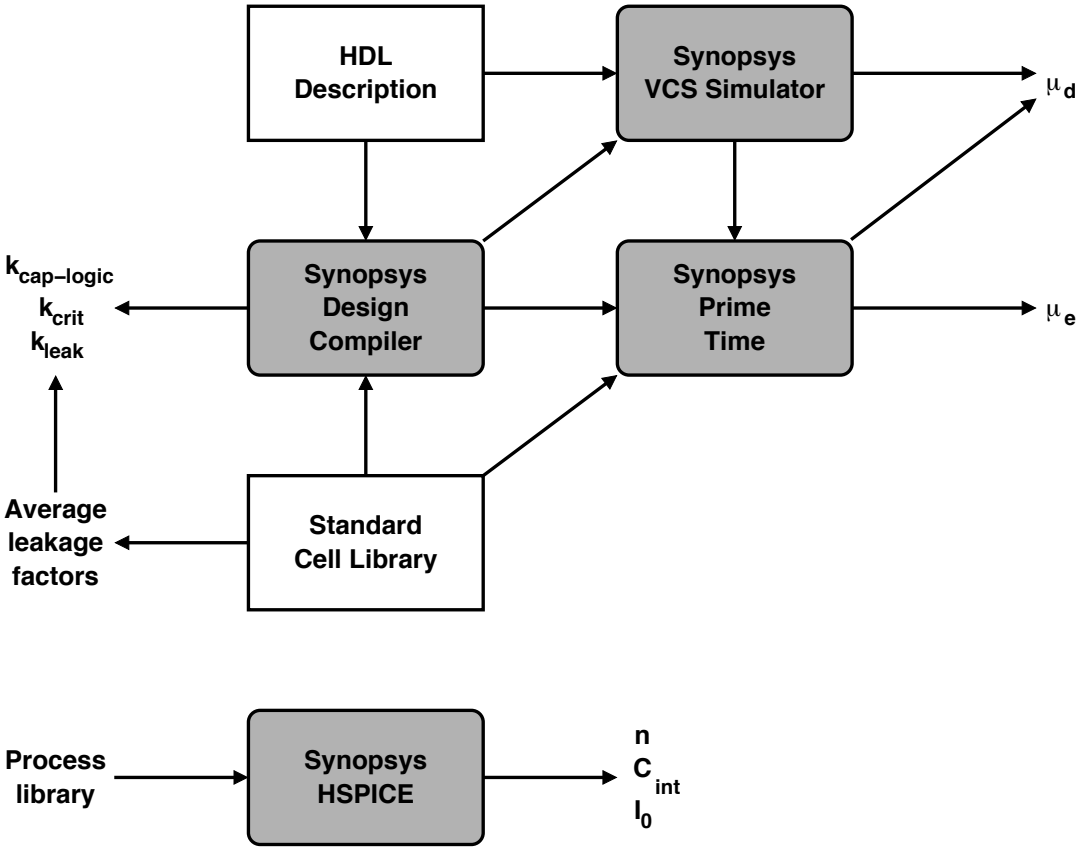


Figure 3.2: Model application flow emphasizing the tools used.

tion. This is achieved by processing the `.lib` file that is supplied by the foundry with custom developed AWK [64] scripts. Second, the process library is characterized for the slope factor n and the inverter internal switched capacitance C_{int}. During this step, inverter characterization and single transistor testbenches are created, and HSPICE is used for simulations. Required data is processed and fit to the model using Mathematica.

After the pre-characterization step, the flow begins with the usual implementation of the digital circuit using hardware description languages (HDLs) as in a standard flow. Design verification is done using an RTL simulator, e.g., ModelSim or Synopsys VCS, and then for the circuit synthesis using a standard cell library, Synopsys Design Compiler is used. During the synthesis process, detailed reports of the circuit properties are generated by the synthesis tool. Afterwards, the synthesis reports are processed using custom developed AWK scripts. Average leakage factors

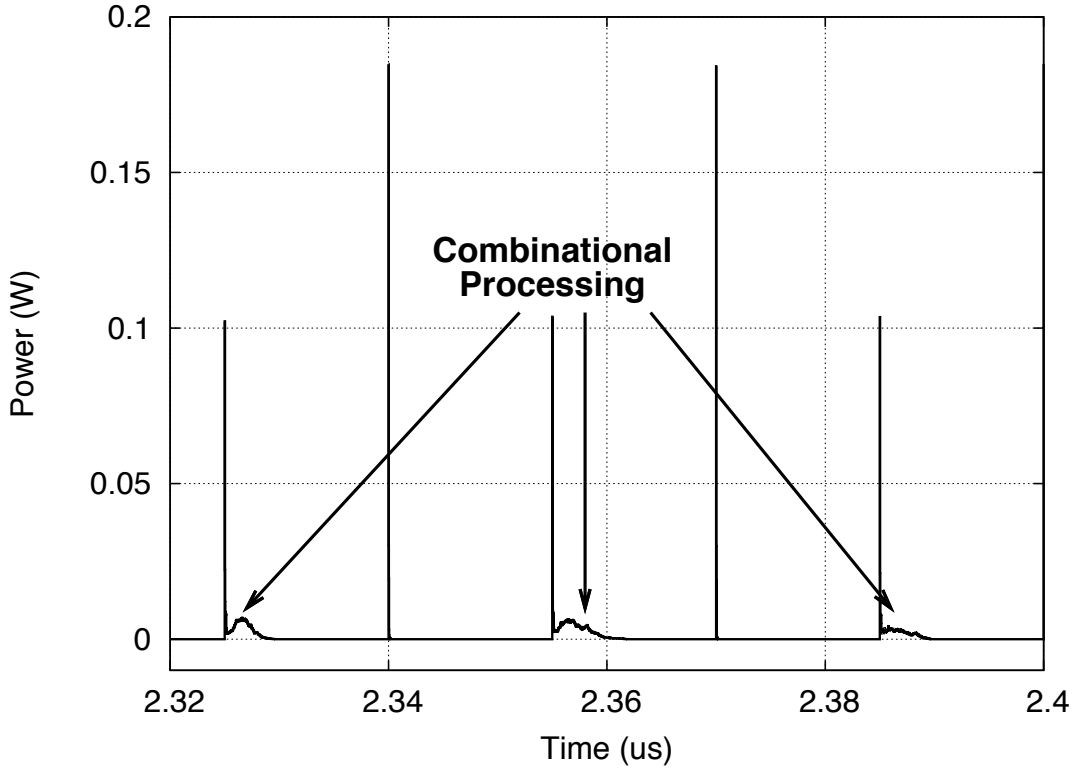


Figure 3.3: Sample Synopsys PrimeTime Power cycle-accurate power waveform to emphasize the current consuming combinational operations. Combinational processing timing information is used for calculating the delay distribution of a mixed combinational-sequential circuit.

from the pre-characterization step are used for the calculation of k_{leak}. As a result of this step, values of $k_{cap-logic}$, k_{crit} and k_{leak} are obtained.

Following the successful synthesis of the circuit, both the synthesized and behavioral model of the circuit are simulated using Synopsys VCS simulator. For the purely combinational circuits the output of the synthesized and behavioral circuits are compared in the testbench and the timing signals are written to a text file. Afterwards, from the resulting data, both the delay distribution and its average (μ_d) are calculated by post-processing. Following the simulation of the synthesized circuit, the power calculation is done in Synopsys Prime Time employing the wire-load models. A cycle-accurate waveform for the power consumption of the circuit is generated and post-processed using Perl scripts for calculating the energy distribution of the circuit. If the characterized logic block is not purely combinational, i.e., block contains memory elements such as registers and latches, comparison of the circuit outputs

are ineffective for delay calculations. For such a case, the delay distribution is calculated by post-processing the generated power waveform. A sample waveform from the Synopsys PrimeTime Power simulations is presented in Figure 3.3. In the figure, combinational processing regions are emphasized. Power spikes just before combinational processing occurs is due to the positive edge of the clock signal and spikes that are not followed by combinational processing are due to the negative edge of the clock signal. Thus, both the distributions and the average values of delay and energy characteristics of the circuit are obtained.

3.4 Accuracy of the Model

We verified the accuracy of the proposed model by both simulations and measurement of an implemented test circuit in a standard CMOS 0.18 μm process. ISCAS85 benchmark circuits [65] are used to verify the accuracy of the model with simulations, and a randomly generated lookup table (LUT) is used as the test circuit on the manufactured chip.

3.4.1 ISCAS85 Benchmark Circuits

We ran simulations to analyze the accuracy of the proposed model. The model parameters are gathered from the synthesis results and switch-level simulations. We simulated the energy dissipation values of the benchmark circuits in HSPICE by running transient simulations. During the simulations 1000 random vectors were applied to the benchmark designs working at their maximum allowed speed, i.e., speed set by

Table 3.1: Accuracy comparison for energy dissipation at the energy-minimum voltage.

Benchmark	Error (%)	Spice CPU Time	Model CPU Time
c499	-2.26	511.09	3.13
c880	3.49	351.35	4.16
c1355	-2.61	471.58	4.21
c1908	0.40	456.08	3.17
c2670	3.05	798.46	5.37
c3540	3.73	1845.05	5.36
c5315	2.98	2529.48	5.61
c6288	-1.79	10118.50	40.34
c7552	-1.70	4368.91	8.38

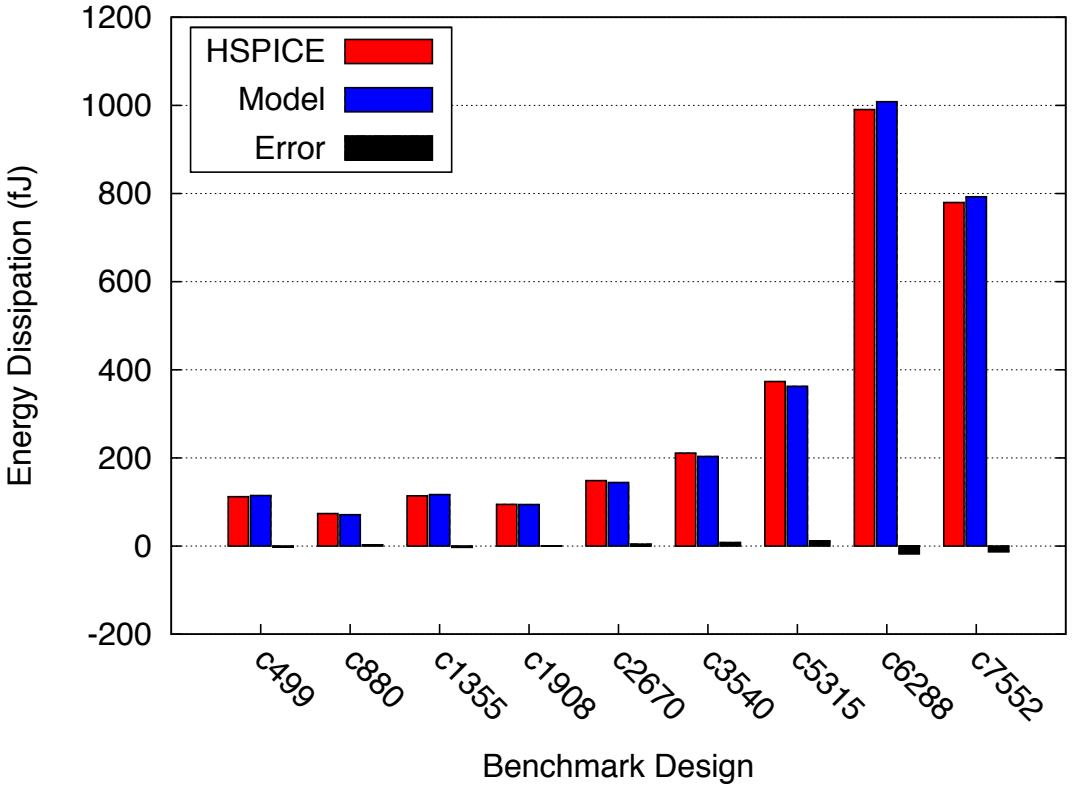


Figure 3.4: Comparison of the energy dissipation values gathered from the HSPICE simulation results and the model calculations based on the synthesis results.

the critical path of the circuit, at their synchronous energy-minimum operating voltages. Same test vectors were applied to the circuits for the switch-level simulations and their switching activity factors were calculated from the Prime Time results. The comparison of the two methodologies in terms of accuracy are shown in Figure 3.4. From the comparison of the HSPICE simulation results and the developed model, we calculate that our methodology predicts the energy dissipation values with less than 3.8% error for all benchmark circuits. The most important benefits of the developed model are usability in the early design phase and the low run time when compared to SPICE level simulations. The error values and the CPU time spent during the simulations for all the test circuits are given in Table 3.1. From the table it may be seen that the developed model and the simulation methodology may be used for early design decisions in a more complex system with very high accuracy and very low runtime.

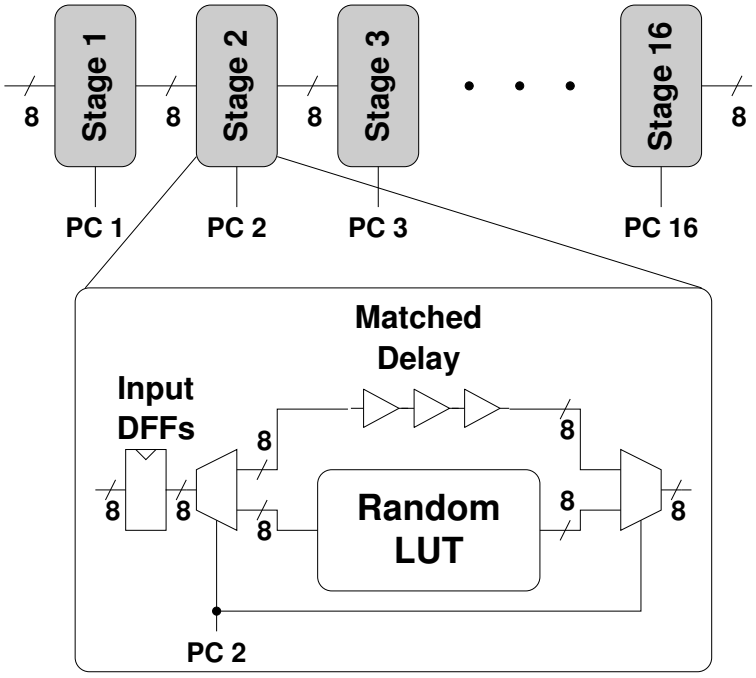


Figure 3.5: 16-stage pipelined circuit. Each stage includes the same randomly generated 8-bit input, 8-bit output look up table in parallel to a matched delay. The data to the stage is directed through one of the paths using a demultiplexer and a multiplexer.

3.4.2 Measurement Results

To verify the accuracy of the energy model with a manufactured implementation, a test circuit as shown in Figure 3.5 is fabricated. Each pipeline stage of the 16-stage pipelined circuit consist of a randomly generated LUT. In parallel to the LUT, there is a matched delay circuit that is matched to the average delay of the LUT. Multiplexers and demultiplexers are used to control the path, on which data is propagated. The randomly generated lookup table realizes a one-to-one function for every possible 256 inputs and outputs. The multiplexer/demultiplexer pair is controlled by means of the PC[1] signals, which are generated by a logic decoder (not shown in the figure). Manufactured Mini-ASIC test chip is presented in Figure 3.6. Two versions of the LUT pipeline, one with (LUT Pipeline 2) and one without (LUT Pipeline 1) registers separating the combinational stages, were fabricated. In addition to the LUT pipelines, four asynchronous circuits, i.e., A1-A4, with completion detection were implemented. Asynchronous test blocks are explained in Chapter 8. Figure 3.7 shows the

[1] Path control.

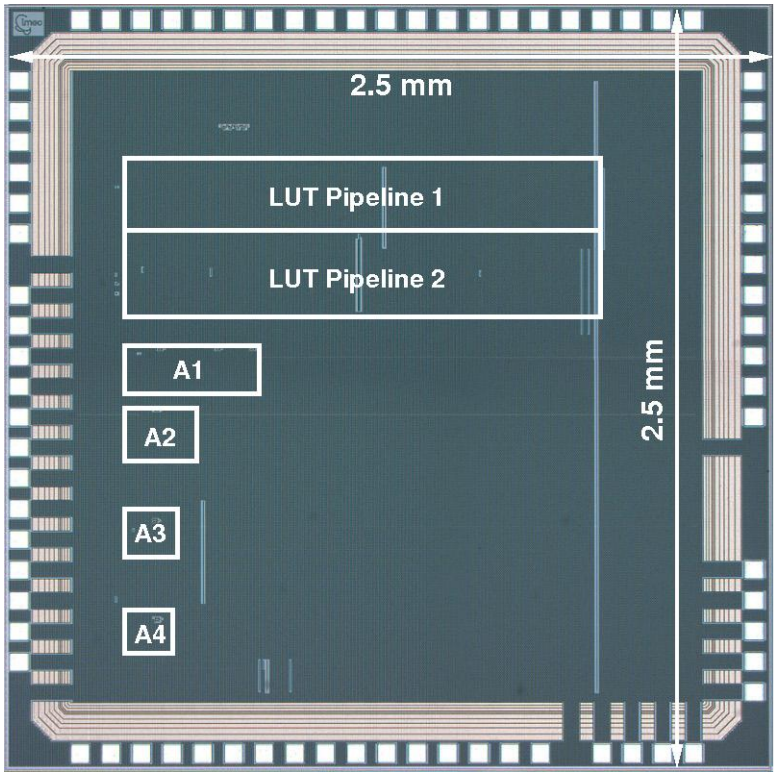


Figure 3.6: Micrograph of the fabricated 0.18 μm chip. Test blocks are emphasized. A1-A4 are asynchronous test blocks with completion detection.

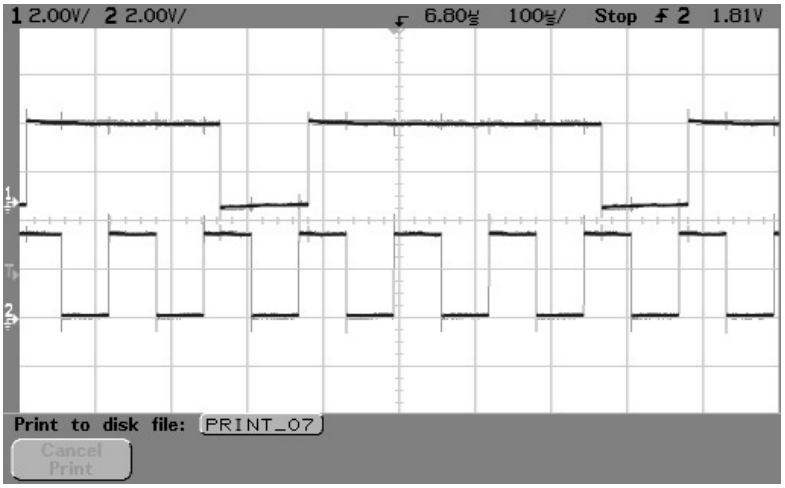


Figure 3.7: Oscilloscope screen capture of the signals from the test board. Top signal shows the most significant bit of the tested circuit and the bottom signal shows the applied clock signal.

oscilloscope screen capture of the most significant bit of LUT Pipeline 1 block with the applied clocking signal. The oscilloscope screen capture shows full-scale signals that are level-converted on the chip, i.e., signals with amplitude of 3.3 V.

We first checked the circuit for correct operation for varying supply voltage and operating frequency pairs for all the possible paths. 17 different path combinations are selectable for the test circuit. The data path through each stage may be selected for propagation through either the matched delay elements or the LUT. During the measurements 4000 random samples were applied to the circuit and the number of errors compared to the expected outputs were extracted. The results of the functional error measurements for the path number 16, i.e., all the data is propagated through the LUTs, are presented in Figure 3.8. During the measurements the frequency was swept from 2 kHz to 30 kHz in steps of 1 kHz and the supply voltage was swept between 0.2 V and 0.4 V in 10 mV steps. From the energy dissipation model, the operating frequency is defined as

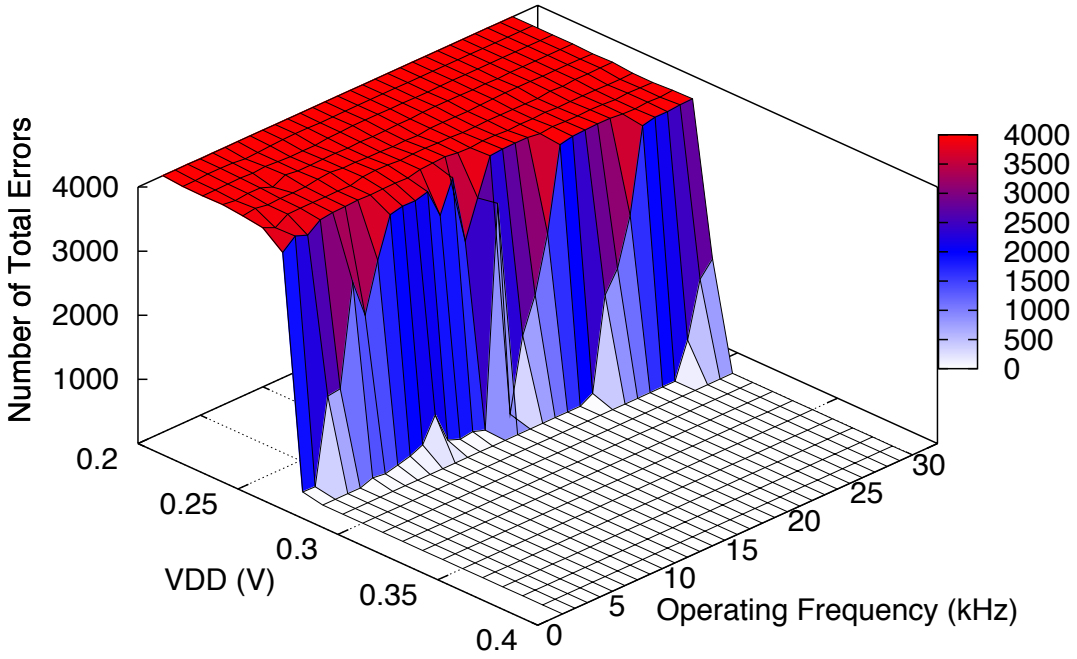
$$f_{op} = \frac{1}{k_{crit} T_{sw\_inv}}. \tag{3.17}$$

Equation (3.17) is over-plotted on the measurement results in Figure 3.8b. During our measurements we observed that for the all the paths tested, the memory elements stopped working below a certain voltage, i.e., 270 mV. Below this supply voltage, number of total errors quickly reached 4000, showing that the circuit is not operating properly for any operating frequency for these voltage values. This value is much higher than the expected reliable operating point for different processes that will be investigated in Chapter 6. Furthermore, the logic analyzer missed some triggering signals below 2 kHz operation during the measurements, so the measurement frequency range was limited to 2 kHz on the lower end not to have any errors in the measurement.

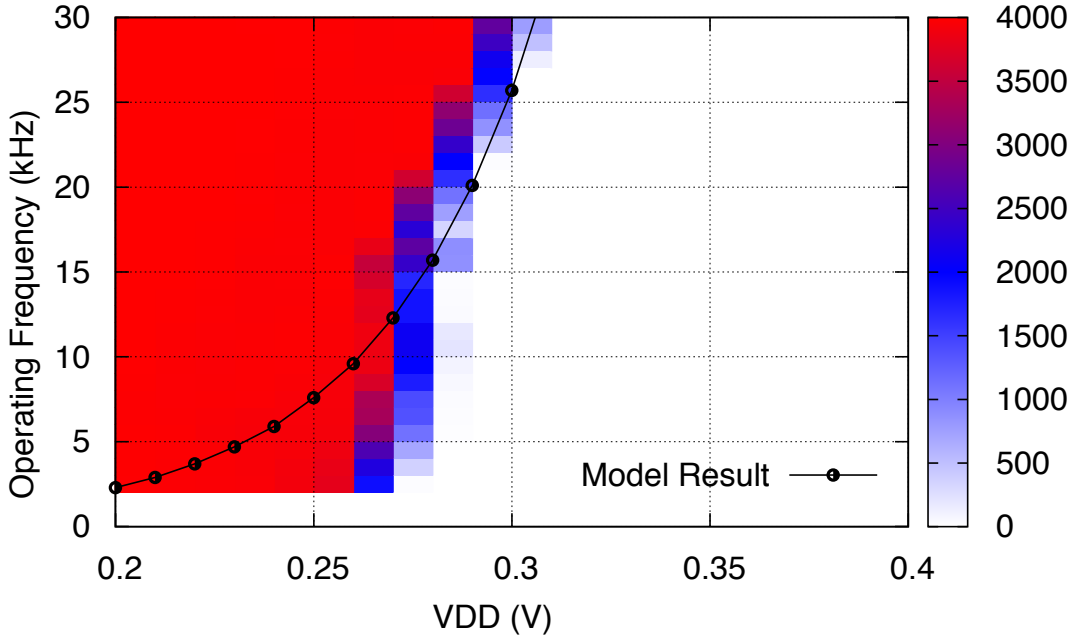
We measured the energy dissipation of the test circuit using a current integrator supplying the circuit that was measured. Same 4000 random samples were applied to the circuit during the energy measurements. Both in the beginning and in the end of the measurement cycle the voltage values at the output of the current integrator were measured. Using the measured data, we calculated the total energy based on the equation

$$E = CV_{DD}\Delta V, \tag{3.18}$$

where ΔV is the voltage difference measured at the output of the integrator. By dividing the value from equation (3.18) by the total number of samples, we found

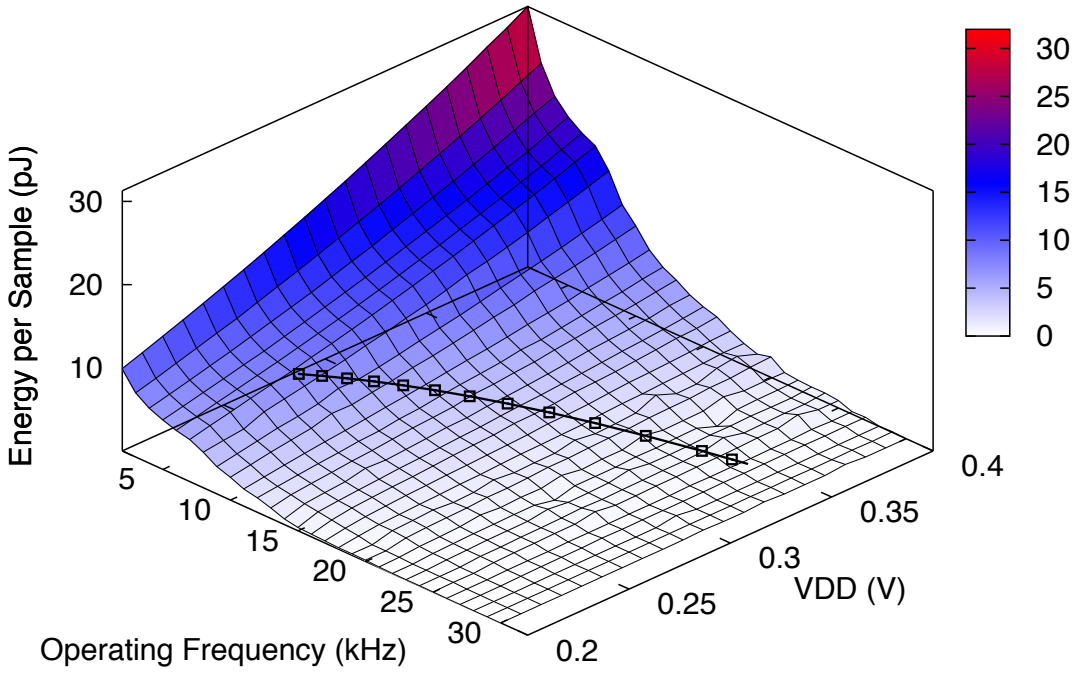


(a) Shmoo plot as a function of VDD and f.

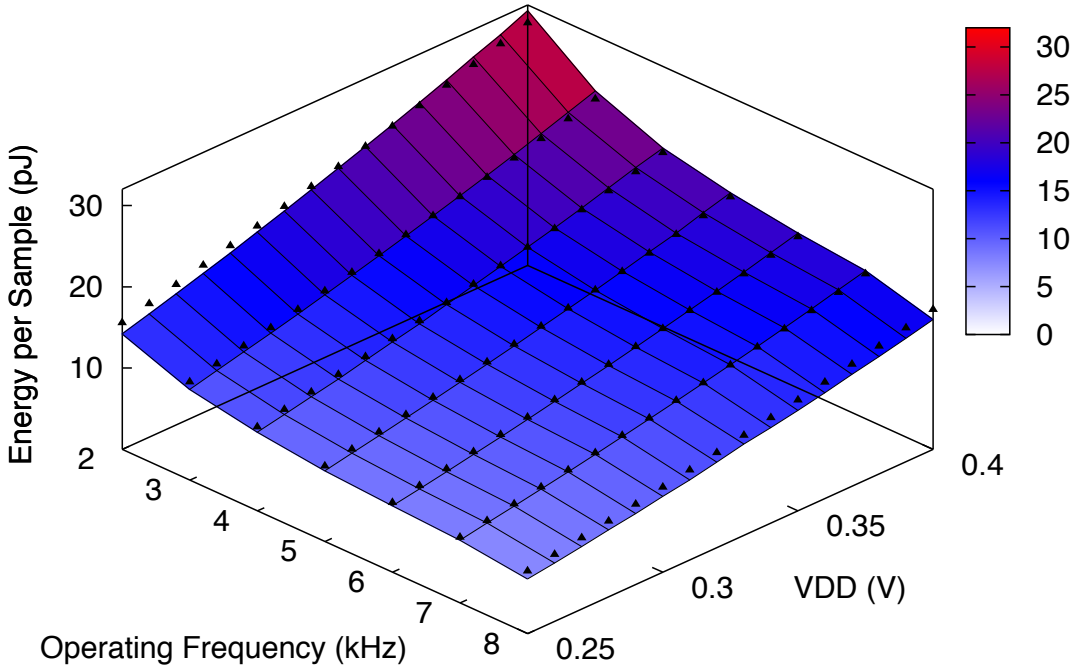


(b) Error measurement with the frequency calculation from the model plotted over.

Figure 3.8: Measured functional error for the case when all the data propagates through the LUTs. (a) shows the shmoo plot with number of total errors, and (b) shows the result with equation (3.17) plotted over for comparison.



(a) Measured energy for all the frequency-supply voltage pairs with equation (3.14) plotted over.



(b) Measured energy for a subset of the measurements points with the energy dissipation values from the sub-threshold energy model over-plotted.

Figure 3.9: Measured energy per sample for path number 16. (a) shows all the swept frequency-supply voltage pairs and (b) shows a restricted subset with the energy calculations from the model.

the energy dissipation per sample. The circuit was measured for all the paths and their respective energy dissipation figures. Measurement results for the path 16 is shown in Figure 3.9. Figure 3.9b shows a zoomed in version of the measurement with the calculated values from the energy model over-plotted. For this plot and the error calculation of the model with respect to the measurement results, we selected a subset of the measurement frequency-supply voltage space. The voltage range was constrained due to the fact that below 270 mV there was no way of knowing if the correct data was processed by the circuit. For the frequency range, for above 8 kHz operation, the measurement setup with the off-chip current integrator and voltmeters was too slow to correctly register the current that is consumed by the circuit, hence the limitation on the upper frequency range. The measured energy dissipation error with respect to the model changes between -4.79% and 3.83% for different data points.

3.5 Conclusions

In this chapter an energy dissipation model for sub-threshold operation is presented. The model is based on the circuit parameters that are obtainable from the standard cell synthesis results and switch level (synthesized Verilog) simulations. Same model is applicable to both synchronous and asynchronous operation with a slight modification. By using the proposed model, energy efficiency of the same circuit structure may be easily investigated in different manufacturing processes without the need for time and resource expensive SPICE level simulations.

Chapter 4

Energy Efficiency of Sub-threshold Circuits

This chapter presents a comparison of energy efficiency of sub-V_T synchronous and asynchronous circuits. The designs are analyzed with the sub-V_T energy dissipation model developed in Chapter 3. Energy dissipation reduction due to asynchronous operation is demonstrated on ISCAS85 benchmark circuits.

4.1 Comparison of Synchronous and Asynchronous Energy Efficiency

To be able to compare energy dissipation of a synchronous and an asynchronous circuit at their energy-minimum voltages (EMVs), V_{DD} in equations (3.12) and (3.14) are replaced by the optimum voltages $V_{opt-async}$ and $V_{opt-sync}$, respectively. The resulting energy equations depend only on the circuit implementation related k-parameters and the switching/delay properties of the circuit for a given manufacturing process.

A reference design, which has an energy dissipation equivalent to 1000 inverter gates with a drive capability of 1, and a critical path delay of 25 inverter gates, is used for preliminary energy efficiency comparison. Figure 4.1 shows the energy profile of our reference design at a switching/delay mean of 0.1 for both asynchronous and synchronous operation. In the comparisons made on this design, unless otherwise noted, the communication overhead parameter $k_{com_{oh}}$ is taken as 0.1, i.e., 10% communication overhead. The optimum operating voltages of the same circuit for synchronous

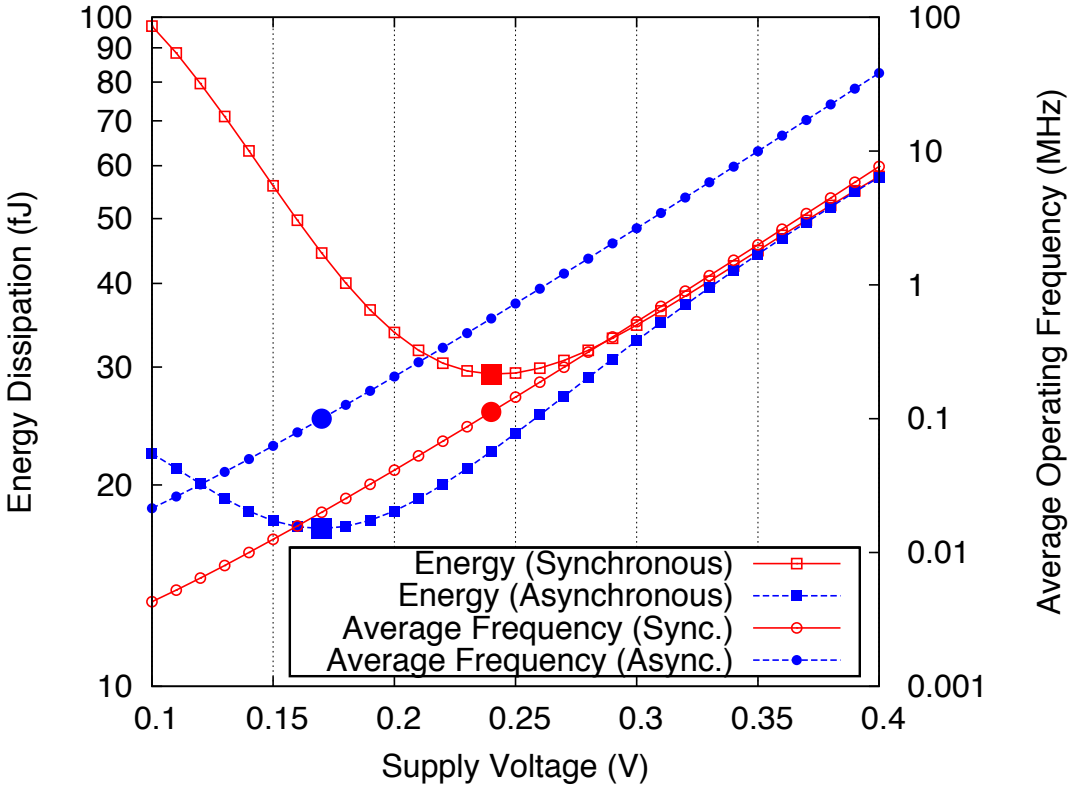


Figure 4.1: Total energy dissipation and average operation frequency for changing supply voltage values for a switching/delay mean value of 0.1. The energy-minimum operating points occur at different voltages for synchronous and asynchronous cases. EMVs and operating speed at EMVs are marked.

and asynchronous operations for the specified mean values are estimated to be at 240 mV and 170 mV, respectively.

When the EMVs for the asynchronous and synchronous cases, equations (3.13) and (3.15), respectively, are compared, it is seen that the equations differ by the $1/(k_{com_{oh}} + \mu_d)$ factor in the argument of the LambertW function. Therefore, as long as $k_{com_{oh}} + \mu_d$ value is less than 1, the EMV for the asynchronous operation will always be lower than the synchronous operation, resulting in lower energy operation. $k_{com_{oh}} + \mu_d$ can be higher than 1 because of i) high communication delay overhead and because of ii) majority of logic path delays being comparable to the critical path delay which increase μ_d to a value close to 1. Furthermore, for the cases where $k_{com_{oh}} + \mu_d$ is less than 1, an asynchronous circuit dissipates less leakage energy than the synchronous counterpart for the same operating voltage.

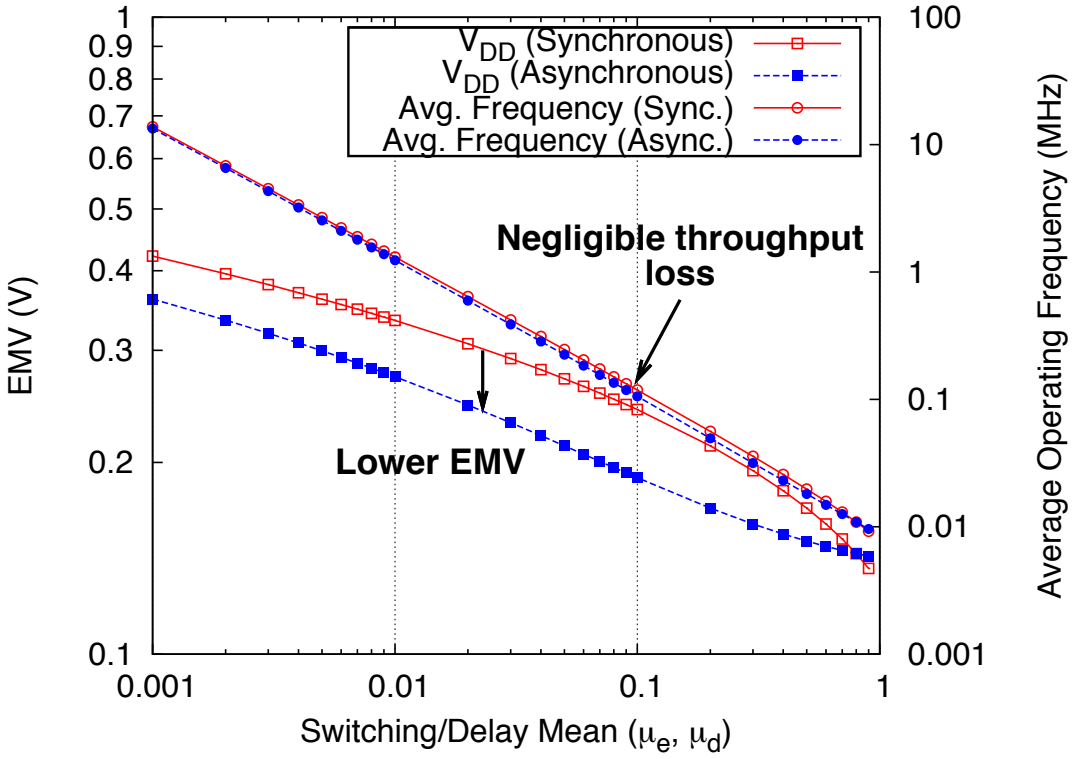


Figure 4.2: Energy-minimum operating voltage and the respective throughput change for varying switching/delay mean values. The throughput loss due to the lower energy-minimum operating voltage is negligible.

Advantages of operating in an asynchronous manner in the sub-threshold regime are twofold for energy dissipation reduction. First, the leakage energy is reduced by reducing the average time the circuit leaks. Due to the lower leakage energy dissipation, the energy-minimum operating voltage value of the circuit decreases and the circuit is operated at a lower energy-minimum operating voltage. This reduction in the supply voltage effectively reduces the switching energy dissipation. This is seen in the plot of EMVs and their respective throughput values in Figure 4.2. The EMVs and average operating frequencies for changing switching/delay properties of our reference design are shown. While working under the conditions defined by Assumptions 1, 2, and 3 in Chapter 3, the EMV of the asynchronous operation is lower, thus reducing the switching energy. The throughput worsens due to lower operating voltage but it is negligible in asynchronous operation because of better than worst case computation delays as shown in the figure.

4.2 Benchmark Circuit Energy Efficiency Comparisons

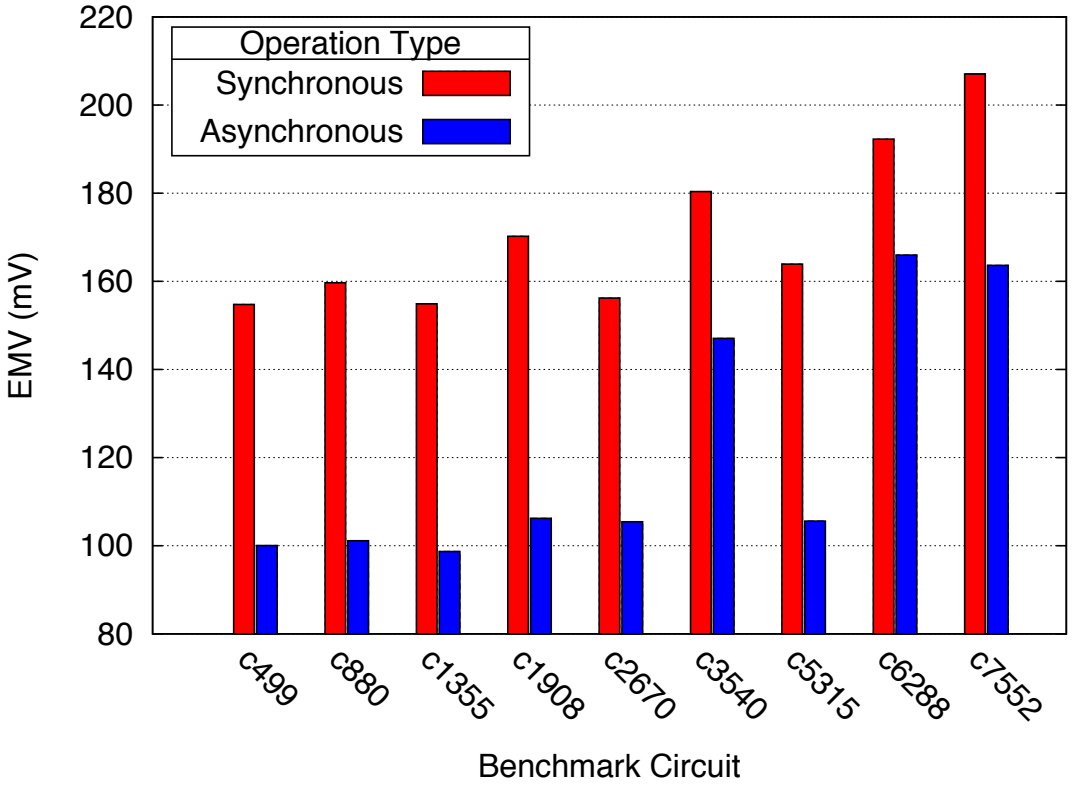
To investigate the effectiveness of asynchronous operation on real world applications for sub-threshold energy reduction, we extracted the model parameters from synthesized ISCAS85 benchmark circuits and their switch level simulations. Using the extracted parameters, we calculated the energy-minimum operating voltages and respective energy dissipation values. The values used in the calculations are given in Table 4.1. $k_{cap-logic}$, k_{leak} and k_{crit} were extracted from the synthesis results, and circuit delay and switching parameters μ_d and μ_e were extracted from switch level simulations. During the switch-level simulations randomly generated (Assumption-2) 10000 vectors were used.

The calculated EMV values for synchronous and asynchronous (under Assumption-1) operations are shown in Figure 4.3a. It is clearly shown in the plot that by employing asynchronous operation in the sub-threshold regime, the EMV is lowered. For example, for the benchmark circuit *c1908*, working asynchronously results in the reduction of EMV by 37.6%.

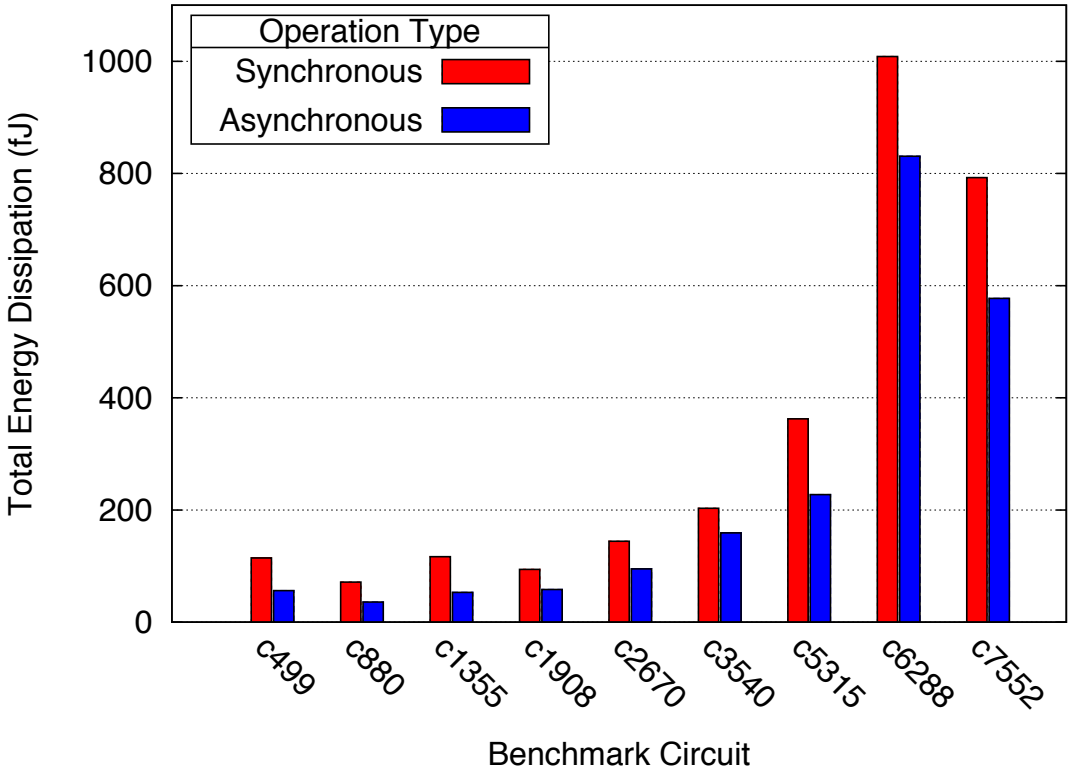
In Figure 4.3b energy dissipation of the benchmark circuits at their respective energy-minimum operating voltages for synchronous and asynchronous cases are shown. Operating the circuits asynchronously in the sub-threshold regime result in considerable energy dissipation savings. The decrease in the energy-minimum operating voltage and reduction in the total energy dissipation are given in the second and third columns of Table 4.2, respectively. By employing asynchronous operation in the sub-threshold regime, energy dissipation reduction up to 51% is observed.

Table 4.1: Parameter values extracted from the synthesis results for the ISCAS85 benchmark circuits.

Benchmark circuit	$k_{cap-logic}$	k_{leak}	k_{crit}	μ_d	μ_e
c499	559	437	57	0.31	0.7
c880	568	207	78	0.32	0.41
c1355	588	410	62	0.28	0.68
c1908	418	300	78	0.41	0.69
c2670	897	454	70	0.48	0.54
c3540	1228	470	120	0.58	0.47
c5315	2080	990	86	0.43	0.56
c6288	4310	1051	307	0.62	0.61
c7552	2498	1576	194	0.44	0.74

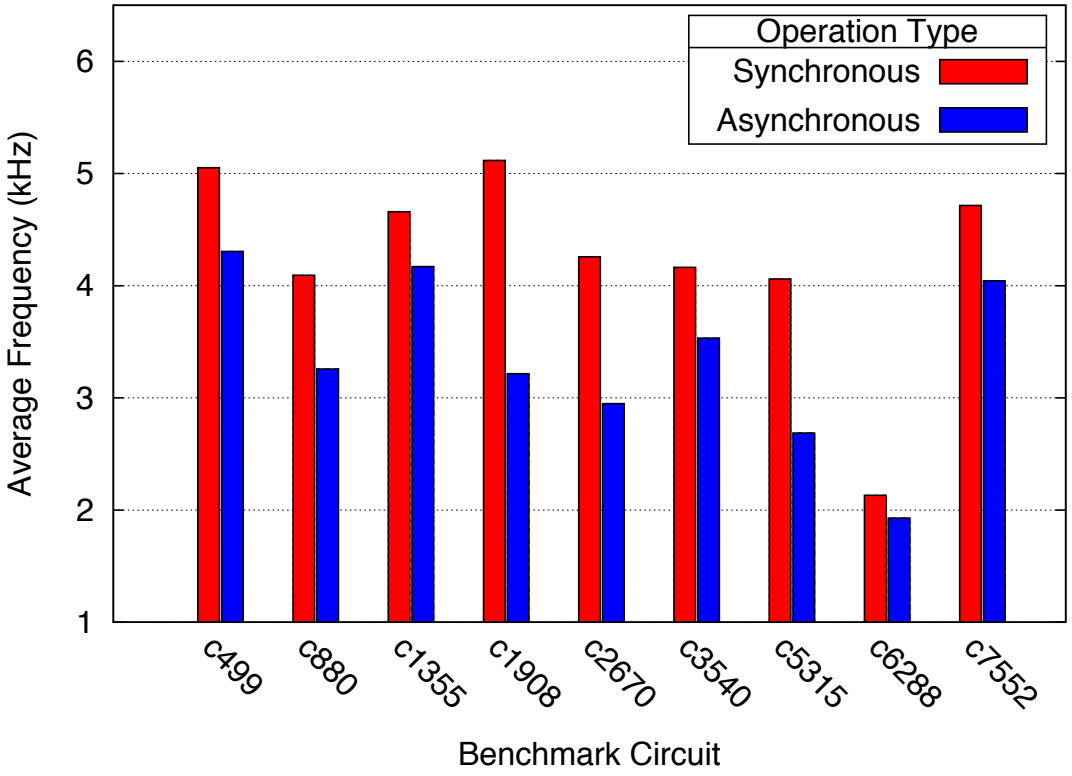


(a) Energy-minimum operating voltage.

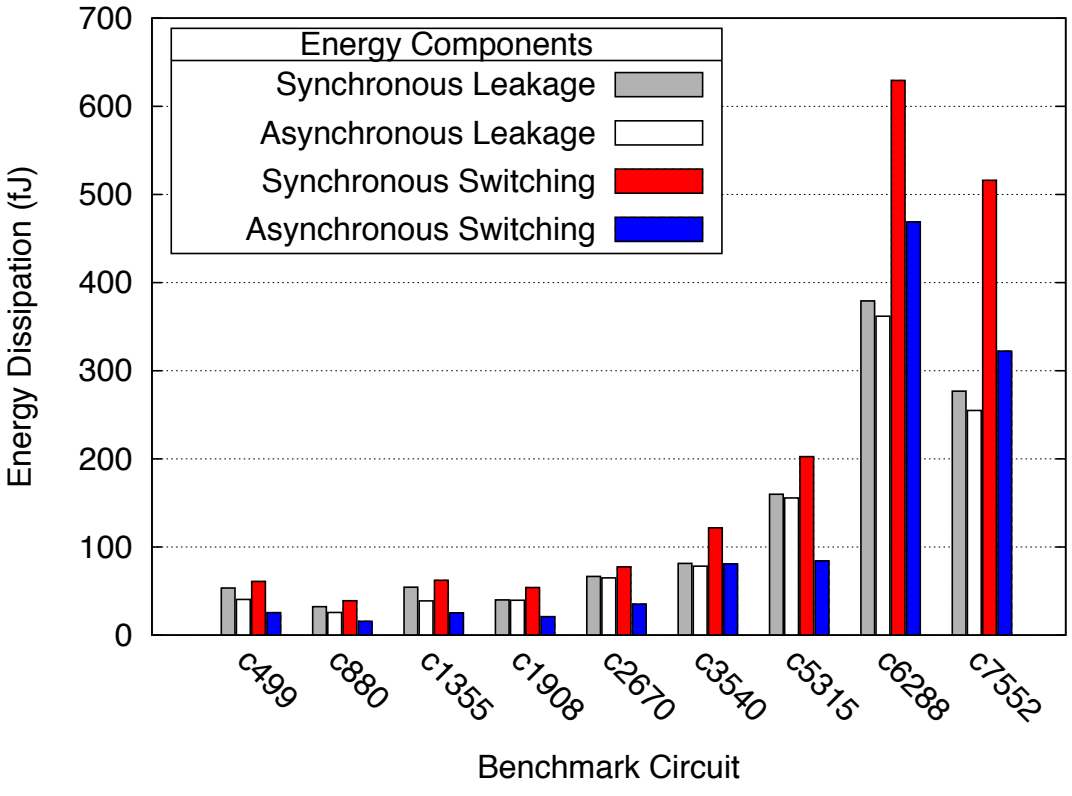


(b) Energy dissipation at the energy-minimum operating voltage value.

Figure 4.3: Energy-minimum operating voltages and relative energy dissipation of ISCAS85 benchmark circuits for both synchronous and asynchronous operation.



(a) Change in the average operation frequency.



(b) Energy dissipation components.

Figure 4.4: Detailed average operation frequency change and energy dissipation plots of the test circuits. The average operation frequency is lower due to lower EMV. Leakage and switching energy dissipation elements in the energy plot are separated for both synchronous and asynchronous operation in the energy plot.

Table 4.2: Change in EMV, energy dissipation, average throughput and energy delay product for the ISCAS85 benchmark circuits.

Benchmark circuit	Supply voltage reduction (%)	Energy dissipation reduction (%)	Average throughput loss (%)	EDP reduction (%)
c499	35.4	51.0	14.8	42.4
c880	36.7	49.6	20.4	36.6
c1355	36.3	54.6	10.5	49.3
c1908	37.6	38.0	37.1	1.4
c2670	32.5	34.2	30.8	5.0
c3540	18.5	21.7	15.1	7.7
c5315	35.6	37.3	33.8	5.2
c6288	13.7	17.6	9.6	8.9
c7552	21.	27.2	14.2	15.1

While the lower-voltage operation may reduce the average throughput of the design as shown in Figure 4.4a, if ultimate minimum energy operation is targeted, asynchronous operation in the sub-threshold domain is the most prominent option that allows the lowering of the EMV. Due to a lower EMV, both leakage and switching energy components of the designs are reduced, as shown in Figure 4.4b. For all the benchmark designs examined, which have different structures and switching/delay properties, the energy dissipation in asynchronous operation is lower than the synchronous operation. In all the benchmark designs, reduction in the switching energy dissipation due to lower EMV is more significant than leakage energy dissipation reduction. Although the average throughput is lower for the asynchronous case (Figure 4.4a), leakage energy, which is inversely proportional to the average throughput of the circuit and semi-linearly proportional to the supply voltage, is always lower for the asynchronous case.

Although our aim is to realize minimum energy operation, we also checked the efficiency of the energy-delay trade-off in the sub-threshold regime between the synchronous and asynchronous operations. To evaluate the energy-delay trade-off efficiency, we used the energy delay product (EDP) metric. The improvement in the EDP due to asynchronous operation is shown in the last column of Table 4.2. Although the average throughput is lower for all the benchmark designs, because of greater reduction in energy dissipation, the EDP of the asynchronous operation is always lower than the synchronous one. From these results, it is concluded that by operating asynchronously the operation speed is efficiently traded for lower energy

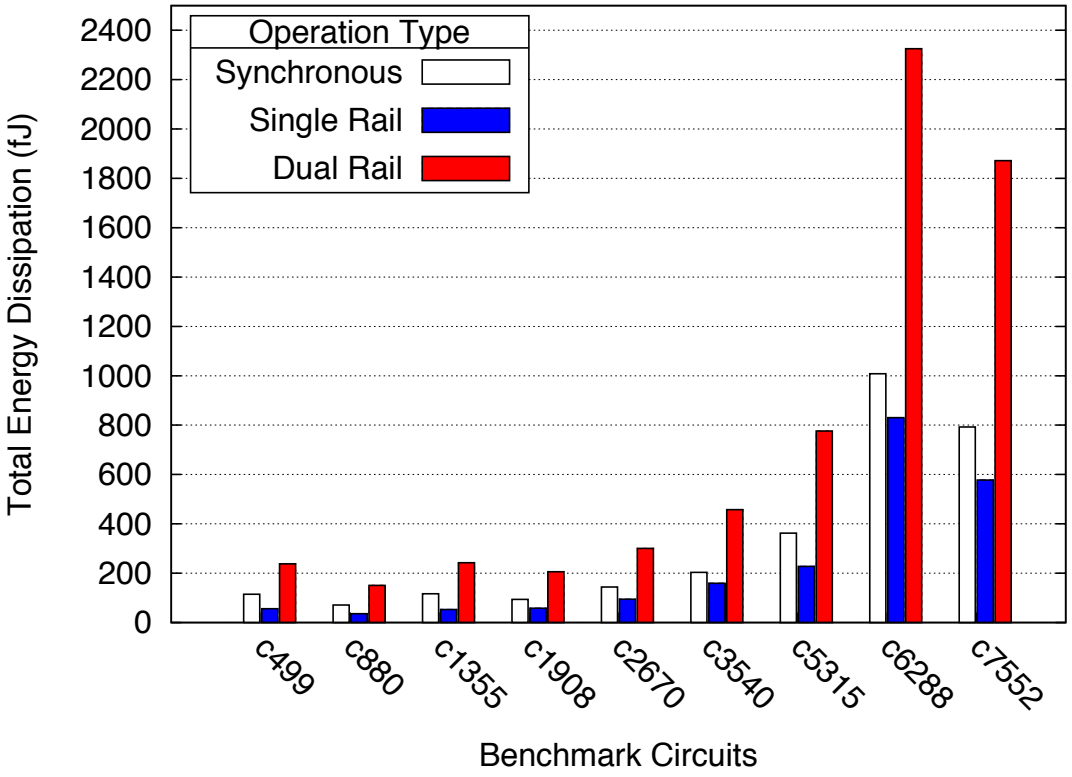


Figure 4.5: Energy dissipation comparison of ISCAS85 benchmark circuits for synchronous, single-rail and dual-rail implementations.

operation and for most of the cases significant reductions in energy dissipation and EDP are realized.

4.3 Implementation of Asynchronous Circuits with Digital Completion Detection

In the previous section advantages of running a circuit asynchronously, i.e., with average case performance was shown. In practice, implementing digital circuits with digital completion detection properties bring circuit overhead due to the inherent redundancy for generating the completion detection signal. Recently in [66] Lotze et al. examined self-timed asynchronous circuits operating in the sub-threshold regime. They implemented and simulated dual-rail multipliers with different bit-widths. For the 8-bit case the simulations showed an increase by a factor of 1.95 and 3.05 for the area and the switching energy, respectively.

To compare a purely digital self-timed implementation, i.e., dual-rail implementation, with the single rail implementation of the benchmark circuits with external completion detection mechanism, the k-factors are updated to reflect the changes in the circuit architecture. Based on the findings in [66], we multiplied k_{leak} by 1.95 and $k_{cap-logic}$ by 3.05. The results of our simulations showing the energy profile of ISCAS85 benchmark circuits for synchronous, single-rail and dual-rail implementations are shown in Figure 4.5. It is seen that for all the circuits, due to the increase in leakage and switching energies, dual-rail implementation has the highest energy dissipation. This confirms the result in [66] where it was suggested that only the critical path of the circuit should be designed using dual-rail techniques.

4.4 Conclusions

In this chapter, sub-V_T energy model introduced in Chapter 3 is used to compare synchronous and asynchronous operations for their energy efficiency. ISCAS85 benchmark circuits are used to represent real-world application examples. Random data sets are applied to ISCAS85 benchmark circuits for demonstrating achievable savings in energy dissipation by operating asynchronously. From the simulation results it is found that, by operating asynchronously, energy dissipation reductions of up to 51% and EDP reductions of up to 49.3% are realizable.

Chapter 5

Digital Event Detection for Cardiac Pacemakers

With the scaling of the integrated circuit manufacturing technologies, it is possible to integrate more devices per unit area on a single chip. This allows designers to implement more complex functionality on a single chip. However, in energy constrained applications, such as implantable biomedical devices, battery size, hence available energy is extremely limited and one of the main constraints. For this reason, in this thesis we are using a digital event detector for cardiac pacemakers as one of our benchmark designs. Due to implantable and battery operated nature of the circuit, enhancing energy efficiency is of primary concern. Furthermore, digital event detector is a complex circuit which enable us to use it as an example of today's typical digital systems.

5.1 Heart

The human heart is the part of the human cardiovascular system that is responsible for blood circulation (see Figure 5.1). It pumps blood through the cardiovascular system by means of rhythmic contractions and relaxations. The heart is divided into four chambers: Upper chambers are called left and right *atria* and the lower chambers are called *ventricles*. The two atria act as blood collectors for the blood returning from the body and the ventricles act as pumps to eject the blood to the body. Dexoygenated blood from the body enters the right atrium through the *superior vena cava*. After entering the atrium, blood passes to the right ventricle through tricuspid valve. Right ventricle pumps the deoxygenated blood through the pulmonary artery to the lungs.

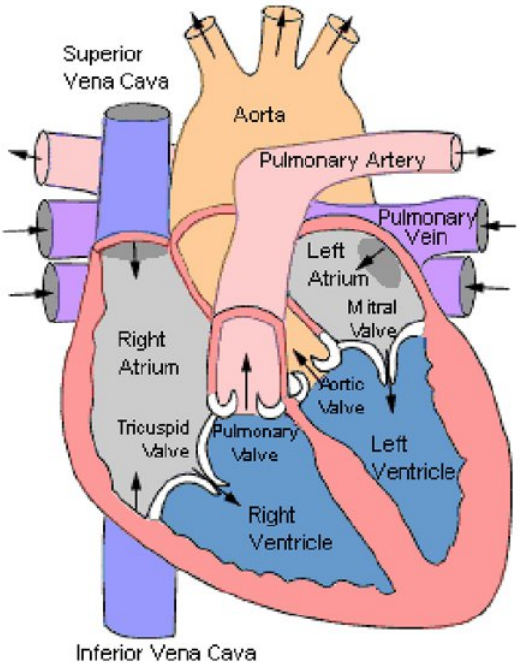


Figure 5.1: The heart (After [2]).

Oxygenated blood returning from the lungs enters the heart via the pulmonary veins. After being collected at the left atrium, blood is passed to the left ventricle through the mitral valve. From the left atrium blood is pumped back to the body through *aorta* [2, 5].

5.2 Cardiac Cycle

The cardiac events that occur from the beginning of one heartbeat until the next are called the *cardiac cycle*. The excitation and conduction system of the heart that is responsible for the control of the pumping action is presented in Figure 5.2. Each cardiac cycle is initiated by the generation of an electrical stimulus signal at the *sinoatrial* (SA) node. This generated signals passes through the *atrioventricular* node, with a small delay of about 0.1 seconds, to the ventricles. This short delay guarantees that both atria and ventricles are filled with blood before contraction. Once the signal enters the AV bundle, it spreads rapidly through the Purkinje fibers to the two ventricles resulting in a coordinated ventricular contraction [3].

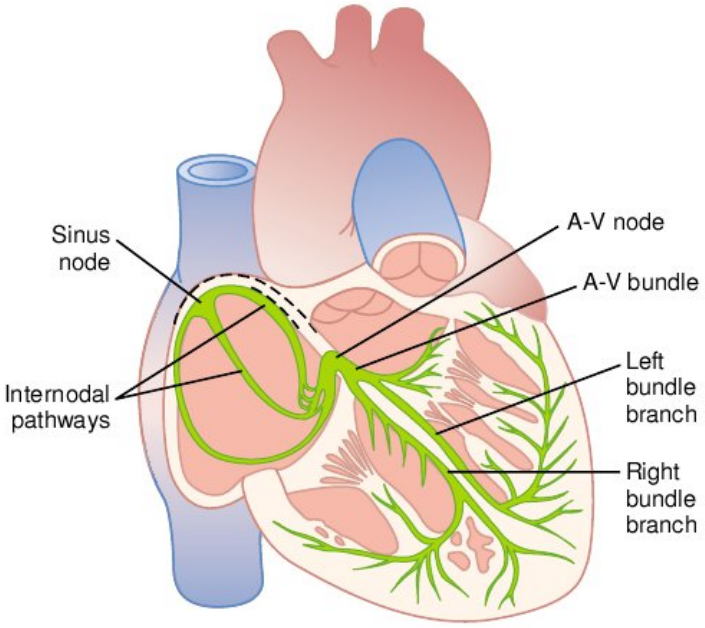


Figure 5.2: The cardiac excitation and conduction system (After [3]).

For the normal functioning of the heart, orderly sequence of activation of the cardiac muscle with regular timing is critical [2]. Several cells in the heart can generate electrical impulses. AV node, SA node and Purkinje fibers generate impulses with about 50 beats per minute, 70 to 80 beats per minute and 15 to 40 beats per minute, respectively. The normal rhythm of the heart, which is between 60 and 100 beats per minute, is controlled by the discharges from the SA node. SA node controls the rhythmicity of the heart due to the fact that the discharge rate of the SA node is considerably faster than that of AV node or Purkinje fibers. Each time the SA node discharges, this signal is propagated to the AV node and Purkinje fibers, resulting in their discharge.

5.3 Cardiac Signals

The electrocardiogram (ECG) is a technique of measuring the electrical activity of the heart on the body surface. Measuring the electrical activity of the heart on the surface of the body is possible due to the fact that when the cardiac impulse is generated in the heart, the signal spreads through the heart to the neighboring tissues reaching the body surface. Figure 5.3 shows a typical ECG signal. The ECG signal is composed

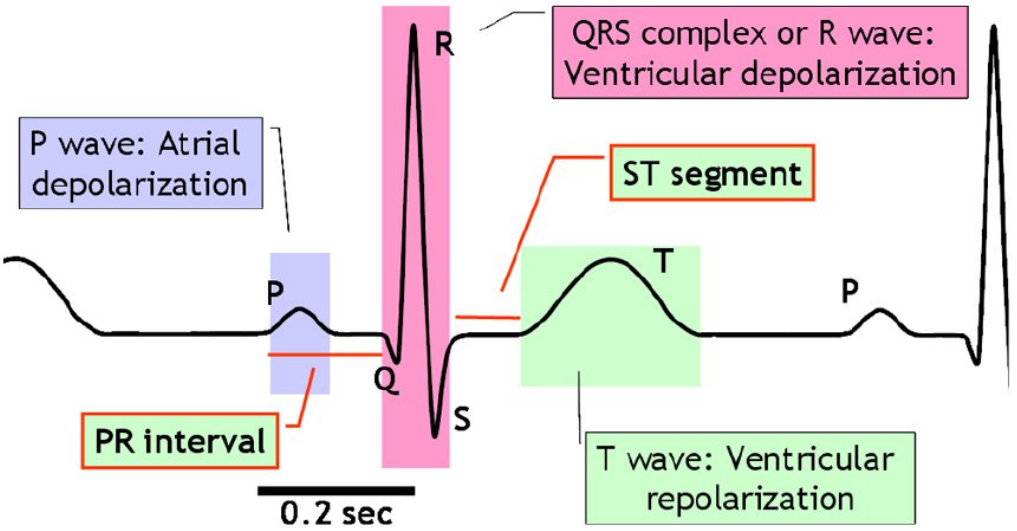


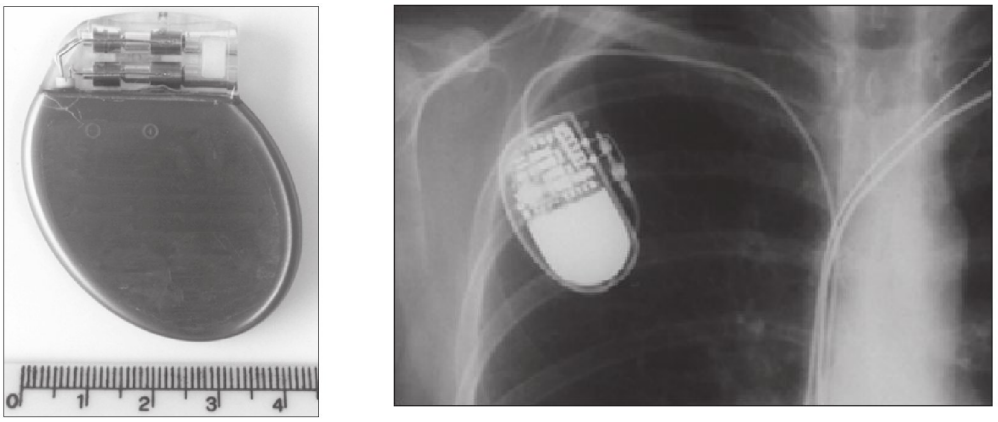
Figure 5.3: Electrocardiogram signal (After [2]).

of a P wave, a QRS complex which consist of Q, R, and S waves; and a T wave. The P wave is caused by the electrical depolarization of the atria and the QRS complex is caused by the depolarization of ventricles. The cause of lower-frequency T wave is the electrical potential generated by the ventricles when they are recovering from depolarization. During the depolarization of ventricles the blood is pumped out of heart. The ventricles remain contracted until after repolarization, i.e., until after the end of the T wave [3, 5].

5.4 Cardiac Pacemakers

A cardiac pacemaker is a device to regulate the heart beat by using electrical impulses. The electrical impulses are delivered to the heart through an electrode. The primary purpose of a pacemaker is to maintain an adequate heart rate. The first successful implementation of a cardiac pacemaker was presented in 1958 [67]. A pacemaker is shown in Figure 5.4a.

A pacemaker operates with a battery and utilizes the energy from the battery to stimulate the heart. The stimulation of the heart occurs by generating pulses and transferring those pulses to the heart through an implantable electrode catheter called *lead*. The purpose of the lead connected between the heart and the pacemaker are two-fold: i) The lead delivers the impulses from the heart to the pacemaker sensors,



(a) An implantable pacemaker

(b) Pacemaker implanted in human body

Figure 5.4: Implantable pacemaker: (a) out of body and (b) in body. (After [4]).

and ii) the lead delivers the pulse generated by the implantable pacemaker to the heart (see Figure 5.4b).

5.5 Cardiac Signal Analysis and Event Detection

In cardiac signal analysis the signal is first filtered with respect to 50/60 Hz noise. Afterwards, the QRS complexes are identified according to specific pre-defined detection rules. The detection is based on testing two or more hypotheses against each other, e.g., if a QRS complex is present or not. The more likely hypothesis is chosen as the outcome of the detection. Statistical information on the signal and noise properties are used for setting the thresholds and in making decisions [68].

Two measures of the performance of a detector are *probability of a missed detection* (P_M) and *probability of a false alarm* (P_F). P_M is defined as

$$P_M = \frac{N_M}{N_T + N_M}, \tag{5.1}$$

where N_M is the number of missed events and N_T is the number of true events. Likewise, probability of a false alarm is defined as

$$P_F = \frac{N_F}{N_F + N_T}, \tag{5.2}$$

where N_F is the number of false events. In this thesis a digital implementation of an R-wave cardiac event detector is used as one of the reference designs to demonstrate and evaluate sub-threshold operation. The event detector is based on the concept of wavelet filtering, which is explained next.

5.6 Wavelet Decomposition

In this section background information about wavelet decomposition is provided. First, Fourier transform is introduced to point out its shortcomings for non-stationary signals. Afterwards, basics of the Wavelet Transform are presented.

5.6.1 The Fourier Transform

The Fourier Transform (FT) is an integral transform that re-expresses a function in terms of sinusoidal basis functions. The final result is expressed as a sum of sines and cosines functions multiplied by amplitude coefficients. For example, in signal processing applications, a time domain signal is usually transformed into a frequency domain signal [69]. The Fourier coefficients are obtained by correlating the input signal $f(t)$ with a sinusoidal wave e^{jwt}:

$$f(w) = \int_{-\infty}^{\infty} f(t)e^{-jwt}dt. \tag{5.3}$$

Equation (5.3) measures the amplitude of the signal $f(t)$ at frequency w. This is possible because FT of e^{jw_0t} is a Dirac function, i.e., $2\pi\delta(w - w_0)$, in the frequency domain.

One of the disadvantages of FT is that it only provides information about the frequency content of the signal; that is, timing information of the signal is lost after transformation. This is illustrated in Figure 5.5. Figure 5.5 (a_1) and (b_1) show a stationary and non-stationary signal, respectively. Stationary signal contains frequencies 10, 25, 50, and 100 Hz at any time instant; however, in the non-stationary signal these frequencies occur at different time intervals. Although time domain signals are different, their frequency spectra shown in Figure 5.5 (a_2) and (b_2) provide similar information about the frequency distribution of the signal.

One solution to retain the time localization of a signal is to apply windowing to decompose the signal $f(t)$ into waveforms that are well localized in time [69]. However, using a windowed part of the waveform reduces the frequency resolution

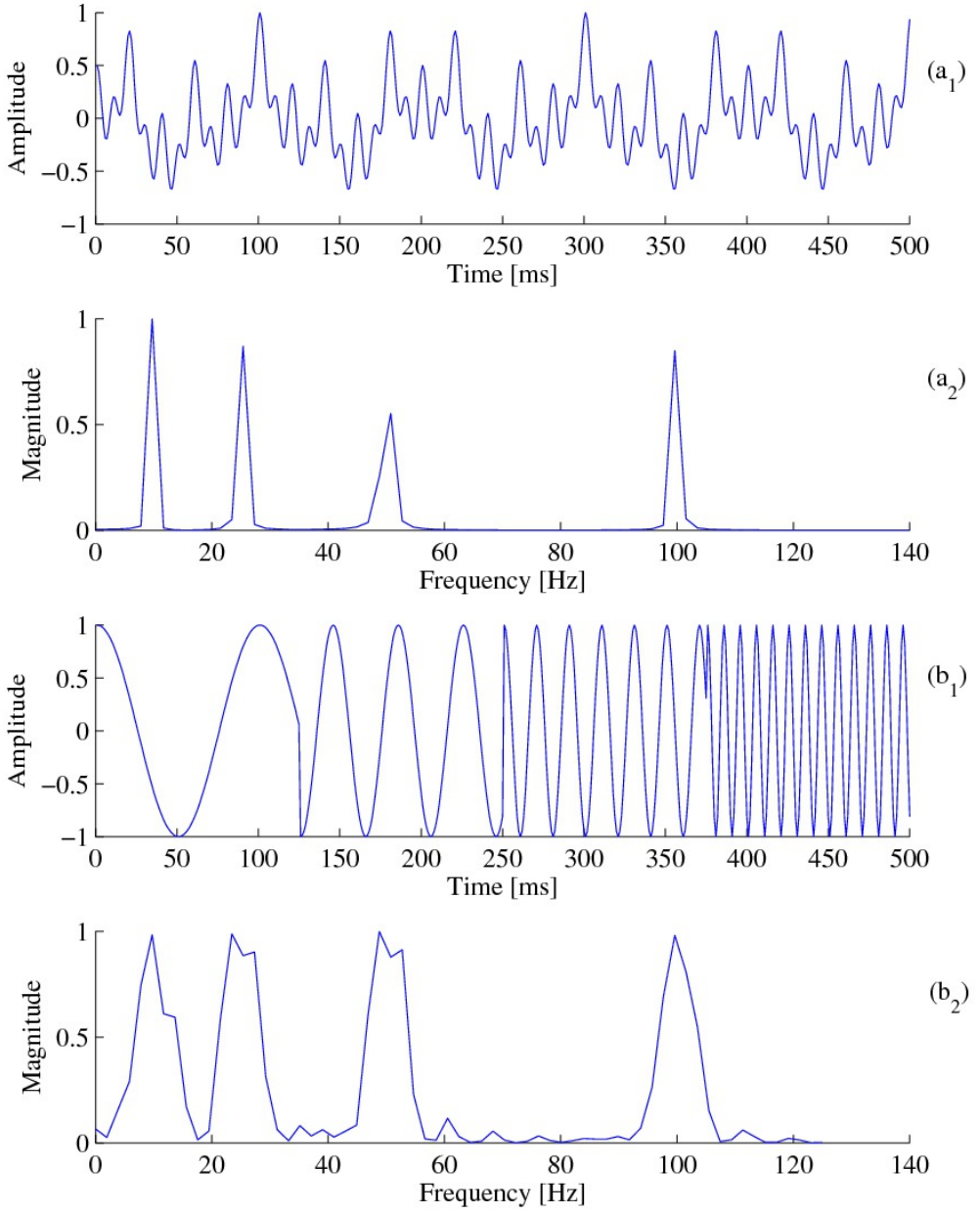


Figure 5.5: The Fourier transform of a stationary (a_1) and a non-stationary (b_1) signal. (a_2) and (b_2) shows the FT of the signals (After [5]).

as the number of samples are limited. The Wavelet transform solves this problem by using basis functions that are well concentrated in both time and frequency.

5.6.2 The Wavelet Transform

The Wavelet Transform (WT) decomposes the analyzed signal into its components as in the Fourier transform. Wavelet decomposition uses wavelets as the basis functions. Wavelet functions may be finite and irregular in shape unlike the basis functions of FT. Furthermore, unlike FT, the WT provides information about both the time and frequency of the analyzed signal.

For the WT, a base wavelet, called the mother wavelet, is chosen. Using this base function, the analyzed signal is decomposed into components appearing at different scales. The continuous wavelet transform (CWT) of a function $f(t)$ is given by

$$W_f(\tau, a) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(t)\psi^* \left(\frac{t-\tau}{a} \right) dt, \tag{5.4}$$

where a is a positive number that defines the scale, τ is any real number that defines the shift, $\psi(t)$ is the mother wavelet and $*$ is the complex conjugate. Scaling in the WT takes place by representing low frequencies by coarser scales, and higher frequencies by finer scales [2,5].

The CWT is highly redundant, thus, the scaling and shifting parameters may be discretized according to a scaling grid. The most popular discretization approach is *dyadic scaling*,

$$a = 2^{-j}, \tau = k2^{-j}, \tag{5.5}$$

where j and k are integers. Introducing equation (5.5) into (5.4), the discrete wavelet transform is obtained as

$$W_f(j, k) = 2^{j/2} \int_{-\infty}^{\infty} f(t)\psi^* \left(2^j t - k\right) dt. \tag{5.6}$$

Wavelet transformation of the previously shown non-stationary signal is presented in Figure 5.6. It can be seen that both the frequency and timing information of the analyzed signal is present. In our implementation of the cardiac event detector, discrete wavelet transformation is used for QRS complex, more specifically R-wave detection.

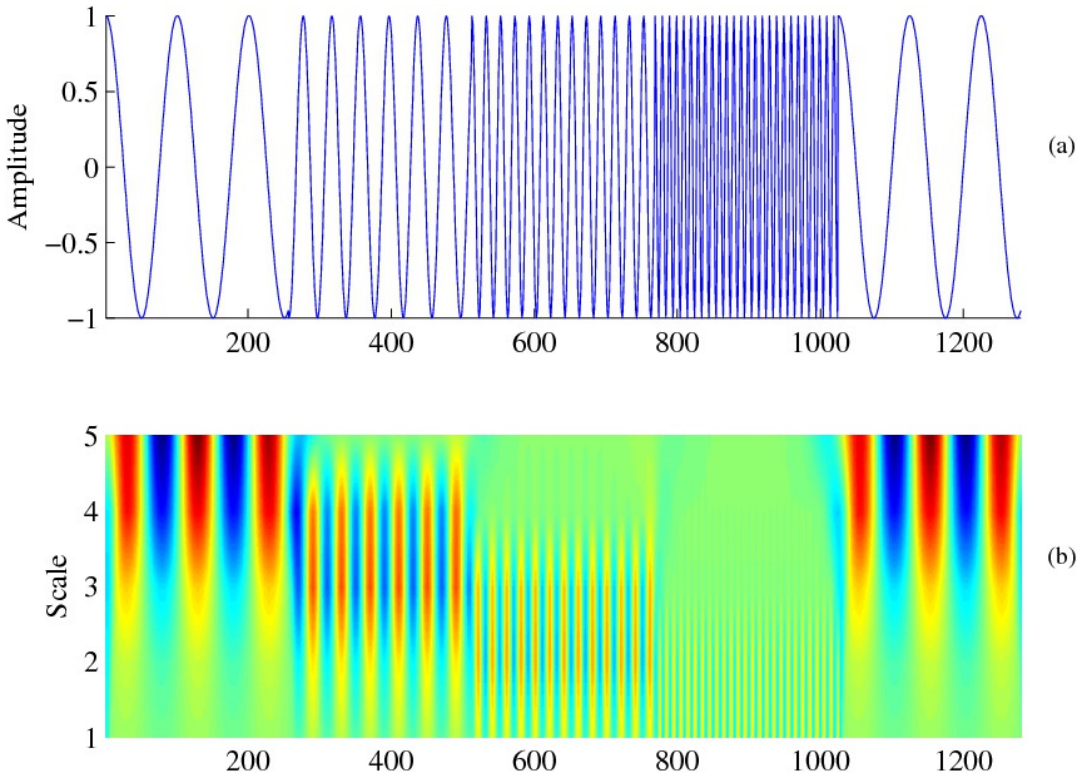


Figure 5.6: A non-stationary (a) signal and (b) magnitude of its discrete wavelet transform. (After [5]).

5.7 Digital Event Detector for Cardiac Pacemakers

This section presents the architecture of a 3-scaled wavelet filterbank, supplemented with a generalized likelihood ratio test (GLRT). Furthermore, noise suppression efficiency of the implementation is discussed. This implementation was realized at Lund University [5] and is used as one of the benchmark designs throughout the thesis [6].

5.7.1 Implementation of the R-wave detector

To achieve a power-efficient hardware mapping, short filters with integer values are chosen, i.e., first order difference, and the impulse response was chosen as a third order binomial function. A more detailed description of the wavelet filterbank and the GLRT is found in [70]. The implemented wavelet filterbank consist of three branches, $q = 2, 3, 4$, that scale and filter the signal $x(n)$ from the analog-to-digital converter,

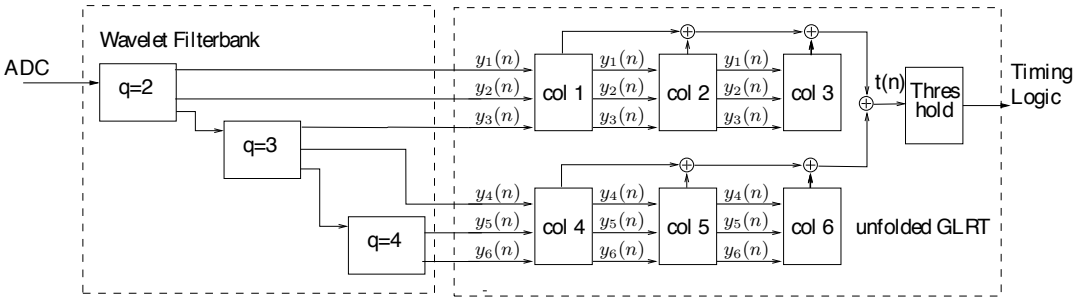


Figure 5.7: Parallel architecture of the wavelet filterbank and GLRT (After [6]).

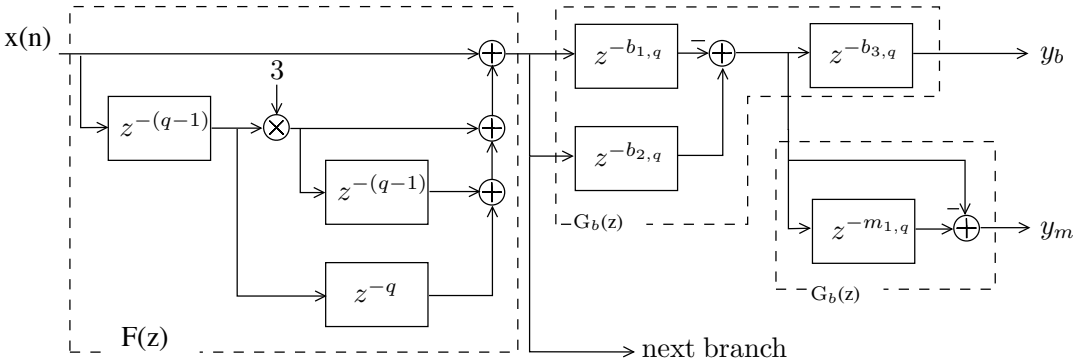


Figure 5.8: Data flow diagram of the first wavelet filterbank branch using Mallat's algorithm, ($q = 2$) (After [6]).

see Figure 5.7 and 5.8. The first biphasic branch realizes a straight-forward implementation as

$$F(z) = 1 + 3z^{-(q-1)} + 3z^{-(2q-2)} + z^{-(2q-1)}$$

and

$$G_b(z) = -1 + z^{-q}.$$

Reusing $G_b(z)$ implements the monophasic filterbank using a single branch for one scale factor and realizes the output of the filterbank. However, in order to center the functions to the longest propagation delay in the third branch, it is necessary to introduce additional delays in $G_b(z)$, see Figure 5.8. The impulse responses of the filterbank are presented in Figure 5.9. It can be observed that the wavelet-based structure offers a high flexibility for various cardiac morphologies.

The decision signal $T(n)$ is computed by the GLRT as

$$T(n) = \mathbf{x}^T(n)\mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{x}(n). \tag{5.7}$$

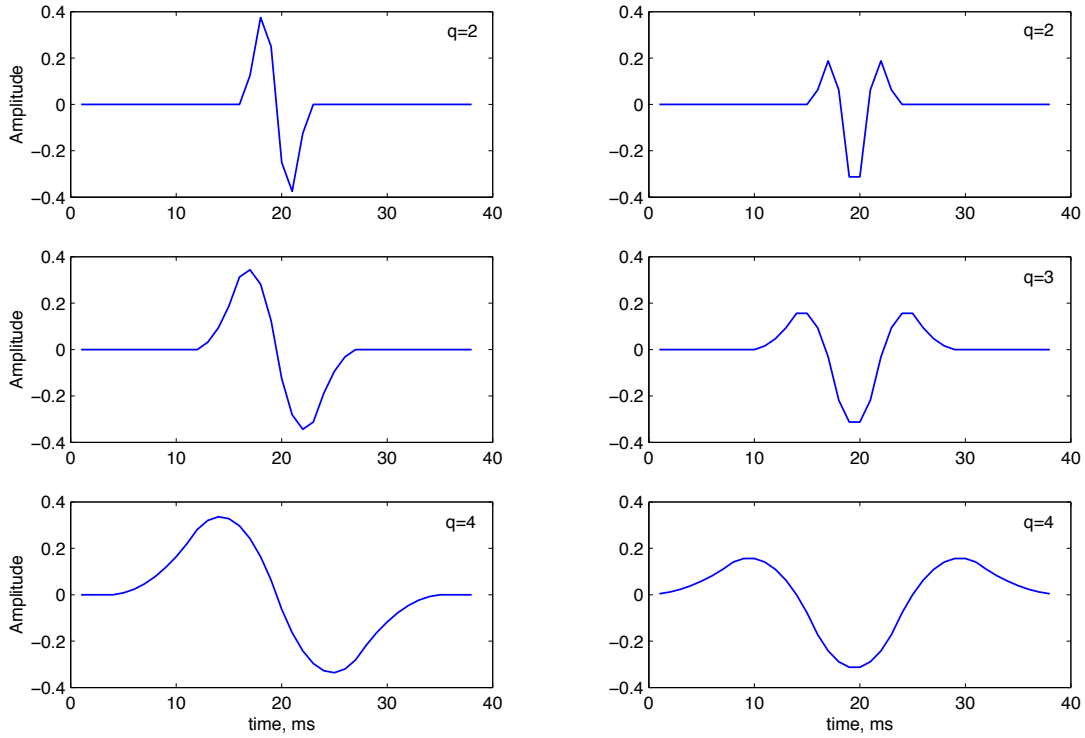


Figure 5.9: Impulse responses of the wavelet filterbank. The biphasic impulse responses $y_{b,k}(n)$ for $q = 2, 3, 4$ are displayed in the left panel and the monophasic impulse responses $y_{m,k}(n)$ in the right panel.

Since $\mathbf{x}^T(n)\mathbf{H} = \mathbf{H}^T\mathbf{x}(n)$, the remaining part of equation (5.7) to be implemented is the multiplication by $(\mathbf{H}^T\mathbf{H})^{-1}$, a matrix which is symmetric and sparse with half of its elements equal to zero,

$$(\mathbf{H}^T\mathbf{H})^{-1} = \begin{bmatrix} 4.3 & -2.8 & 0.7 & 0 & 0 & 0 \\ -2.8 & 4.5 & -1.8 & 0 & 0 & 0 \\ 0.7 & -1.8 & 1.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.8 & -2.3 & 0.6 \\ 0 & 0 & 0 & -2.3 & 4.2 & -1.4 \\ 0 & 0 & 0 & 0.6 & -1.4 & 1.7 \end{bmatrix}. \tag{5.8}$$

The multiplication of $\mathbf{y}(n)$ with the first column of $(\mathbf{H}^T\mathbf{H})^{-1}$ and the first element of $\mathbf{H}^T\mathbf{x}(n)$ is carried out as depicted in Figure 5.10, where $c_{i,i}$ are elements of $(\mathbf{H}^T\mathbf{H})^{-1}$ and $y_{3k+j}(n)$ the output of the filterbank. The schematic in Figure 5.10 represents the block referred to as *col i* in Figure 5.7, which needs to be replicated six times to realize the multiplication with the columns of the matrix $(H^TH)^{-1}$ in equation (5.8). To simplify the implementation, the matrix coefficients $c_{i,i} \cdots c_{i,i+2}$ are replaced with

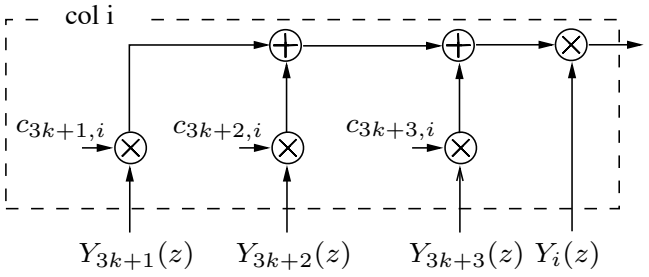


Figure 5.10: Data flow diagram of a unfolded block in the GLRT.

rounded integer values, which did not degrade performance. Thus, the multiplications are realized by *shift-add* instructions. Hence, the unfolded realization of the GLRT requires six generic multipliers and 17 adders.

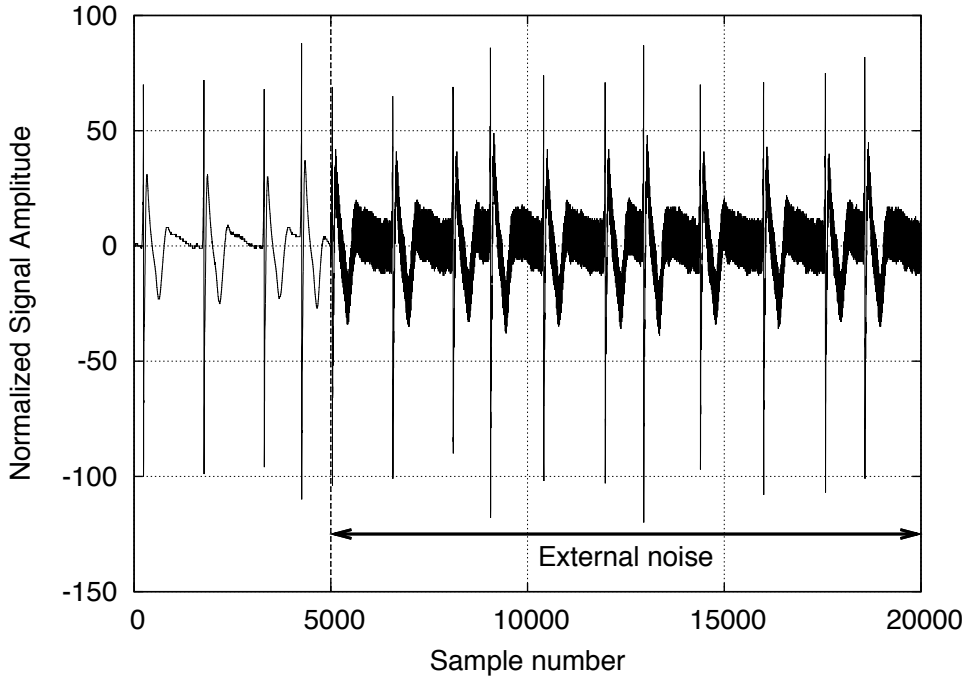
5.7.2 Hardware optimization

The architecture is optimized by register minimization, numerical strength reduction, and internal word-length optimization. The outputs of the filter are bit-optimized by tracing the maximum required word-length for full precision, when processing real-life data from an electrogram (EGM) database. Thus, full dynamic range is sustained by minimizing the internal word-length to 10 bits, which, in turn, results in narrower adders and multipliers in the following GLRT [71, 72]. The optimized design consists of 8750 NAND2 equivalent gates and has a logic depth of 143 fan-out-of-four inverter delays [6].

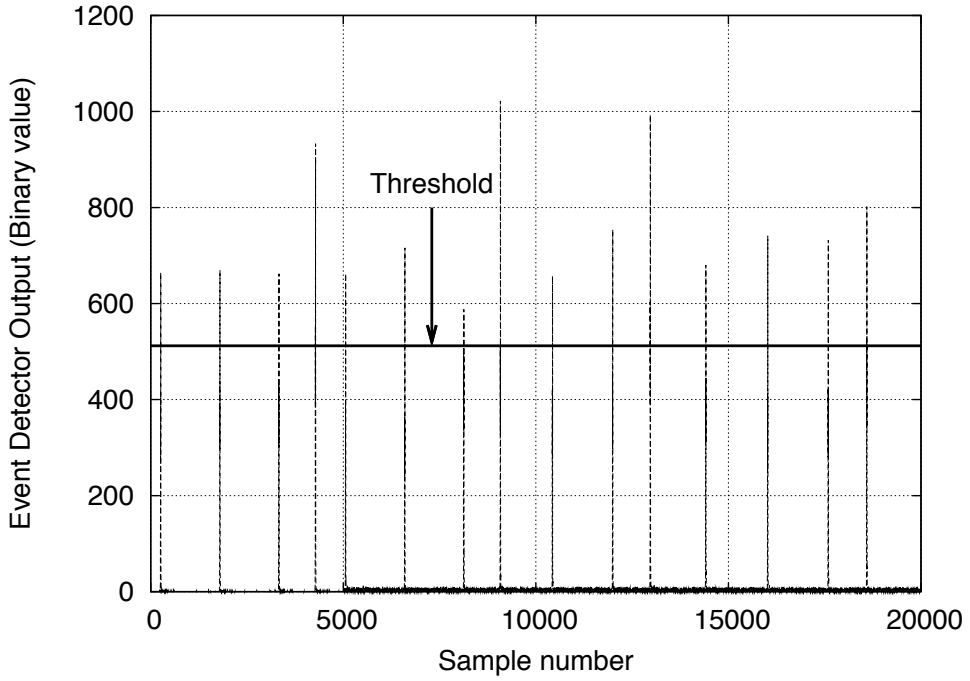
5.7.3 Detector Performance

The detector implemented in this study qualifies for pacemaker applications with reliable detection performance in noisy environments, validated on cardiograms recorded and digitized during pacemaker implantation [73]. Detection performance is measured by computing the probability of true detections (PD) and false alarms (PFA). A true detection is defined as an event occurring within 50 ms of the annotation, whereas events outside this interval are declared as false alarms. All signals in the EGM database (3200 events), are fed to the detector, and the detected events are classified as PD and PFA. It is found that the detector has a PD of 0.997 and PFA of < 0.001, which is rated as reliable performance. A typical input signal before the wavelet filterbank and the decision signal of the GLRT are presented in Figure 5.11. The upper pane shows a typical signal taken from the EGM database and fed to the circuit. To

simulate a worst-case noise scenario, interference is abruptly added to the signal at t_n, which results in an exaggeratedly high disturbance level. The output of the GLRT in the lower pane shows the effectiveness of the noise suppression performed by the filter. The highest peaks in the signal indicate cardiac activity, whereas the remaining waves are treated as noise. The SNR improves dramatically, and cardiac activity is detected by a programmable and dynamically updated threshold level.



(a) Pre-recorded cardiogram



(b) Output signal of the hypothesis test

Figure 5.11: Upper Pane: Digitized pre-recorded cardiogram as subjected to the event detector. Lower Pane: Output signal of the hypothesis test that is subject of the threshold function.

Chapter 6

Process Selection for Energy Minimization

Migrating the digital circuit designs to a smaller feature sized technology enable the designers to lower the energy dissipation. It was suggested in [35] that migrating the design from 250 nm node to 90 nm node for the same performance operation results in 61% energy dissipation reduction. In this analysis the effects of variability in the sub-threshold regime was not investigated. In [74] optimal process selection for different applications in the sub-threshold regime was investigated. Five industrial processes with feature sizes ranging between 250 and 65 nm were explored. However, foundry supplied, free process options were not considered in the analysis. In this chapter, contrary to the results presented in [74], we show that by employing correct process options based on the required application performance, migrating to a smaller feature sized technology is beneficial due to the reduced capacitance.

In our analysis we compared different CMOS technologies using foundry supplied models for our operating frequencies of interest, i.e., 1 and 32 kHz, frequencies widely used in biomedical applications. We investigated 180, 130, 90 and 65 nm process technologies with both standard, low-leakage (LL) and high-V_T (HVT) options whenever available. In our analysis no special techniques for leakage reduction such as power gating or reverse body-bias are considered.

6.1 Examined Process Properties

For our analysis of optimal process selection for sub-threshold energy minimum operation, we examined processes with feature sizes between 180 and 65 nm. In our analysis

Table 6.1: Extracted process parameters for process comparison. All the data presented was extracted by SPICE simulations using the foundry supplied model data.

Process	n_n	n_p	$I_{0,n}$ (pA)	$I_{0,p}$ (pA)	C_{inv} (fF)	C_{int} (fF)
180nm	1.291	1.532	3.545	4.844	5.9	4.5
130nm	1.416	1.518	20.297	32.621	2.14	1.84
130nm LL	1.467	1.545	0.923	3.098	2.14	1.84
90nm	1.273	1.414	909.949	1840.75	2.66	1.1
90nm LL	1.392	1.369	5.209	9.588	2.66	1.1
65nm LL-SVT	1.477	1.405	13.951	4.251	0.89	0.44
65nm LL-HVT	1.439	1.534	0.835	1.243	0.89	0.44

we also investigated the effects of process options targeting low-power/low-energy applications on the energy profile of digital circuits. As our benchmark design, we used the cardiac digital event detector for pacemaker applications presented in Chapter 5.

For the application of the energy model presented in Chapter 3 with different manufacturing processes, multiple parameters need to be extracted. As explained in Section 3.3, saturation off current I_0, sub-threshold slope n, and C_{int} needs to be extracted from SPICE simulations. To extract these parameters, we implemented testbenches for all the processes examined and fitted the results to the current equations of NMOS and PMOS transistors. The results of data extraction for the examined processes are presented in Table 6.1. When only standard process options are compared, increase in the NMOS and PMOS saturation off currents, i.e., $I_{0,n}$ and $I_{0,p}$, respectively, are observed. Furthermore, both gate (C_{inv}) and internal (C_{int}) capacitances scale with the minimum feature size. One important thing that is easily noticeable in Table 6.1 is the very high leakage current for both PMOS and NMOS transistor in the 90 nm regular process. According to the extracted data, migrating from the 130 nm node to the 90 nm node results in leakage current increase by a factor of 45 and 56 for NMOS and PMOS transistors, respectively. Furthermore, it is obvious that during the process development, 90 nm node was not optimized for low-power/low-energy applications. Contrary to what is expected, the low-leakage (LL) process option for the 90 nm node leaks more than the LL standard V_T (SVT) option of the 65 nm node.

6.2 Model Validation

The energy model presented in Chapter 3 is validated against SPICE simulations using ISCAS85 benchmark circuits. Simulations are run for 1000 cycles with ran-

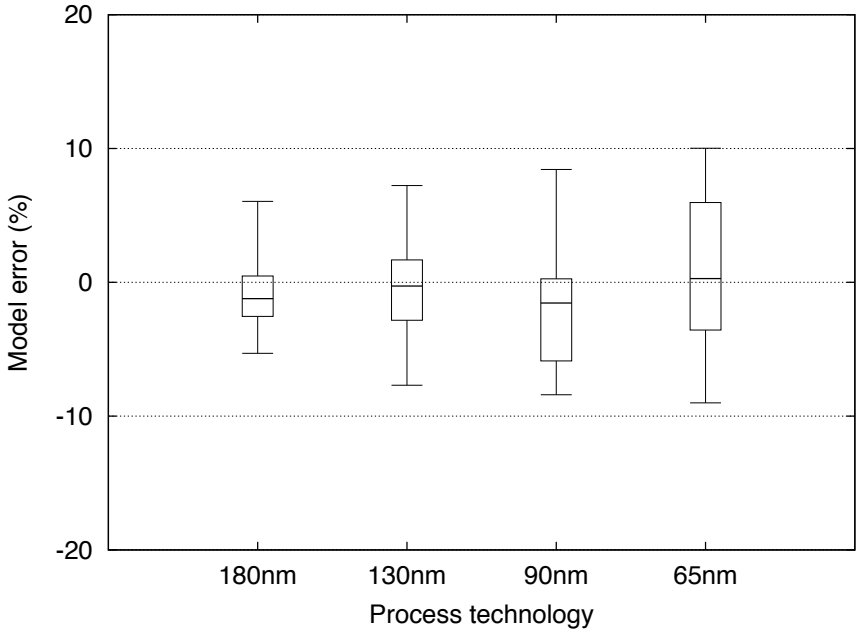


Figure 6.1: Error of the high-level energy dissipation model across different processes.

dom input data for three discrete sub-V_T operating voltages, i.e., 0.1 V, 0.2 V and 0.3 V. The same random data input is also used in the switch level simulations and the energy dissipation values obtained using the model are compared to the SPICE level results. The total error for all the ISCAS85 benchmark circuits for different processes in box-plot format is shown in Figure 6.1. From the comparison, it is computed that the mean error values are -0.71, -0.35, -1.48, 0.61 for 180, 130, 90 and 65 nm processes, respectively. As shown in Figure 6.1, with smaller feature size the first quantile error boundaries increase, but are still below 6 % error. The simulation time of the proposed model is on average $270\,X$ times faster compared to SPICE simulations [33]. Thus, it is proven for different processes that the proposed model is offering high accuracy at a short simulation time, making it highly efficient for high level architecture/process exploration for sub-V_T operation.

6.3 Effects of Process Variation in Sub-threshold Regime

Semiconductor fabrication like all the manufacturing processes is subject to both random and deterministic process variations. The effects of process variation has

been studied extensively in the literature and they may be grouped into global and local variation [75].

Global variation effects all the devices on a single die equally and results in device characteristic variations between the dies. On the other hand local variation affects the devices on the same die and consists of both random and systematic components. In this part, the effects of the random threshold voltage (V_T) variation on the circuit operation in the sub-threshold regime will be shown.

In probability and statistics, the log-normal distribution is defined as the probability distribution of any random variable whose logarithm is normally distributed. The log-normal distribution occurs frequently in sub-threshold circuit design due to the fact that the current depends exponentially on V_T and it is assumed that V_T is normally distributed in local process variation. A basic definition of the log-normal function is given in Appendix B. As sub-threshold current depends on the V_T exponentially, the sub-threshold current and parameters that have a first order relationship with the sub-threshold current, such as leakage energy and circuit delay, show a log-normal distribution under the assumption of normal distribution of V_T mismatch.

The delay variation of a single inverter from a standard cell library in a 0.18μm process is shown in Figure 6.2. The data for the delay variation is gathered from 1000 point Monte Carlo simulations. As it is seen from the figure, the delay variation is minimal at the nominal supply voltage of the process (VDD=1.8V) and has a Gaussian distribution as expected. On the other hand, in the sub-threshold operating regime, the delay has a log-normal distribution with a long tail on the right (VDD=0.3V). This kind of distribution curve implies that the below average delays deviate slightly from the expected value while above average delays can be as high as several times the mean.

For comparing the effects of supply voltage scaling on the circuit performance parameters, the coefficient of variation is a better metric. The coefficient of variation (CV) is defined as the measure of dispersion of a probability distribution and it is defined as the ratio of the standard deviation to the mean and is given by

$$c_v = \frac{\sigma}{\mu}. \tag{6.1}$$

The CV for various supply voltage values is shown in Figure 6.3. The sub-threshold operating regime is easily distinguishable on the figure for the supply voltage val-

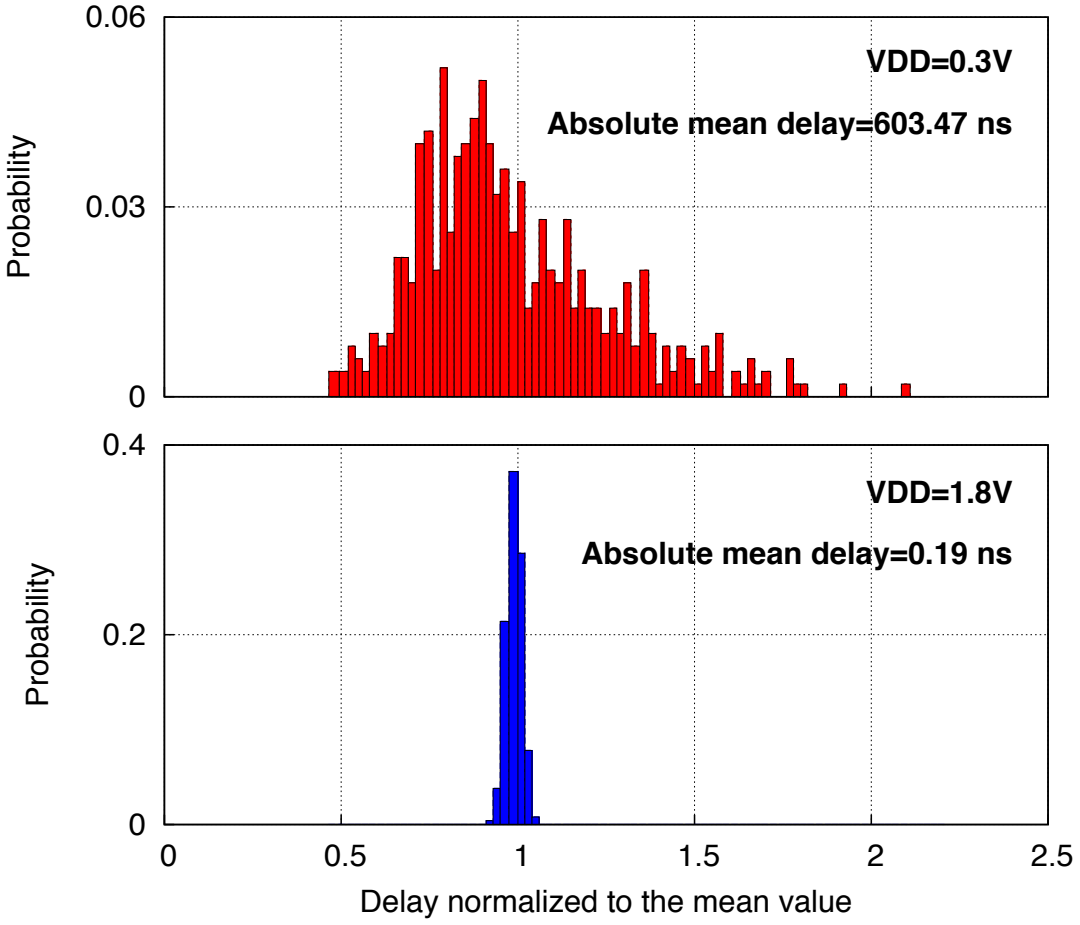


Figure 6.2: Comparison of the delay times for a single inverter under V_T variation. The variation of the average delay is gathered from SPICE level Monte Carlo simulations.

ues less than or equal to 0.4V. Above 0.4V, the delay distribution is Gaussian and dispersion is considerably less than the sub-threshold supply voltages.

From the above analysis it is concluded that for reliable operation in the sub-threshold regime, synchronous circuits must be over designed by a huge margin compared to the above-threshold operation. On the other hand, asynchronous circuits are insensitive to process variations. For an asynchronous system, a job duration is irrelevant because the completion of operation is signaled to the following modules by the previous ones. So for better reliability and yield, asynchronous operation is a strong candidate for sub-threshold operation.

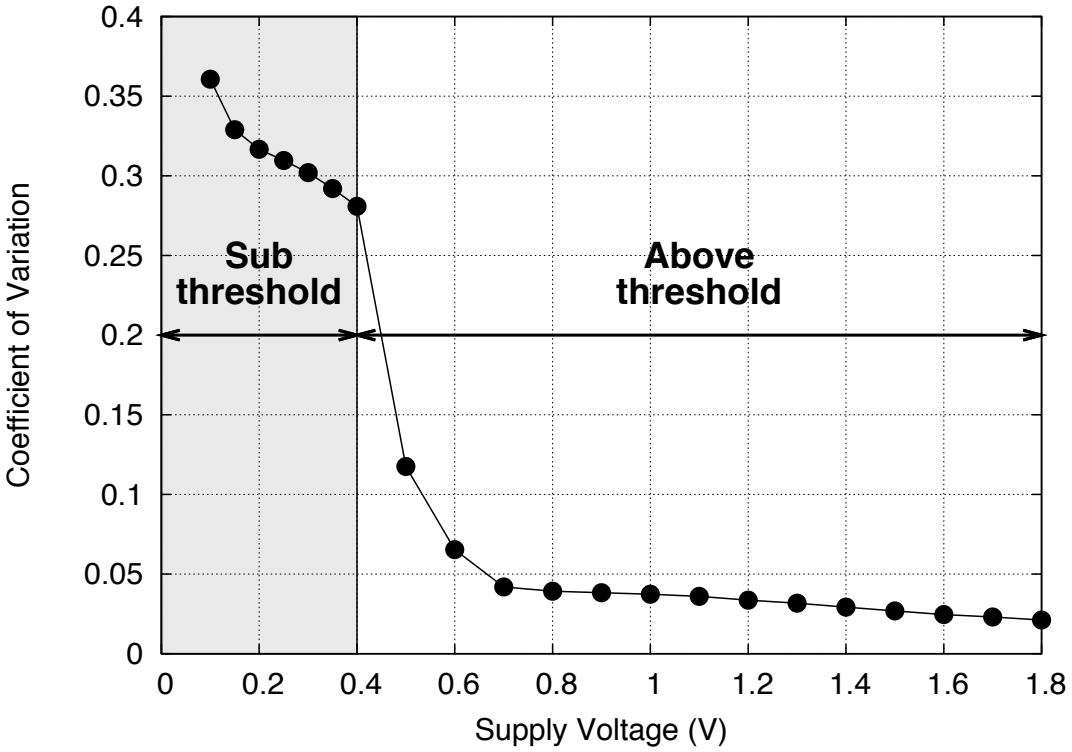


Figure 6.3: Coefficient of variation for various supply voltage values. The sub-threshold and above-threshold regions are marked on the graph for $0.18\mu m$ digital CMOS process.

In Figure 6.4 the energy dissipation overhead due to operating at a lower frequency for realizing higher yield (95%) for the ISCAS85 benchmark circuits is shown. Lowering the operation frequency for higher yield results in higher leakage energy, and moving away from the energy-minimum operating point. From the SPICE simulations, we found that the energy penalty for targeting a higher yield can be up to 40.8% with respect to the typical operation case on the ISCAS85 benchmark circuits.

6.4 Process Variation in Modern Technologies

In modern nanometer technologies random dopant fluctuation (RDF) and global process variations are the dominating effects. Both effects result in shifts from the nominal threshold voltage. Due to the fact that sub-threshold drain current depends on the threshold voltage exponentially, any change in the threshold voltage dominates other process variation effects in the sub-threshold regime.

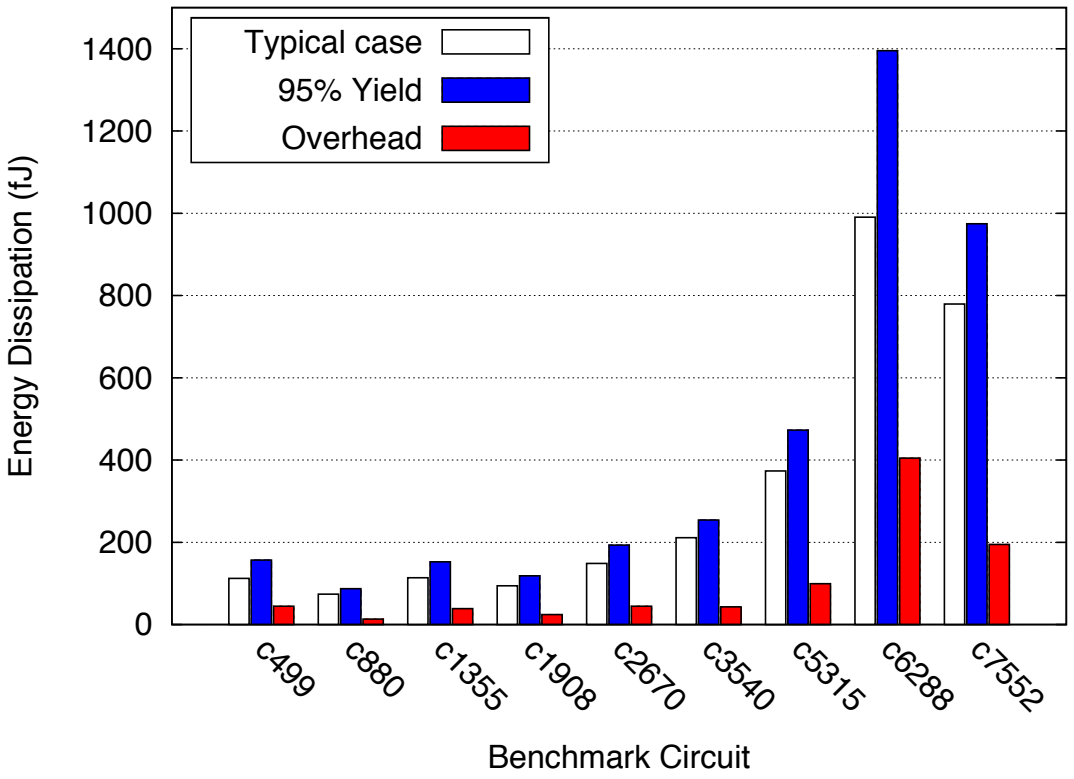


Figure 6.4: Energy overhead due to operating the sub-threshold circuits for functionally correct operation with a 95% yield.

Local process variations during the manufacturing of the circuit, such as geometric variations, are overpowered by the change in the threshold voltage. For example, short-channel effects like drain induced barrier lowering (DIBL) and their effects on the performance and reliability of sub-threshold digital circuits are less pronounced [42].

As previously shown in Section 6.3, sub-threshold circuits are more susceptible to random process variations than their above-threshold counterparts. In some cases, these random process variations cause functional failure. Functional failure of the static CMOS circuits due to the random process variation, i.e., threshold voltage variation, may be investigated using the static noise margin values derived from the voltage transfer characteristic (VTC) curves of digital gates.

To investigate the functional failure of digital gates, we simulated digital gates in different processes following the methodology described in [7] and [43]. The methodology is based on calculating the static noise margin (SNM) of the SRAM cells using butterfly plots. Butterfly plots for two gates are formed by superimposing the VTC

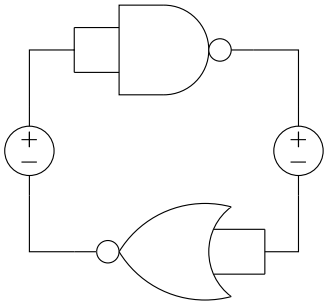
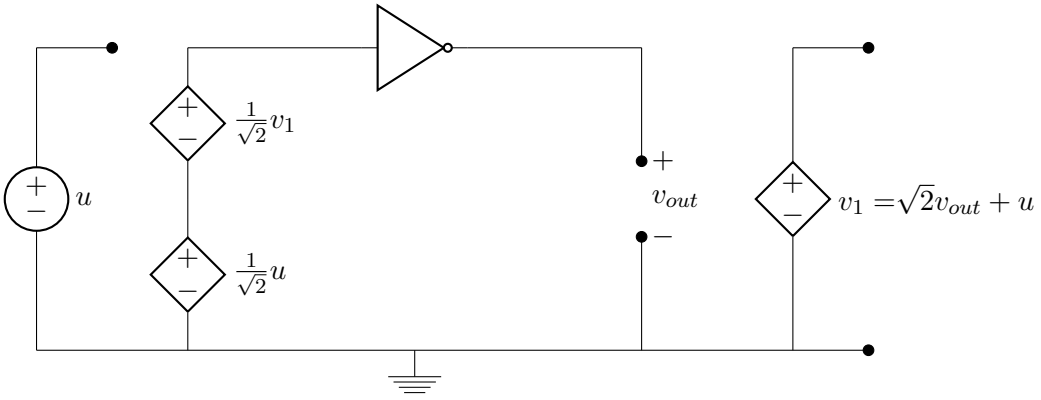
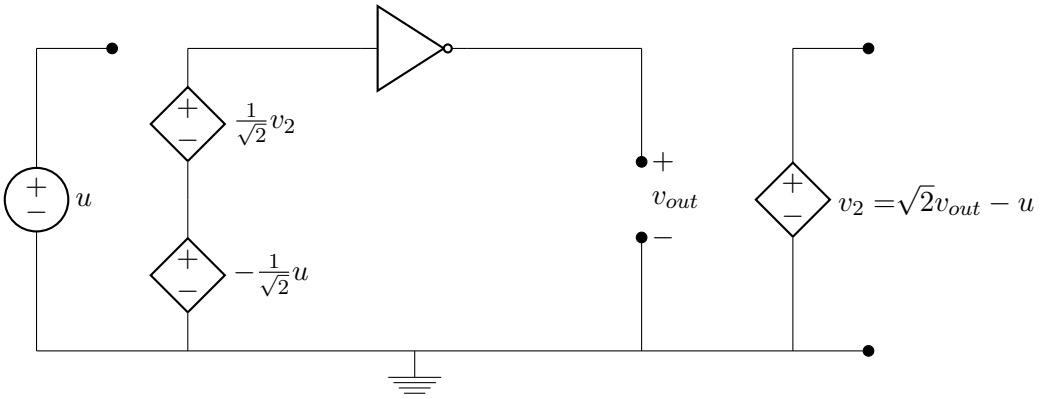


Figure 6.5: Static noise margin testbench for failure rate simulations.



(a) First gate



(b) Second gate

Figure 6.6: SNM testbenches for automatic extraction of SNM values (After [7]).

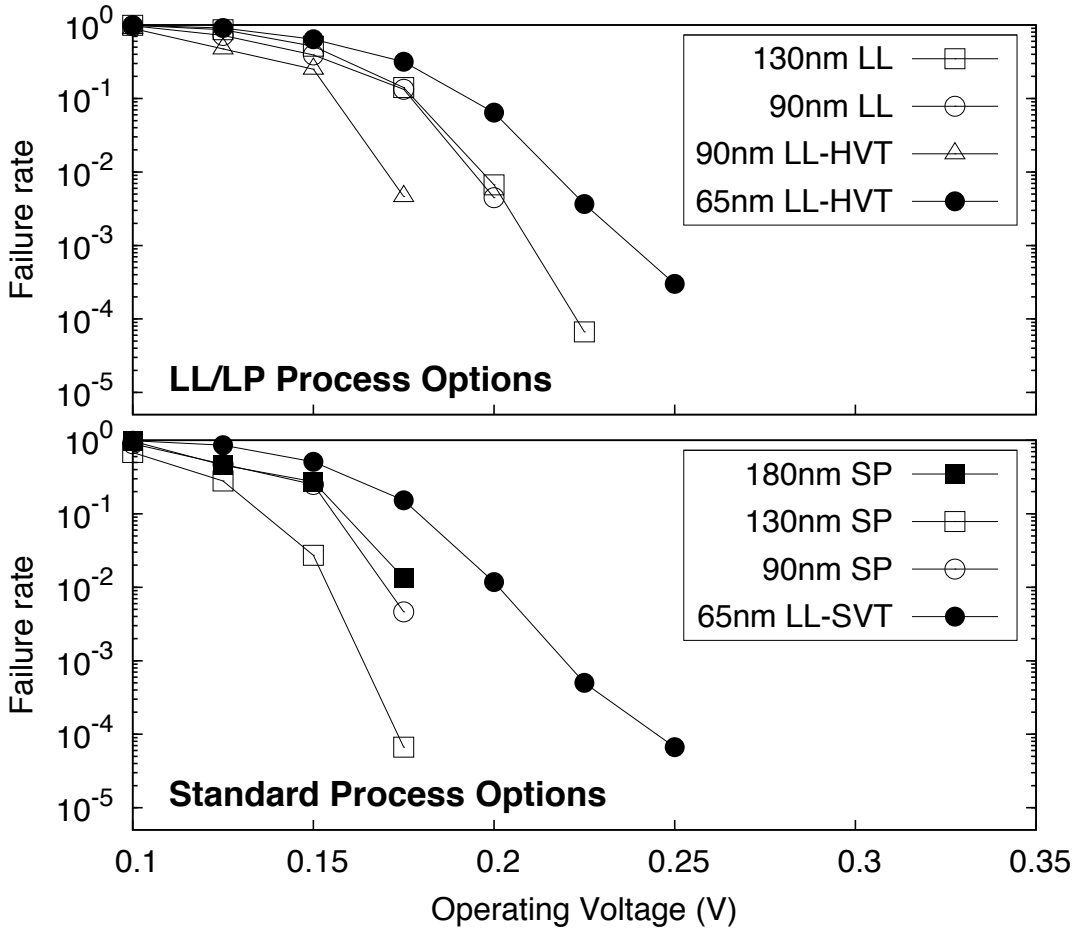


Figure 6.7: Static noise margin failure rates across multiple technologies for changing supply voltages. Top pane shows the low-power (LP) / low-leakage (LL) process options and bottom pane shows the standard process (SP) failure rates.

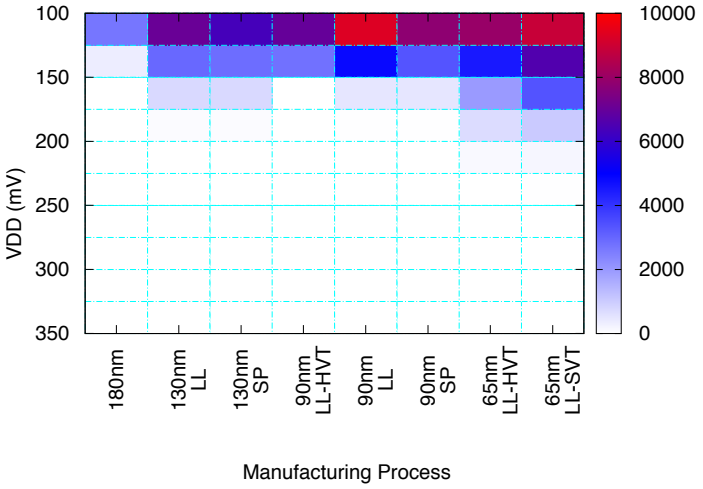
of one gate over the mirrored VTC of the other. This formation simulates whether a logic level can be regenerated in case these two gates are in a logic path one after another. A sample testbench which includes the static-noise sources is shown in Figure 6.5. To simulate the scenario that different gates are connected in back-to-back fashion, testbenches for extracting the SNM data automatically are set, following the simulation method in [7] (see Figure 6.6). In this simulation method, SNM square diagonals are calculated by rotating the voltage axes by 45 degrees. In the rotated plane the diagonal of the SNM square for any voltage is calculated by subtracting one VTC curves from the other. If the calculated value is negative, for that specific supply voltage value SNM is negative, and thus the logic value cannot be regenerated.

We extracted the SNM failure rates of the gates from 10k-point Monte Carlo simulations. For each process the simulations are run for supply voltage values which are varied between 0.1 V and 0.35 V with 25 mV steps. Simulated functional failure rates are presented in Figure 6.7. In this figure simulation results for NAND2-NOR2, NAND3-NOR3 and NAND4-NOR4 pairs are shown. Only operating points with SNM failures are marked. Supply voltage values which realize operation with less than 0.001 failure rate are taken as the minimum reliable operating voltage in our calculations during the rest of this thesis. From Figure 6.7, it is seen that with smaller sized technologies reliable operating voltage increases as expected. It is also noticeable from the figure that special low-power/low-leakage process options have higher failure rates at a given voltage or higher reliable operating voltage.

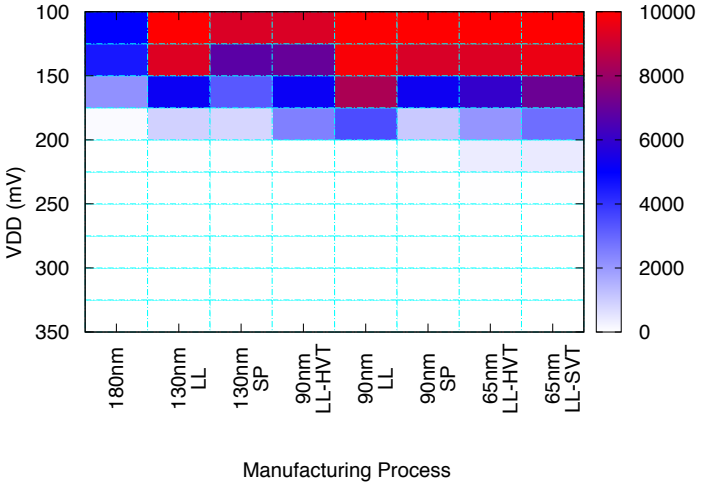
We also investigated the effects of process variation on gates with different number of transistors. For this study, we simulated the failure rates of NAND2-NOR2, NAND3-NOR3 and NAND4-NOR4 gate pairs in separate testbenches. The results of the simulations are presented in Figure 6.8. From the figures, it can be seen that NAND3-NOR3 pair have the highest failure rate for all the processes for varying supply voltage values. We concluded that this occurs because NAND3-NOR3 pair has lower voltage headroom per transistor than NAND2-NOR2 pair. Furthermore, although NAND4-NOR4 pair has the lowest voltage headroom per transistor, because of the number of increased devices, effects of random variation are more evenly spread, resulting in a peak in failure rates for NAND3-NOR3 gates. Another interesting observation from the simulation results is that by employing only higher fan-in gates in 65 nm process, i.e., NAND4-NOR4, it is possible to reduce the reliable operating voltage down to 200 mV. This reduction is important for circuits that have their energy-minimum operating voltage (EMV) lower than the reliable operating voltage or for circuits that are constrained by an external operating frequency, as will be explained in next section.

6.5 Process Comparison

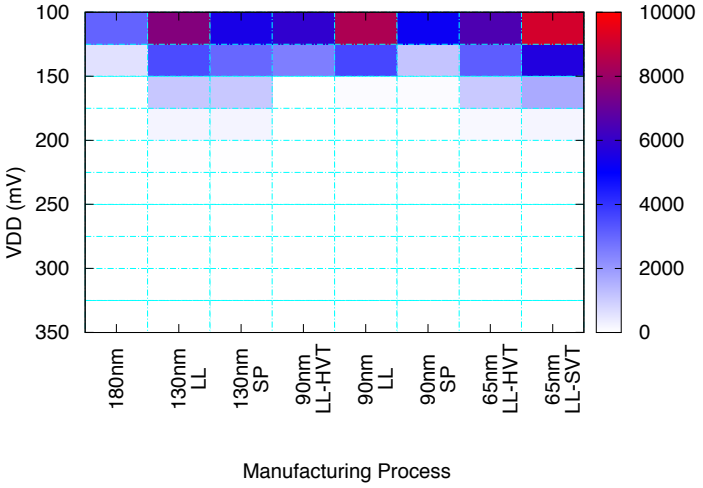
In sub-V_T circuits, EMV depends on the circuit components and process properties according to equation (3.15). The EMV imposes a corresponding clock frequency of the circuit which is not always practical, if the required clock frequency is dictated by external design constraints. Operating at any supply voltage value other than the energy-minimum voltage results in energy dissipation overhead. In Table 6.2, EMV, operating frequency, and the leakage current of the cardiac event detector synthesized



(a) NAND2-NOR2



(b) NAND3-NOR3



(c) NAND4-NOR4

Figure 6.8: Error of the high-level energy dissipation model across different processes. (a) NAND2-NOR2, (b) NAND3-NOR3, and (c) NAND4-NOR4 pair errors are shown.

Table 6.2: Supply voltage, operating frequency and the leakage current of the cardiac event detector at EMV.

Process	VDD (V)	f_{max} (kHz)	Leakage (nA)
180nm	0.34	49.3	142.1
130nm LL	0.32	13.7	24.2
90nm LL	0.29	58.7	84.3
90nm LL-HVT	0.30	29.2	44.4
65nm LL	0.30	165.3	136.7
65nm LL-HVT	0.31	24.8	15.5

in different technologies are shown. It can be seen that at EMV, all the operating frequencies are much higher than our low-end frequency of interest, i.e., 1 kHz.

In [74], optimal technology selection for sub-V_T operation was investigated based on the required clocking frequency and the duty cycle of the circuit. Authors concluded that for low-frequency and low duty cycle operations, older technologies are optimal in terms of energy dissipation. However, only standard threshold voltage technologies were considered. Almost all the smaller feature sized technologies, i.e., 130 nm and below, come with LL process options. The drawback of LL and HVT technologies is the speed penalty. Nevertheless, biomedical applications usually have a low requirement on processing speed, which in turn makes LL and HVT technologies a favorable process choice. Therefore, from now on, in this chapter only technologies with LL and/or HVT feature will be considered for analysis.

Structure of a design is captured in the sub-threshold energy model by k_{cap}, k_{crit}, k_{leak}. The mean μ_e of the switching energy distribution in equation (3.3) depends on the data processed by the circuit, and varies insignificantly across technologies. These small variations are due to minor energy characteristic differences imposed by varying properties of leaf cells in different technologies. Moreover, since all k values are normalized to a basic inverter in a library, EMV varies insignificantly across technologies if the circuit structure is not changed. The sub-threshold slope factor n is the main parameter that changes the optimum supply voltage across technologies. In the technologies examined we found that n varied between 1.38 and 1.49 which resulted in changes in the optimum supply voltage from 0.29 V to 0.34 V (Table 6.2).

For a sub-threshold digital circuit that is integrated in a large system, there are three operating cases based on the operation frequency. When the system frequency is equal to the operating frequency of the sub-threshold circuit at EMV, then energy-minimum operation is realized working at the energy-minimum supply voltage. In

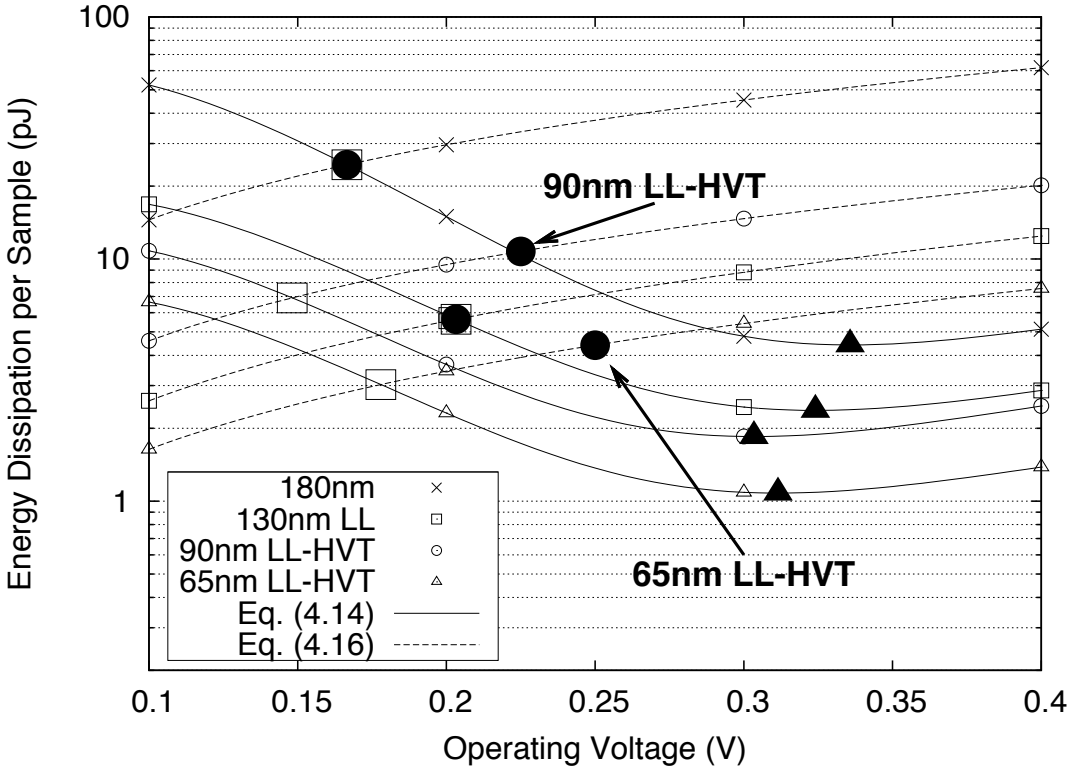


Figure 6.9: Energy dissipation change for varying supply voltages across technology nodes, f=1 kHz.

case where the required operating frequency is higher than the frequency at EMV of the circuit, operating voltage is increased according to equation (3.17) to match the required frequency of operation. If the circuit needs to operate at a lower speed than imposed by EMV, the supply voltage needs to be set to the maximum of either [76]:

- supply voltage which the circuit operates at the required frequency
- supply voltage that satisfies a defined failure rate in ultra-low voltage operation mode.

The energy curves of the R-wave event detector for the examined processes are presented in Figure 6.9. On the curves, EMVs are marked with triangles. However, the external clock frequency of the event detector is lower than the maximum operating frequency at EMV. Thus, operating the cardiac event detector at the optimum voltage is not beneficial due to the excessive leakage energy as leakage energy scales with the operating voltage for a fixed frequency. This case, i.e., circuit operating at a externally set fixed clock frequency, is illustrated by over-plotting equation (3.16) in Figure 6.9.

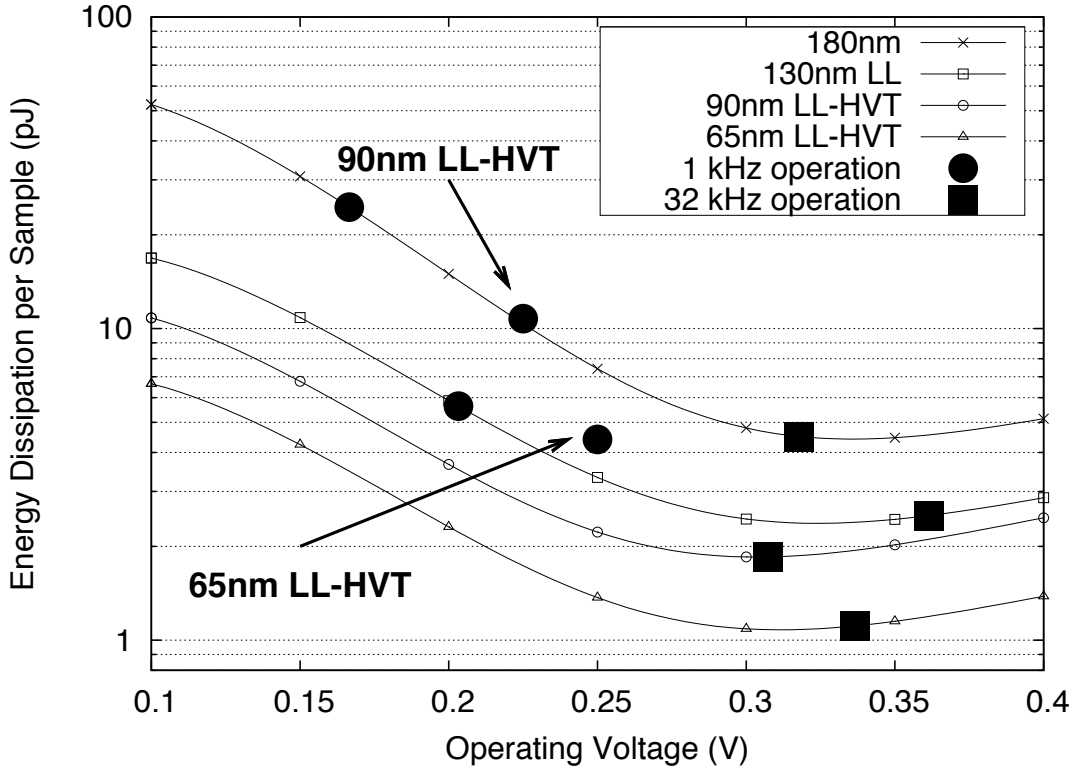


Figure 6.10: Energy dissipation change for varying supply voltages across technology nodes, f=1 and 32 kHz.

While operating at a frequency lower than the EMV frequency, the circuit should be operated at a supply voltage that is adequate to both the clock speed and the reliability requirements. The supply voltages and their respective energy dissipation values that satisfy a 1 kHz clock are indicated by squares in Figure 6.9. Furthermore, due to ultra-low voltage operation, SNM degrades sharply with lower supply voltages, imposing a higher reliable operating voltage. The operating points for 1 kHz operation when reliability is taken into consideration are marked with dots. For 180 and 130 nm processes, the supply voltages that corresponds to a 1 kHz clock are higher than the reliable operating voltages. However, for 90 and 65 nm processes, the supply voltages that satisfy reliability is higher than 1 kHz operation voltage. Although the system clock frequency is fixed at 1 kHz, these higher supply voltages are used to satisfy the reliability criterion. These non-optimal operating points result in excessive energy dissipation in 90 and 65 nm processes, and, thus, the operating points move off from equation (3.14) curves to equation (3.16) curves.

Table 6.3: Supply voltage and energy dissipation across processes for different operating frequencies.

	1 kHz		32 kHz	
Process	V_{DD}	En. (pJ)	V_{DD}	En. (pJ)
180nm	0.17	24.5	0.32	4.5
130nm LL	0.20	5.6	0.36	2.5
90nm LL-HVT	0.23	10.8	0.31	1.9
65nm LL-HVT	0.25	4.4	0.32	1.1

We also studied the operation of the cardiac event detector at a higher frequency, i.e., 32 kHz. Figure 6.10 shows how the operating points change if the clock frequency is increased from 1 kHz to 32 kHz. As reference, 1 kHz operating points from Figure 6.9 are marked as well. While operating at 32 kHz, the operating point voltage is higher than the reliable operation voltage for all processes. Thus, all operating points are located on the energy curves plotted with equation (3.1), unlike the 1 kHz case. An increase of the clock frequency from 1 to 32 kHz, requires a higher supply voltage. This results in higher switching energy dissipation. However, due to the fact that the leakage time is reduced by a factor of 32, total energy dissipation decreases.

6.6 Discussion

Table 6.3 presents the operating points that realize the lowest energy operation across technologies for 1 and 32 kHz operation while satisfying the reliability constraint. Energy dissipation is reduced by 21.4% and 42.1% for 1 and 32 kHz clock frequency, respectively. These reductions are realized by migrating from 130 to 65 nm and 90 to 65 nm, for 1 and 32 kHz operation, respectively. In our analysis it is found that even though 90 nm process offers LL-HVT option, the leakage current is still higher than 130 nm LL and 65 nm LL-HVT processes. Hence the investigated 90 nm LL process is unattractive for low-speed, low-energy applications. Furthermore, the analysis shows that if smaller process nodes are well-tuned for low-leakage operation, migrating to smaller technologies is beneficial as the energy curves in Figure 6.10 shift according to C_{inv} in equation (3.14), which scales with the technology feature size.

6.7 Conclusions

In this chapter we presented the optimal process selection for a real world circuit implementation working in the sub-threshold regime. Unlike the optimal process selection for sub-V_T studies published before, we show that with special low-power process options, which are available for 130 nm and below technologies, it is beneficial to migrate to smaller feature sized technologies as the energy dissipation curves shift with technology scaling. The energy dissipation reduction of the digital cardiac event detector is confirmed for a corner case operation of 1 kHz, where migrating technologies results in 21.4 % energy dissipation reduction even when taking the reliability into consideration. For 32 kHz operation, technology migration results in 42.1% energy efficiency improvement. For higher speed operation, technology migration is still expected to enhance energy efficiency if designs are operated at their EMVs under reliability constraints, as for a design, energy curves for different technologies shift according to the basic capacitance value, i.e., gate capacitance of an inverter.

Chapter 7

Energy Reduction by Hardware Optimization

This chapter presents hardware and design structure oriented energy reduction techniques for sub-threshold circuits. First, common practices such as pipelining and parallelism are evaluated for synchronous and asynchronous operations. Afterwards, a digital signal processing (DSP) oriented technique, i.e., folding, is applied to the basic digital cardiac event detector circuit for reducing leakage while operating at frequencies lower than the EMV frequency.

7.1 Basic Architectural Improvements

In this section basic architectural improvements such as parallelism and pipelining is investigated for their energy efficiency in the sub-threshold regime for both synchronous and asynchronous operations. During the application of the energy model to parallel and pipelined cases, we focused on direct transformations from synchronous design to asynchronous design.

7.1.1 Parallelism

A parallel architecture is presented in Figure 7.1. Parallelism has been used in above-threshold logic circuits for either improving the throughput, or for reducing the energy dissipation by trading the throughput improvement for lower supply voltage, hence lower power consumption [77]. In this section, we study the effects of parallelism on the operation of synchronous and asynchronous circuits. Our study in parallel

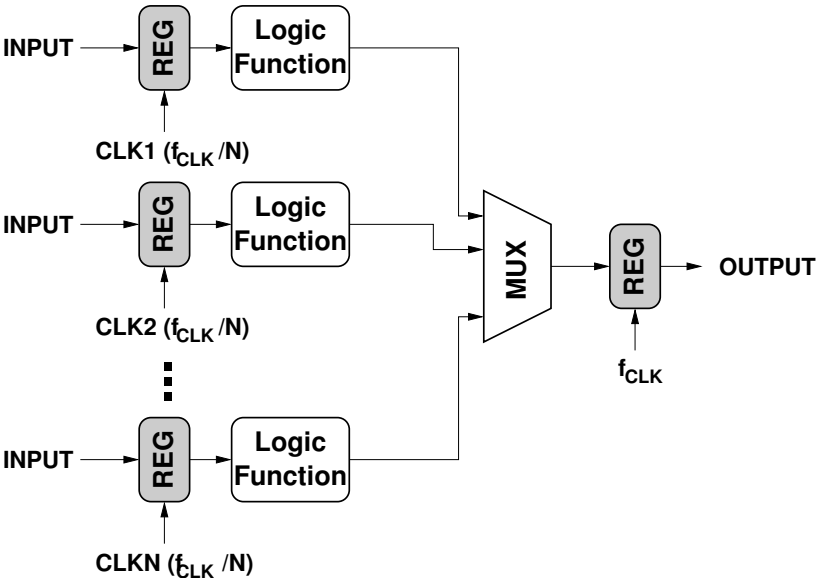


Figure 7.1: Parallel system that consist of N copies of the same logic function running in parallel.

architectures focuses on so-called embarrassingly parallel operations that run N copies of the same logic function in parallel.

Assuming the parallelizing energy dissipation overhead, such as multiplexers/demultiplexers, is negligible compared to the energy dissipation of the main logic block, the asynchronous parallellized energy equation for one observation frame is defined as

$$E_T = M\mu_e k_{cap-logic} C_{inv} V_{DD_{parM}}^2 + M k_{leak} I_0 V_{DD_{parM}} T_{ref}, \tag{7.1}$$

where M is the degree of parallelization (number of parallel blocks), and T_{ref} is the observation frame for the reference case. During this time frame, M computations take place, hence the throughput is improved by a factor of M. By dividing equation (7.1) by M, energy dissipation per copy of the logic block is found. It is easily seen that the resulting equation has the same form as (3.5) and the energy-minimum operating voltages are the same as equations (3.13) and (3.15) for the asynchronous and synchronous cases, respectively. This means, regardless of the number of parallel copies of the same circuit, energy-minimum operating voltage of M copies of the same circuit is fixed and equal to the single copy case.

As a different case, the voltage may be lowered so that the throughput is fixed to that of a single block for M copies, resulting in energy sub-optimal operation. For

this case the open form of timing relation in terms of the new and reference supply voltage is given by

$$k_{crit}\frac{C_{inv}V_{DD_{parM}}}{I_0 e^{V_{DD_{parM}}/(nU_t)}} = Mk_{crit}\frac{C_{inv}V_{DD}}{I_0 e^{V_{DD}/(nU_t)}}. \tag{7.2}$$

Solving (7.2) for $V_{DD_{parM}}$ we find the new supply voltage that realizes the same throughput for M parallel copies of the circuit as

$$V_{DD_{parM}} = -nU_t W_{-1}\left[-\frac{MV_{DD}e^{-V_{DD}/(nU_t)}}{nU_t}\right]. \tag{7.3}$$

If the same procedure is followed for the calculation of the reduced voltage for synchronous operation as well, the same result as in (7.3) is found. To be able to see the effects of parallelization on the energy dissipation of both synchronous and asynchronous operation, the switching/delay distribution mean is fixed to a value, and affects of changing the supply voltage is observed. During the calculations, a communication overhead of two-tenth of the critical path of the main circuit for asynchronous operation is assumed. For the asynchronous and synchronous cases, the effects of parallelism are shown in Figure 7.2.

As it is clear from the plots, parallelism do not reduce the energy per operation for both the asynchronous and synchronous cases. In the ideal case without any parallelism overhead, the energy-minimum operating point moves but still results in the same minimum energy dissipation. This comes from the fact that while reducing the voltage for trading the throughput for lower energy dissipation, delay of the circuits increase exponentially and the parallelization and supply voltage reduction only result in the shifting of the energy-minimum operating point to higher voltage values. Shifting of the optimum operating voltage to higher values comes from the fact that the leakage energy increases more than the switching energy with parallelization, and the optimum operating point for the system moves to a relatively faster operating region. Based on the presented results, the benefit of parallelization in the sub-threshold regime is to increase the throughput. Once the energy-minimum operating voltage of the circuit is known, the structure may be parallellized to increase throughput. This technique was employed by Sze et al. in Ref. [78] for realizing a UWB baseband processor operating with a supply voltage of 0.4V.

Compared to the parallellized case, the energy dissipation reduction due to the asynchronous operation for different levels of parallelization is the same regardless of the number of parallel blocks for a fixed communication overhead making asyn-

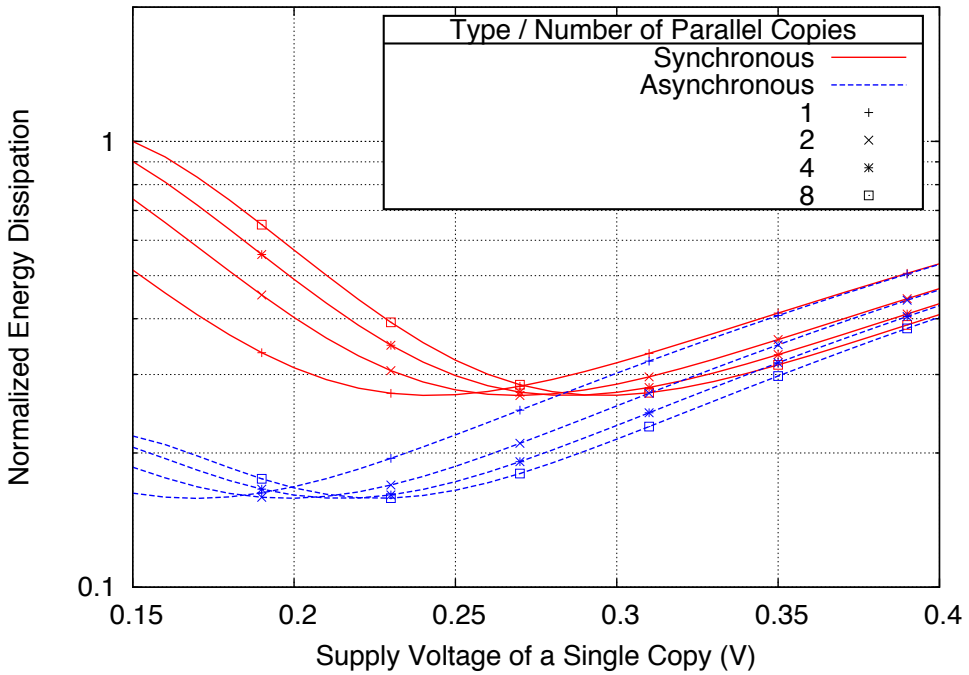


Figure 7.2: Effects of parallelism on the energy dissipation.

chronous sub-threshold operation more energy efficient for the required throughput. It should be emphasized that, if asynchronous operation specific techniques, such as parallel (wagging) FIFOs [52, 79] are used during the implementation, further energy reduction in sub-threshold operation is possible.

7.1.2 Combinational Logic Only Pipelining

Another widely used architectural improvement to either improve throughput or to reduce energy dissipation is pipelining, see Figure 7.3. In this section effects of pipelining on energy dissipation of sub-threshold operation will be analyzed. As in the previous section, we will focus on direct architectural transformations. During our simulations

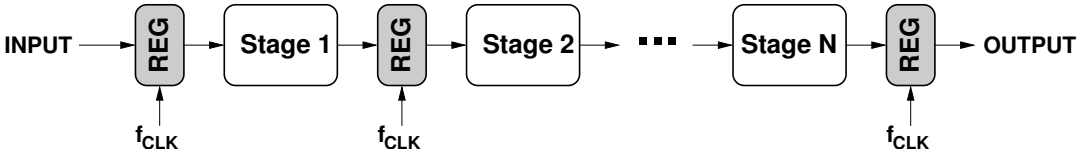
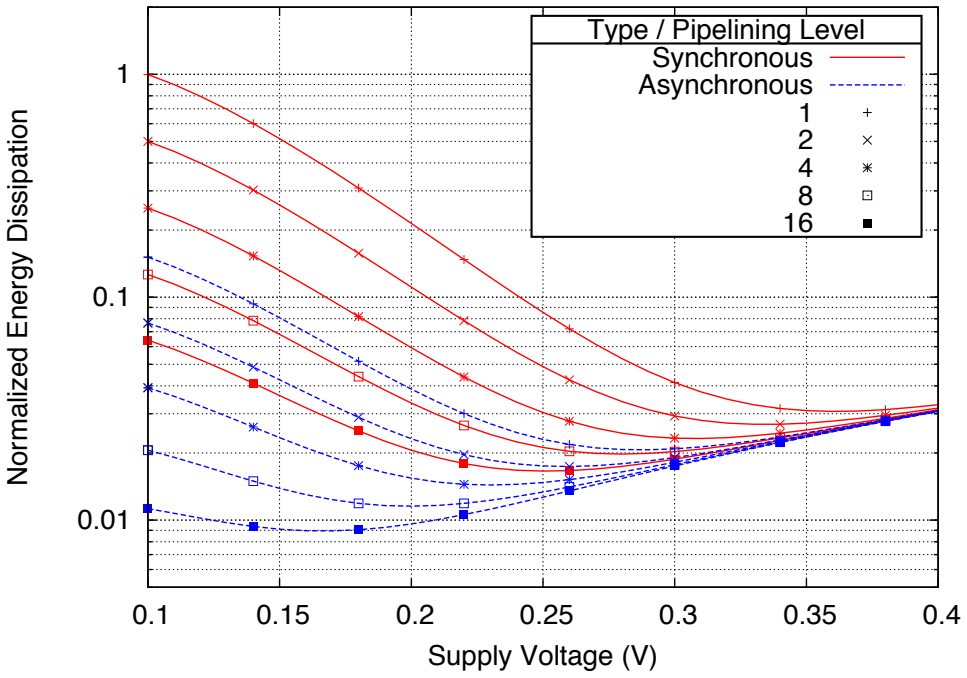
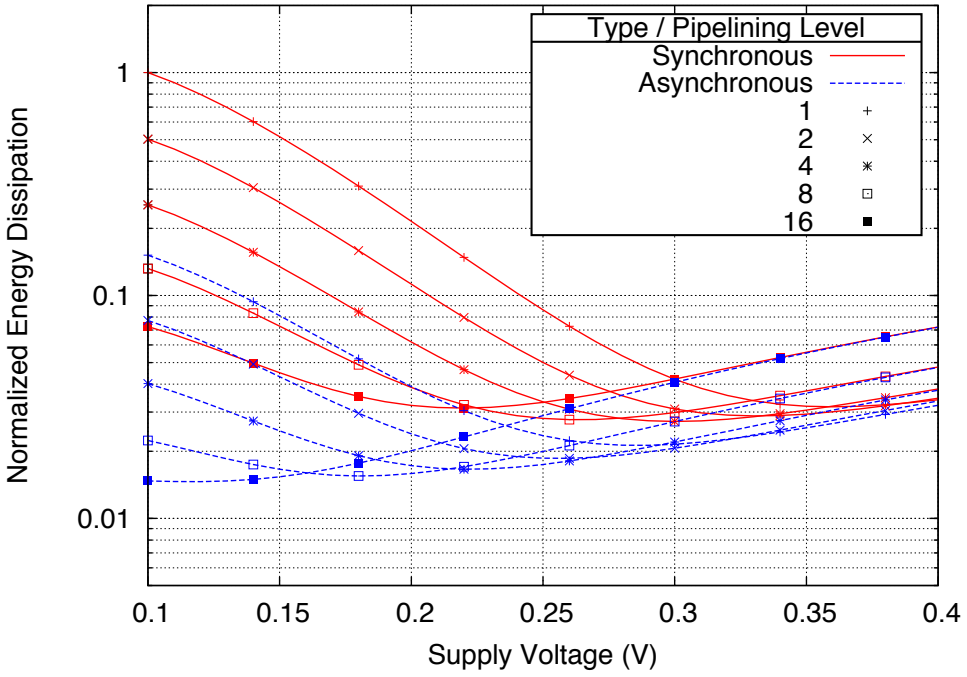


Figure 7.3: Pipelined system with a pipelining level of N.



(a) Without pipelining overhead.



(b) With pipelining overhead.

Figure 7.4: Energy dissipation for different levels of pipelining: (a) Without and (b) with pipelining overhead.

we employ the same memory elements, D type flip-flops in our case, for both synchronous and asynchronous operation. Latches, which are usually used as memory elements in asynchronous design increase the energy efficiency of asynchronous designs when compared to flip-flop based synchronous design. Although latches may also be used for synchronous design as well, their inclusion in the design is costly in terms of design and verification time and is not the common practice. To be able to investigate the improvement in energy dissipation solely due to the average-case performance property of asynchronous operation and not to give asynchronous design an unfair advantage, we will assume that same type of memory elements are used for pipelining for both cases.

Unlike employing parallel copies of the circuit, the hardware overhead due to the pipelining is not negligible and needs to be included in the energy dissipation equations. The hardware overhead in pipelining is due to the sequential elements for dividing the circuit into equal-delay parts and their related routing capacitance. In our analysis we will only include the extra switched capacitance of the memory elements. Following the pipelining analysis methodology presented in [80] for above-threshold operation and adapting it for sub-threshold operation, the asynchronous energy dissipation for a single computation with a pipelining level of N is defined as

$$E_T = C_{inv}V^2\Bigg(\mu_e k_{cap-logic} + \mu_e B k_{cap-flop}N^{\rho} + \frac{k_{crit}}{N}(k_{leak} + B k_{leak-flop}N^{\rho})(\mu_d + k_{comm_oh})\Bigg), \tag{7.4}$$

where B is the word-length of the circuit, $k_{cap-flop}$ is the extra capacitance factor due to a single memory element, $k_{leak-flop}$ is the average leakage factor of a single memory element and ρ is the memory element growth factor that is super-linearly proportional to the pipelining depth N [80]. If the number of memory elements grow linearly with the number of pipelining stages, ρ is set to 1. However, it was shown in [81] that the number of memory elements grow super-linearly with increasing pipelining, so in our analysis we will assume that $\rho = 1.2$. It should still be noted that ρ parameter depends on the circuit structure and is a design-specific parameter.

Applying the same methodology to the synchronous case, we get the synchronous pipelined energy dissipation as

$$E_T = C_{inv}V^2\Big(\mu_e k_{cap-logic} + \mu_e B k_{cap-flop}N^\rho + \frac{k_{crit}}{N}(k_{leak} + B k_{leak-flop}N^\rho)\Big). \tag{7.5}$$

In equations (7.4) and (7.5), $k_{cap-flop}$ and $k_{leak-flop}$ are estimated from the standard cell library specifications. Calculations for both synchronous and asynchronous pipelining cases with and without the overhead due to the sequential elements are presented in Figure 7.4. The following deductions from the calculations and the figure are made:

- Although significant energy reduction is realized for synchronous operation without overhead, when the overhead is included in the calculations energy dissipation reduction is marginal.
- Asynchronous operation benefits from pipelining more than the synchronous operation, and in the 0.18μm CMOS technology, the limits of pipelining is set by the lowest working voltage of the sequential elements.

Furthermore, the energy reduction of asynchronous operation with respect to the synchronous operation for different levels of pipelining for changing switching/delay probabilities is analyzed. The results of the analysis are presented in Figure 7.5. The top curve in the plot shows a pipelining level of 16. By employing pipelining at this level, an energy reduction of up to 49% may be realized with respect to synchronous operation. For switching/timing coefficients greater than 0.9, energy dissipation of asynchronous operation is higher. Higher energy dissipation is due to increased leakage energy which is a result of the asynchronous communication overhead. An analysis for the optimum pipelining for different switching/delay probabilities of the circuit is performed. The results of the analysis are shown in Figure 7.6. For the calculations in Figure 7.5, the same level of pipelining was applied to both synchronous and asynchronous operation at a specific switching/delay mean. On the other hand, for the results shown in Figure 7.6, minimum energy pipelining level and respective energy dissipation for a specific switching/delay mean is calculated for both synchronous and asynchronous operations separately and the obtained results are compared. As it is seen from the figure, asynchronous operation may realize an energy reduction of up to 47% with respect to the synchronous operation with optimal pipelining.

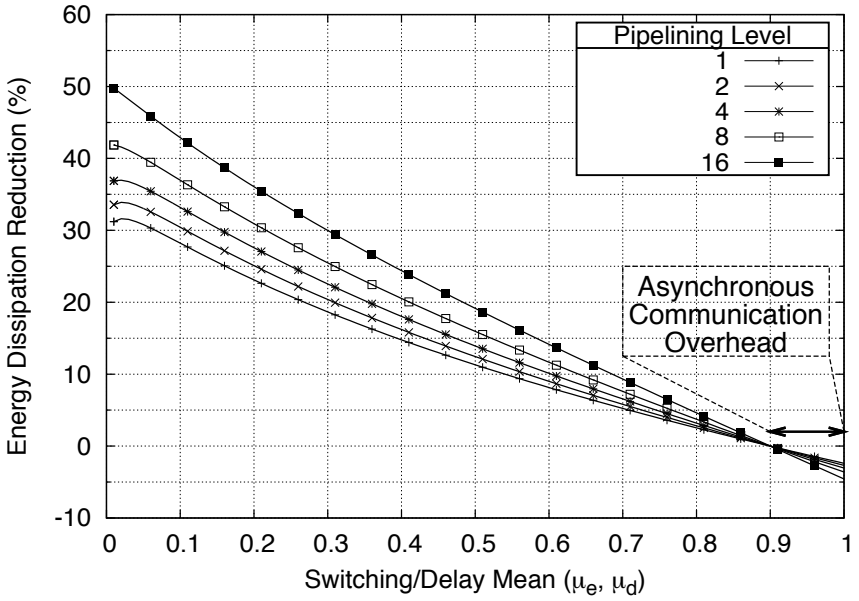


Figure 7.5: Realizable energy gain with asynchronous operation for different pipelining levels. The region where synchronous operation results in lower energy dissipation due to the asynchronous communication overhead is emphasized.

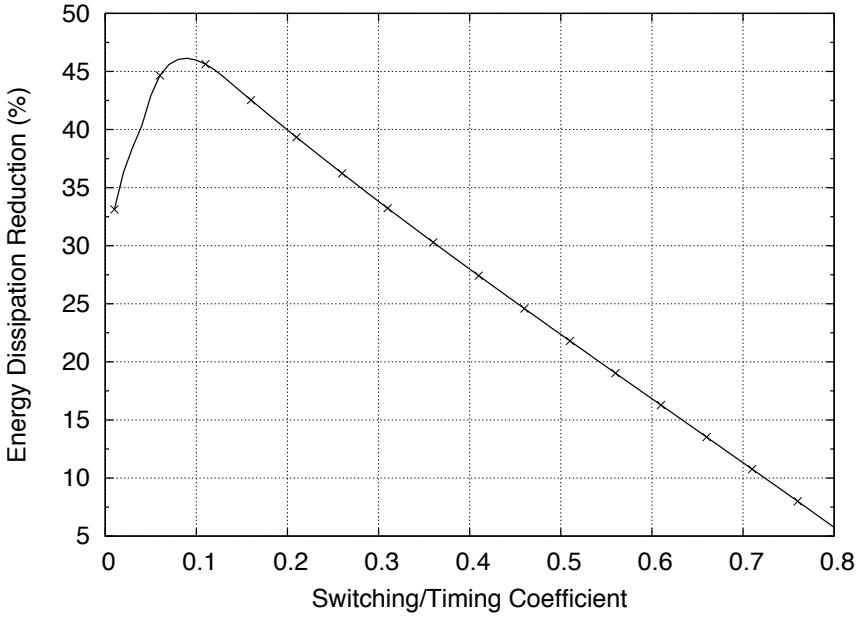


Figure 7.6: Realizable energy gain for optimum pipelining at different switching/delay probabilities.

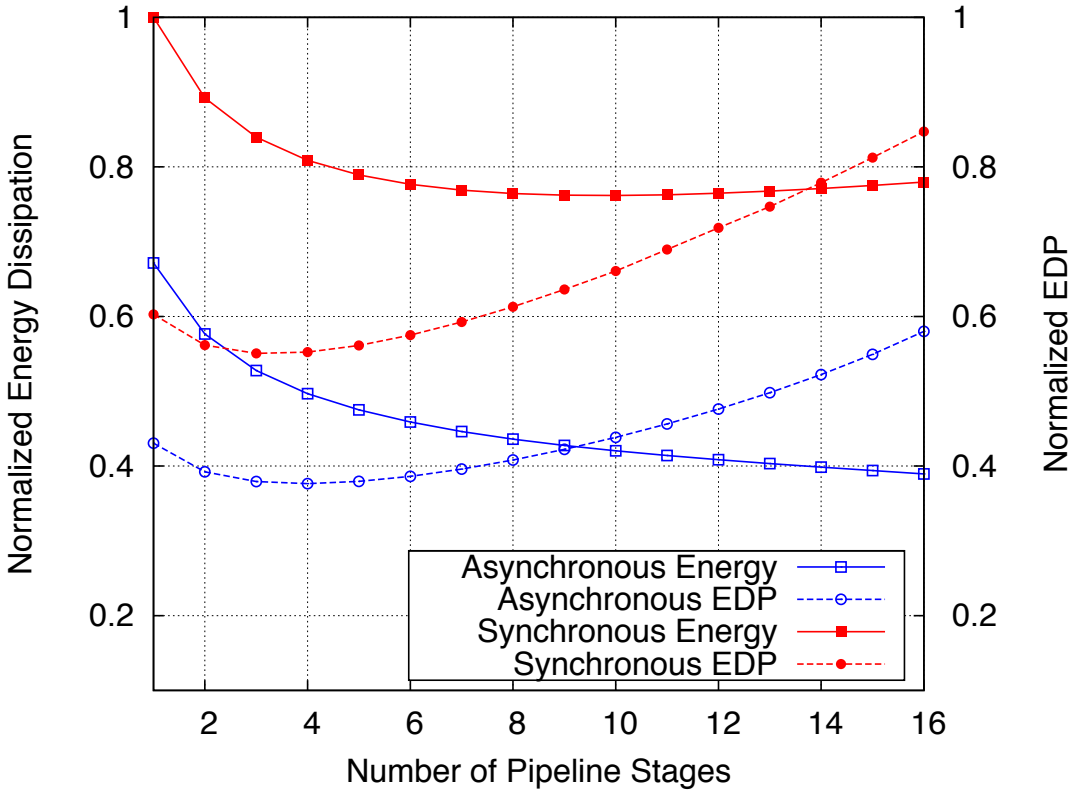


Figure 7.7: Optimum number of stages for EDP/Energy minimization.

Similar numerical simulations were run for a relatively bigger system. The simulated system consist of 25000 NAND2 equivalent gates. The energy dissipation and energy-delay product (EDP) of the system for both synchronous and asynchronous operation for changing pipelining levels are calculated. The results are shown in Figure 7.7. Optimum pipelining depth for synchronous and asynchronous minimum EDP is different, i.e., 3 and 4 levels, respectively. One important thing that should be noted is that although there exists a minimum energy operation pipelining level for the synchronous case, for the asynchronous case no such limit exists and even lower minimum energy dissipation occurs with increasing pipelining level. If synchronous and asynchronous operations are compared for their energy efficiency and EDP at their minimum EDP points, asynchronous operation dissipates 41% less energy for a 32% lower EDP.

We investigated the energy dissipation reduction due to pipelining for real world applications on the benchmark designs. The energy dissipation values for unpipelined and for pipelined cases are shown in Figure 7.8. Energy dissipation values in the

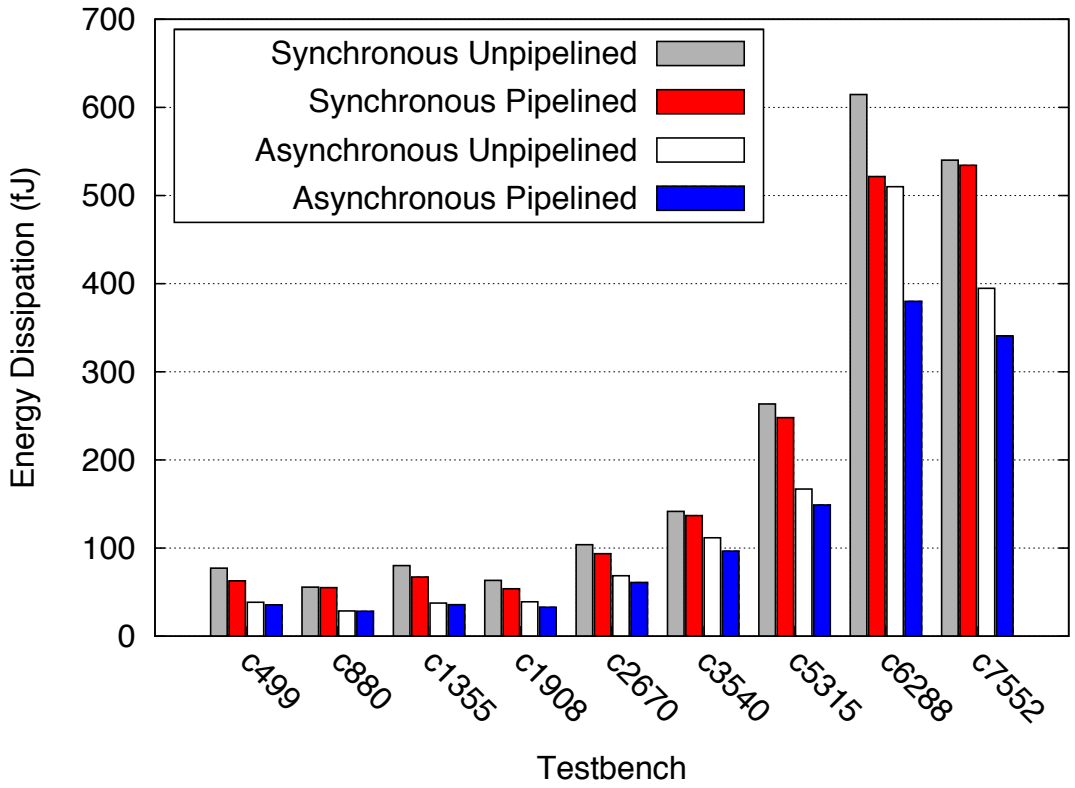


Figure 7.8: Energy dissipation reduction in ISCAS85 benchmark circuits due to pipelining. Both pipelined and unpipelined minimum energy dissipation values for asynchronous and synchronous are shown.

pipelined structure represent the dissipation at the optimal pipelining level for minimum energy operation. The unpipelined architecture has flip-flops at the output pins, hence the energy dissipation differences when compared to Figure 4.3b. The word-length value (B) is chosen as the minimum of number of input pins, number of output pins, or the number of flip-flops that have a total capacitance of 25% of total switched capacitance of the purely combinational logic block. A value of 25% is chosen based on the complex industrial circuits and taken from [80]. The word-length values used in the computations and the percentage of the energy dissipation reduction are given in Table 7.1. The reduction values given relative to the unpipelined cases for synchronous and asynchronous operations separately. One thing that should be noted from Figure 7.8 is that for all the reference designs, unpipelined asynchronous energy dissipation is lower than the synchronous energy dissipation for both pipelined and unpipelined cases. By applying pipelining on asynchronous circuits, even lower energy dissipation is achieved.

Table 7.1: Word-length values used in the calculations and energy reduction due to pipelining.

		Energy reduction (%)	
Testbench	Word-length B	Synchronous	Asynchronous
c499	14	18.6	7.7
c880	14	0.8	1.3
c1355	15	16.2	4.9
c1908	10	15.0	15.8
c2670	22	9.8	11.0
c3540	22	3.3	13.6
c5315	52	5.9	10.8
c6288	32	15.1	25.5
c7552	62	1.1	13.7

7.1.3 Pipelining for Register Heavy Circuits

In this section effects of pipelining on the cardiac event detector that has been described in Chapter 5 is investigated. The gate composition of the event detector includes substantial amount of registers, see Table 7.2. According to the synthesis results presented in the table, leakage contribution of the combinational gates and registers are almost equal at 50.23% and 49.77%, respectively.

For circuits that consist of a high number of registers, instead of assuming linear division of the circuit critical path by equally spaced pipeline stages with super-linearly increasing number of registers as in the previous section, an incremental methodology is followed. Both switching and leakage overhead due to the extra registers are added to the starting case based on the pipelining level. The architecture of the circuit on which pipelining in different levels is applied is presented in Figure 7.9.

During the first step of our pipelining exploration, we implemented the register banks to divide the critical path of the circuit into shorter paths. Using this manual pipelining technique we generated and synthesized three different architectures.

Table 7.2: Composition of the cardiac event detector in terms of combinational logic gates and registers.

Logic element	kLeak	Area
Register	6987.74	5670.60
Combinational	7053.26	8578.44
Total	14041.00	14249.04

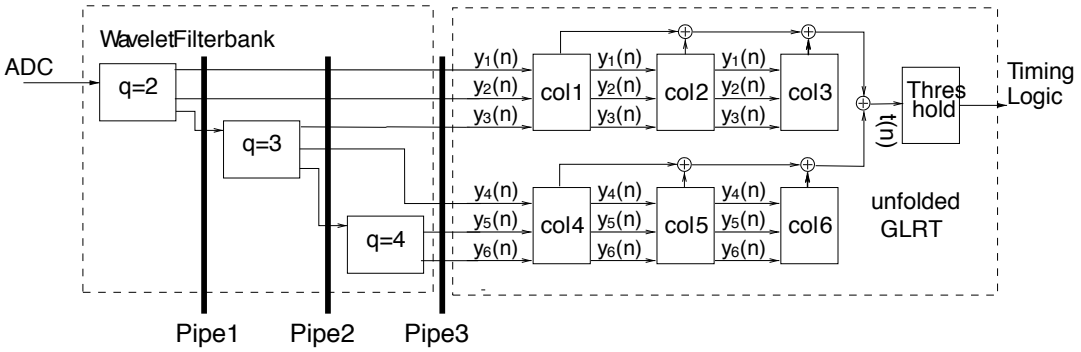


Figure 7.9: Parallel architecture of the wavelet filterbank and GLRT. Manually placed pipeline stages are shown.

These architectures are created by adding pipelining registers to the circuit shown in Figure 7.9. In the first architecture only one pipelining stage is used, that is between $q = 2$ and $q = 3$ banks. The second architecture in addition to the first, includes another pipelining stage between $q = 3$ and $q = 4$ filter banks. This also necessitates adding another pipelining stage at the y_1 and y_2 outputs of the $q = 2$ filter bank. Third implementation with three main pipelining stages is similar to the previous ones and another register set is inserted at the output of $q = 4$ bank. To be able to equalize the delay of the data to the GLRT block, a new set or equalization registers on y_1, y_2, y_3 and y_4 signals, which are going into the GLRT block, are inserted.

In the second phase of our pipelining exploration, we employed different pipelining options of the digital circuit synthesis engine. For synthesis we used Synopsys Design Compiler (DC). DC offers three main methods for optimizing the critical path of the circuit. These methods from [82] are `pipeline_design`, `balance_register` and `optimize_registers`. As `pipeline_design` method is only applicable to purely combinational circuits such as ISCAS85 benchmark circuits, we employed different combinations of `balance_register` and `optimize_registers` commands. `optimize_registers` command optimizes timing (for a given specification) and area (minimizes the registers) and introduces registers to the design whenever necessary. On the other hand `balance_registers` command moves existing registers to realize a shorter critical path and balancing the pipeline stages. In this last method no area minimization is performed.

Using different combinations of the synthesis engine pipelining commands, three different circuit designs were created:

- **onlyBalReg:** Only `balance_registers` command is used to optimize the timing.
- **pipedOptReg:** Only `optimize_registers` command is used for timing optimization.
- **pipedBalReg:** First `optimize_registers` command is run followed by `balance_registers`.

Following the synthesis of all different versions of the event detector, data extraction and simulations as explained in Section 3.3 are applied to all the designs. K-parameters and switching properties of the circuits are presented in Table 7.3.

Using the design data presented in Table 7.3, energy estimation model is applied to all the architectures, and their energy profiles are generated. The results are presented in Figure 7.10. Analyzing the simulation results in the figure and the table, we found that manual pipelining results in mediocre energy dissipation reduction when compared to the synthesis tool based results. The greatest energy dissipation reduction is realized by the *pipedOptReg* circuit followed by *pipedBalReg* and *onlyBalReg*. So it may be concluded that for optimizing timing and register usage of the sub-threshold digital circuits, synthesis tools may be used reliably and they perform better than manually pipelining circuits.

To view the effects of changes made to the cardiac event detector circuit in a compact way, we may visualize the circuits in terms of their normalized $k_{Leak}k_{Crit}$ and $\mu_e k_{Cap}$ parameters. These values are normalized to the basic architecture, i.e., unpiped circuit. Normalizing these values and plotting them as in Figure 7.11 gives quick insight into the results of the optimizations. Lines $x = 1$, $y = 1$ and $y = x$ are added as references. $\mu_e k_{Cap}$ and $k_{Leak}k_{Crit}$ represent the switching energy and leakage energy components of the energy profile, respectively.

Table 7.3: Composition properties of multiple versions of pipelined circuits.

Circuit	kCap	kLeak	kCrit	μ_e
pipe1	18022.6	14121.8	462.8	0.33
pipe2	18153.7	14384.3	389.9	0.30
pipe3	18377.2	14843.5	380.8	0.29
pipedBalReg	18752.4	15356.9	229.4	0.27
pipedOptReg	17216.0	12460.	283.5	0.29
unpiped	17979.3	14041.	519.0	0.34
unpipedBalReg	18883.8	15676.3	219.4	0.28

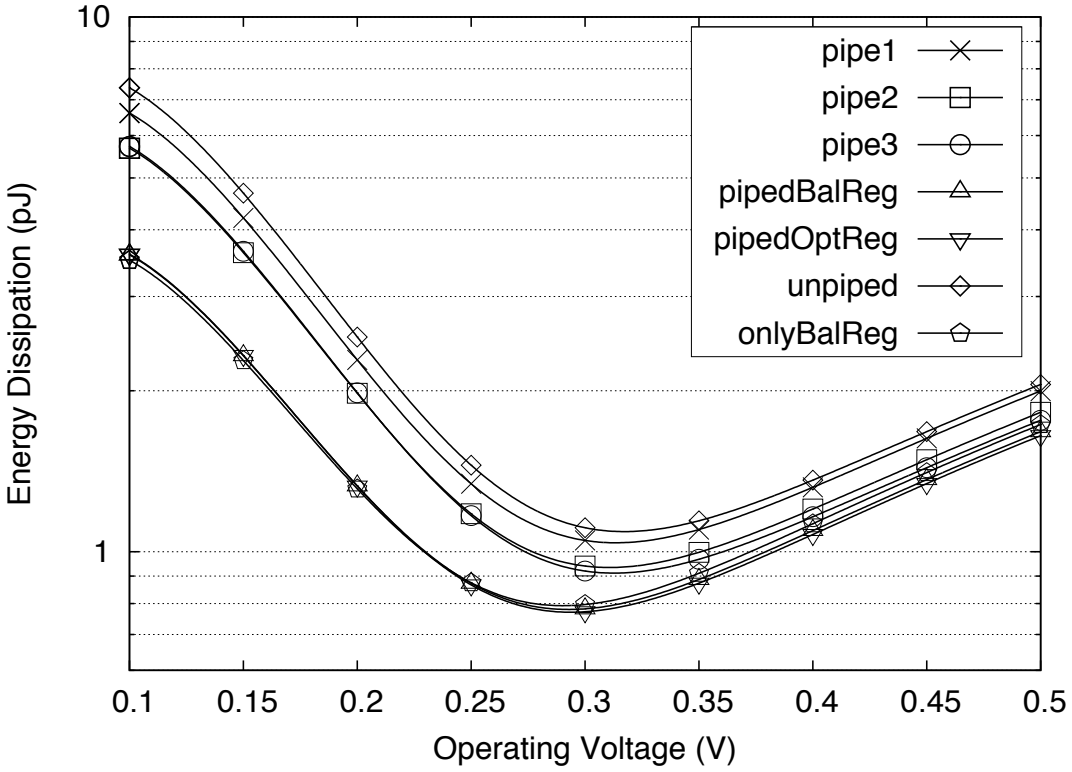


Figure 7.10: Energy curves for different pipelining levels.

In Figure 7.11 any value greater than 1 means that the modification resulted in overhead. For example, if the normalized $k_{Leak}k_{Crit}$ parameter of a design is greater than 1, this means for the same operating voltage leakage energy of the modified design is greater. Furthermore, $y = x$ line is used to deduce whether the modification resulted in lowering the energy-minimum voltage or not. Energy optimum voltage from Chapter 3 is

$$V_{opt-sync} = 2nU_t - nU_tW_{-1}\left[-\frac{2e^2k_{cap-logic}\mu_e}{k_{crit}k_{leak}}\right]. \tag{7.6}$$

Due to the properties of the LambertW function, as the ratio of $\frac{\mu_e k_{Cap}}{k_{Leak}k_{Crit}}$ increases, energy-minimum operating voltage shifts to a lower value due to higher relative switching energy. This case is shown in the plot by the region over $y = x$ line. In Figure 7.11 all the pipelined designs are above the $y = x$ line, thus have lower EMV than the unpiped version, as already illustrated in Figure 7.10. Furthermore, being on the upper region of $y = x$ line signifies that the new architecture has a higher switch-

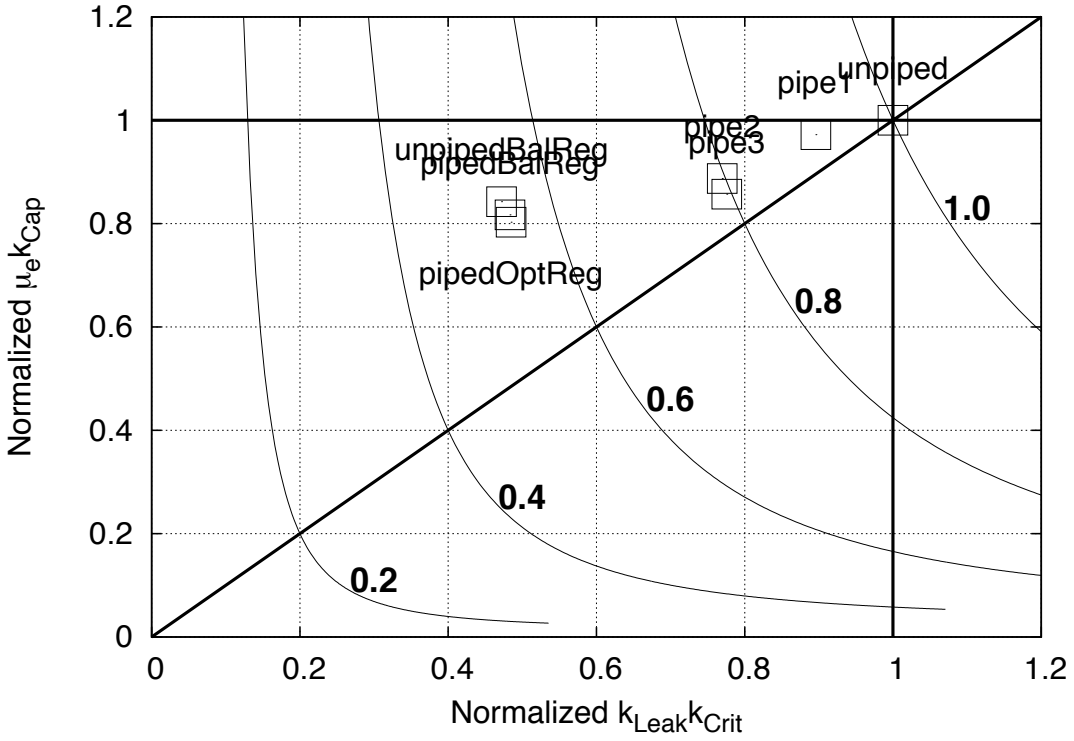


Figure 7.11: Pipeline comparison based on normalized kFactors with normalized equal energy contours over-plotted.

ing/leakage energy ratio when compared to the reference design. Normalized equal energy contours are plotted in the figure to show the change of energy efficiency with the change in circuit parameters. From the figure the effects of incremental changes to the reference design may be visualized.

7.2 Architectural Folding

In this part of the thesis architectural folding for reducing the energy dissipation of the cardiac pacemaker event detector is examined. Initially, both the wavelet filterbank and the GLRT in Figure 7.9 were folded. However, for the wavelet filterbank it turned out that the controller and register overhead were larger than savings achieved by reusing the adders. Consequently, only the GLRT is mapped as a by three and six folded architecture, i.e., the unfolded GLRT in Figure 7.9 is replaced by the architecture in Figure 7.12. In Figure 7.12 folding by three is illustrated. The output $y_1 \cdots y_6$ of the wavelet filterbank is subjected to a block called *Column Multiplier* (CM). This block realizes concurrently the multiplications by $c_{i,i} \cdots c_{i,i+2}$, and holds the products

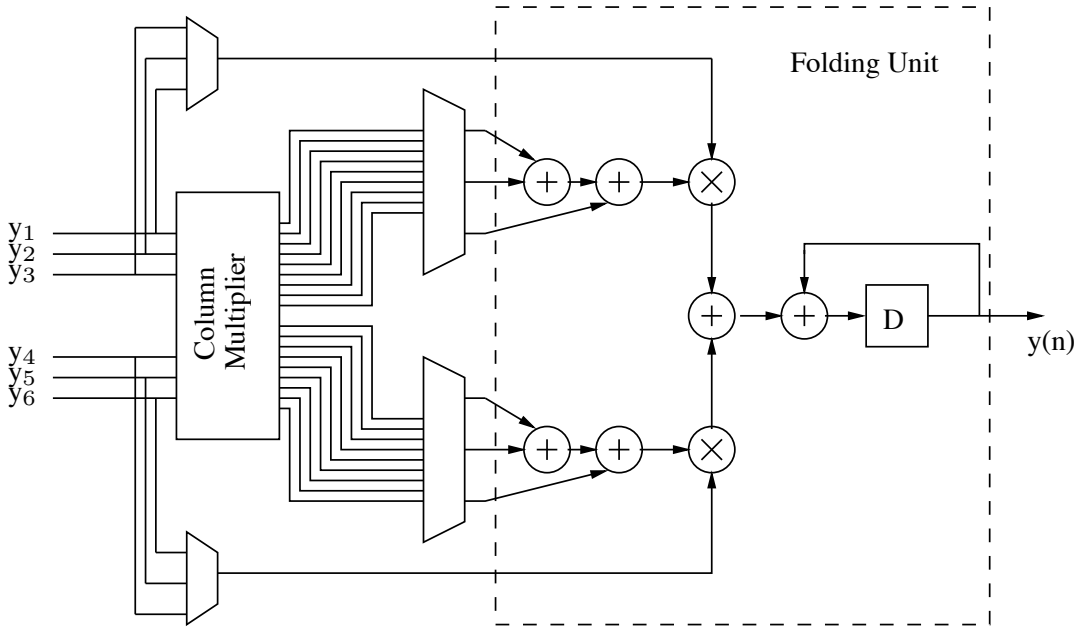


Figure 7.12: Folded by three architecture of the GLRT.

for several clock cycles until processed by the *folding unit*. Folding of CM leads to an area overhead since the coefficients are integer values. The de-multiplexers receive a control signal from a controller and switch the products to the adders, and switches $y_1 \cdots y_6$, correspondingly. The arrangement in Figure 7.12 realizes the folded by three version of the unfolded structure presented in Figures 5.7 and 5.10. The HW cost of the folded architectures are listed with the unfolded realization in Table 7.4. The numbers show clearly the gain in area, i.e., the area cost for GLRT in PF3 and PF6 is reduced by 42 % and 49 % respectively. To maintain throughput, the folded GLRT needs to be clocked three or six times higher than the wavelet filterbank.

Table 7.4: HW cost of a by three (PF3) and six (PF6) folded GLRT.

	Add	Mult	GLRT Area (μm^2)
Unfolded	35	6	6793
PF3	25	2	3912
PF6	21	1	3436

7.3 Architectural folding in Sub-V_T Operation

This section presents the energy dissipation results of the unfolded (UF), and by three (PF3) and six (PF6) folded architectures. Static noise margin (SNM) failure rates are taken into consideration to find an optimal operation point.

Table 7.5 shows the circuit parameters of the synthesized architectures. By employing a higher folding factor, the total gate count of the circuit is reduced. This results in lower leakage energy dissipation for the same operation time. The data is fed to the cardiac event detector at a speed of 1 kHz, and in order to maintain throughput, the GLRT operation frequency in UF, PF3 and PF6 architectures are 1, 3 and 6 kHz, respectively. Figure 7.13 shows the sub-V_T energy dissipation curves for one clock cycle. The continuous lines show the energy dissipation while working at the speed of the critical path, i. e., minimum leakage time, and the dashed lines show the dissipation while working with a fixed clock. The circuits need to be operated at least at a V_{DD} value that meets the requirement on the maximum clock frequency, i.e., 3 and 6 kHz, indicated by the black dots, which are lower than the EMV values. If V_{DD} is raised higher than indicated by the black dots while working at an externally set speed, E_{total} from equation (3.1) will increase (dashed lines). The higher achievable clock frequency at EMV due to a higher V_{DD}, hence lower leakage time, cannot be utilized since the clock speed constraint is external. Thus, if there is an external speed constraint, then, working at a voltage value higher than the value that satisfies the speed requirement results in energy overhead. The only way to operate at the EMV with external speed requirements when the speed requirement is lower than the operating speed and reduce the energy dissipation to a minimum is to employ power-shutdown, which is not investigated in this thesis. It should be noted that although power-shutdown and working at EMV will reduce the energy dissipation, it will introduce energy overhead and a more complicated design process.

Theoretically, the supply voltage of sub-V_T circuits can be reduced down to 50 mV [15], in practice at such voltage values functional failures occur due to the process variations. It was found that the supply voltage value which realizes operation

Table 7.5: Composition properties of the synthesized circuits.

Architecture	k_{cap}	k_{leak}	k_{crit}	EMV (V)
Unfolded	17820	13358	608.051	0.33
PF3	12550	10991	463.259	0.34
PF6	10303	9794	396.805	0.36

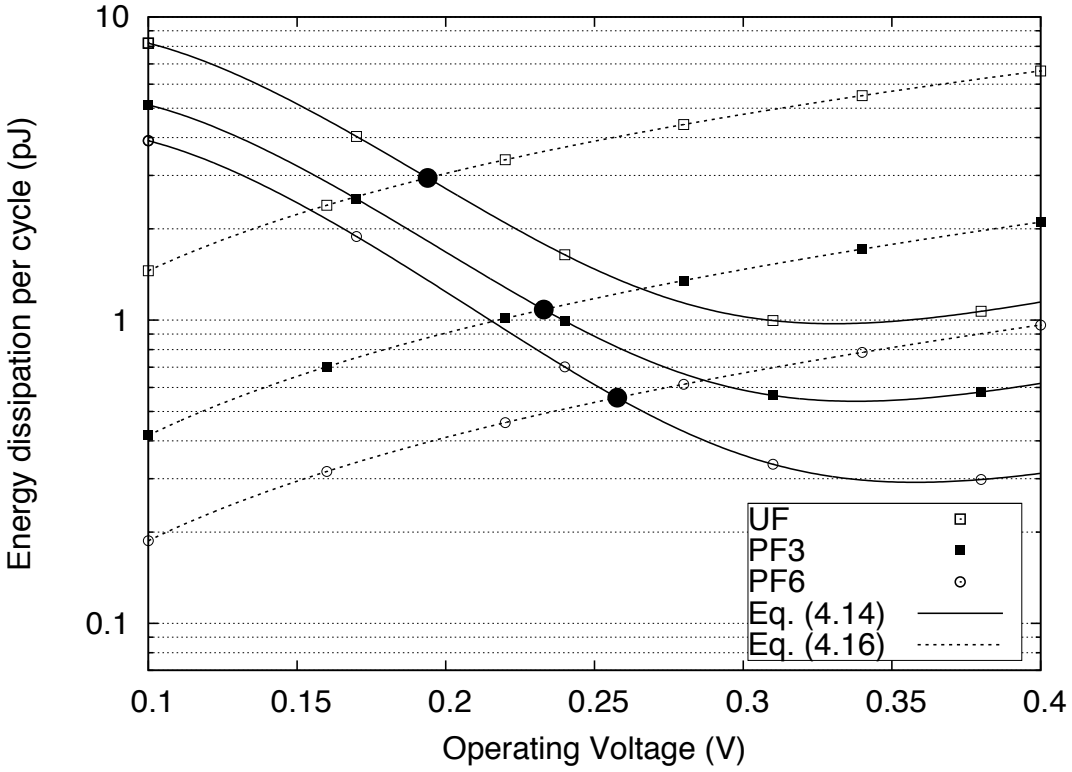


Figure 7.13: Sub-threshold energy dissipation curves of different architectures.

with less than 0.001 failure rate for a 65 nm process is $0.25V$ and this value is taken as the minimum reliable operating voltage (ROV). This results in UF and PF3 operating voltages rising to $0.25V$, causing energy dissipation overhead. The PF6 architecture still operates at $0.26V$ as in Figure 7.13 to satisfy the speed requirement. From now on, the mentioned supply voltages will be taken as the operating voltages of different architectures.

To sustain throughput in a folded architecture, the computation of one sample requires 3 and 6 clock cycles for PF3 and PF6, respectively. Therefore, the switching energy per cycle for the folded architectures are multiplied by their respective folding factors to obtain the switching energy per sample. Moreover, since the idle part of the circuitry leaks during the calculation, the total leakage time of all the architectures is the same and is 1 ms per sample. Thus, it is necessary to multiply leakage energy per clock cycle by the applied folding factor as well. Since the throughput is an external speed constraint, all the architectures process the data at the same amount of time. Gate count reduction minimizes leakage energy, and hence the average leakage scaling factor (k_{leak}) of the circuit. Figure 7.14 shows the energy dissipation components of

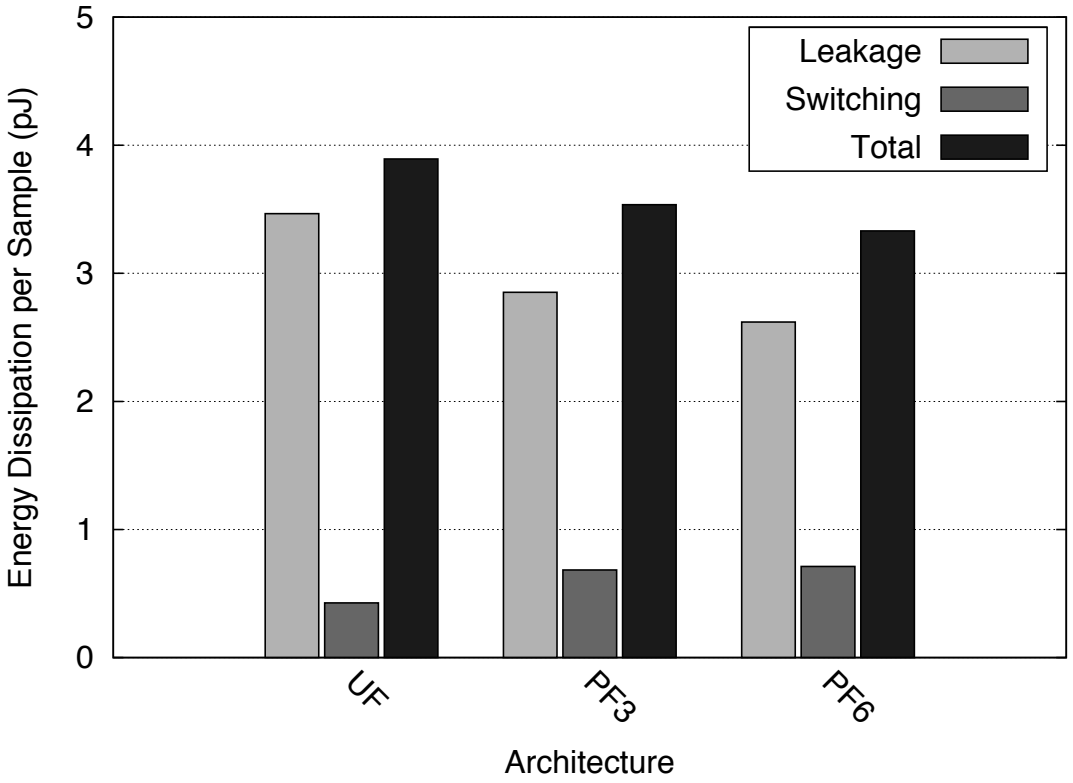
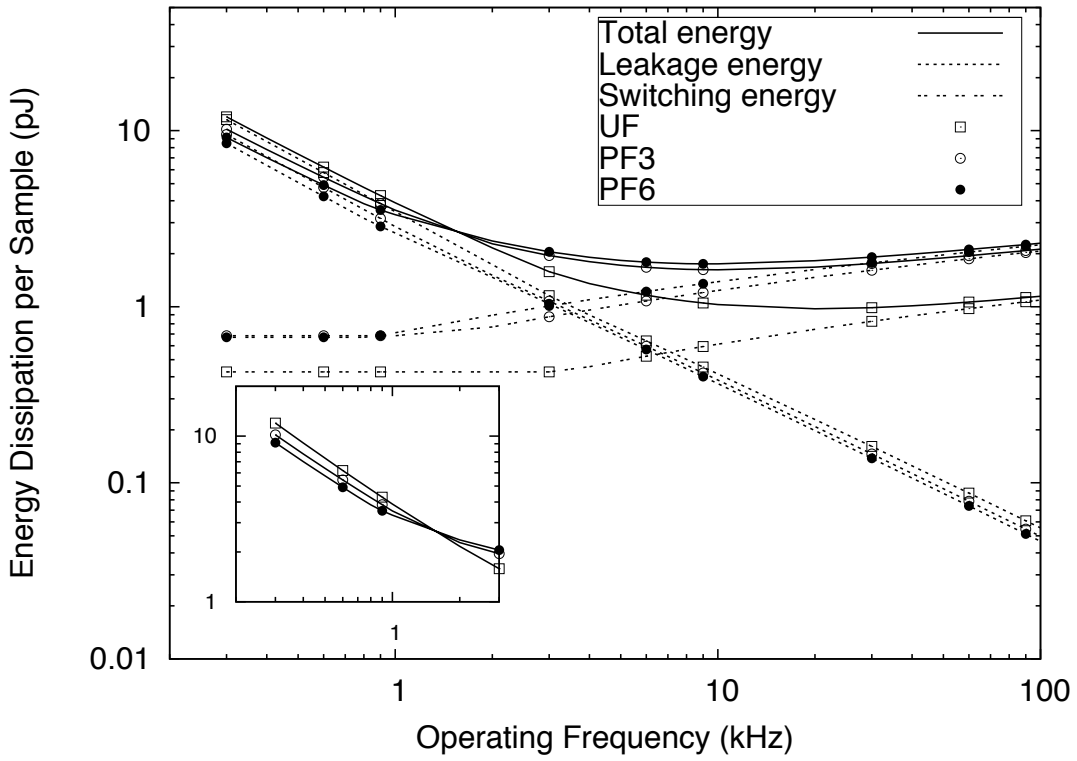


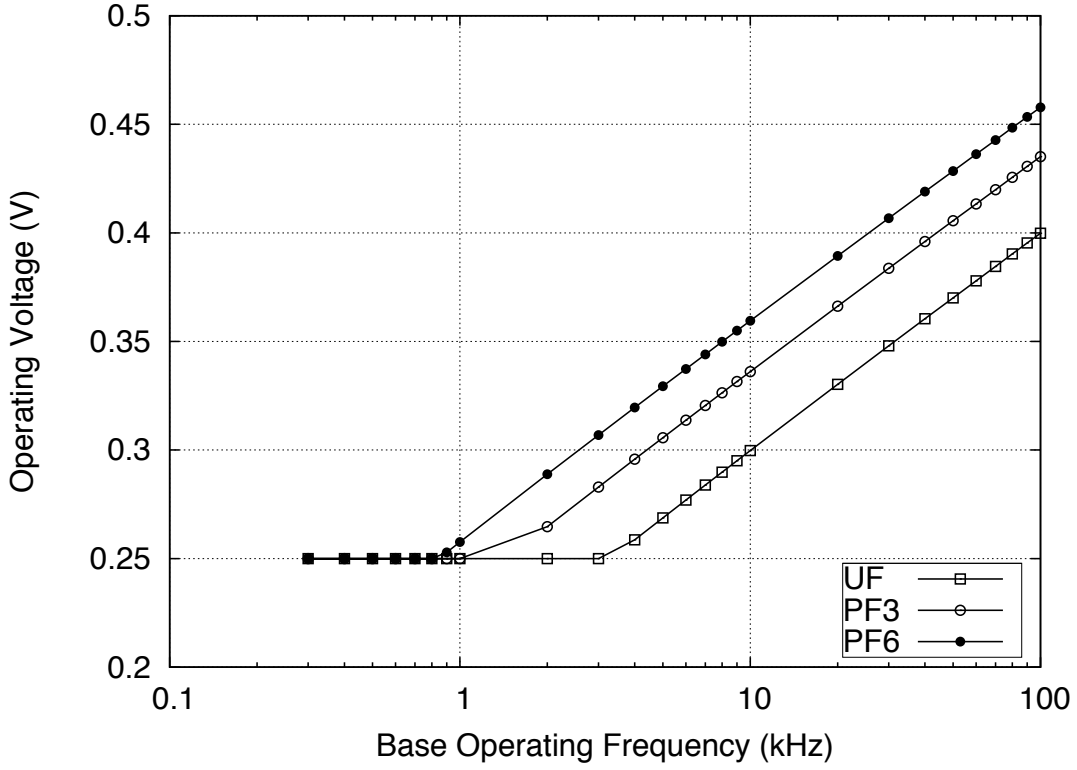
Figure 7.14: Energy dissipation components of different architectures.

the designed architectures per sample. Since all circuits need to be supplied with a voltage lower than EMV, they will operate in the leakage dominated region. From Figure 7.14, it is seen that by increasing the folding factor, the switching energy increases. This is due to the increase in the complexity of the control circuit. However, although the switching energy increases, it is offset by the reduction in the leakage energy, reducing the overall energy dissipation. By going from the UF architecture to the PF6 architecture, the overall energy dissipation per sample point is reduced by 14.4%.

Figure 7.15(a) shows the region where folding is beneficial for energy reduction. Until 1 kHz base frequency, i.e., operation frequency of the UF architecture, folding reduces leakage energy. Above this base frequency, the supply voltage of the folded architectures are increased beyond ROV to satisfy the operating frequency criterion as shown in Figure 7.15(b). This increase in the supply voltage increases both switching and leakage energy dissipation, making the folded architectures less energy efficient than the unfolded one.



(a) Total energy dissipation and its components.



(b) Required operating voltage for a fixed operating frequency.

Figure 7.15: Energy dissipation for the same base processing speed. (a) shows the total energy dissipation with energy components, and (b) shows the required supply voltage for changing base frequency values.

7.4 Conclusions

In this chapter architectural hardware optimizations to reduce energy dissipation are presented. Using the sub-threshold energy model, the effects of architectural improvements on the sub-threshold circuit operation for synchronous and asynchronous cases have been investigated. It was shown that the parallel processing in the sub-threshold domain does not reduce the energy per operation and may only be employed to increase the throughput of a design. On the other hand, by employing pipelining, especially in the asynchronous operation, substantial energy reductions may be realized.

Architectural folding of the wavelet based cardiac event detector is also presented. It is shown that the total area in the most optimized architecture is reduced by 31%, which results in corresponding leakage reduction. Thereby, energy dissipation is reduced by 14.4%. The switching energy due to controller and register overhead increases by folding, but the total leakage reduction offsets this increase in energy dissipation. The operating voltage, which satisfies both speed and failure rate requirement, is determined as 0.26 V, where the circuit dissipates 3.3 pJ per sample.

Chapter 8

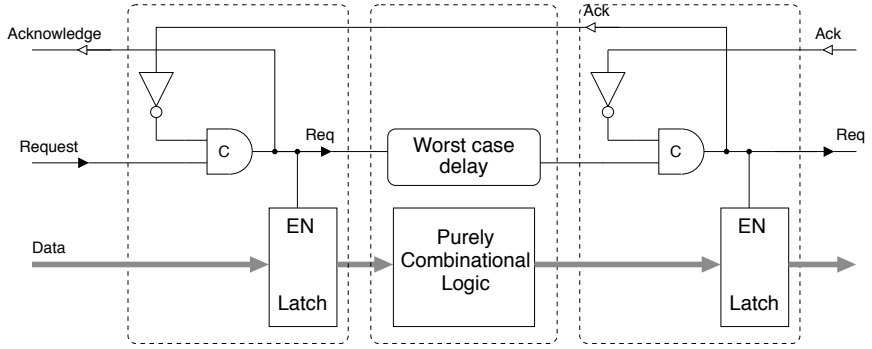
Current Sensing Completion Detection

In Chapter 4 advantages of asynchronous operation to reduce energy dissipation of sub-threshold circuits is presented. In this chapter we present a current-sensing completion detection method to realize average-case performance for single-rail digital CMOS circuits. The technique presented is based on modulating the delay of the control signals according to the actual time it takes for the data to be processed. The circuit examples and the simulation results presented in this chapter are implemented using a commercial standard CMOS 0.18 μm process.

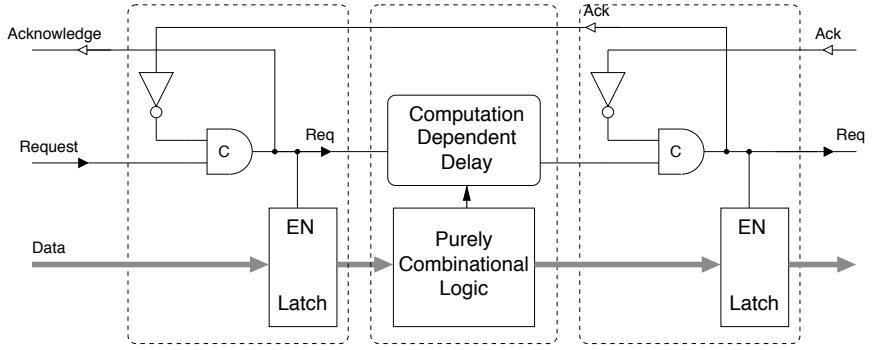
8.1 Modulated Matched Delay

Let us consider a 4-phase bundled data asynchronous micro-pipeline shown in Figure 8.1. As mentioned in Chapter 2, the main problem with this configuration when operating in the sub-V_T regime is the matched delay line. Due to process variations, this delay line has to be severely over-constrained, reducing the operating speed, and thereby also directly reducing energy efficiency of a circuit. In order to achieve the maximum energy efficiency, the time a digital circuit leaks in idle mode needs to be reduced. Reduction in leaking time results in lower leakage energy and moves the energy minimum voltage (EMV) to a lower value, thus reducing the switching energy as well.

Instead of using a a fixed delay line, one solution is to detect the completion of the operation. One way to implement a completion detection circuit is to monitor the current drawn by the combinational circuitry. As long as the combinational gates are



(a) Fixed delay implementation



(b) Computation based delay implementation

Figure 8.1: 4-phase bundled data pipeline (After [1]).

switching, dynamic power is consumed. This is detectable through the supply current I_{VDD} of the combinational gates. There are several implementations of completion detection circuits that use current sensing in the literature [58,59,61]. These methods rely on bipolar transistors, and/or resistors with high values, which are not always available in a standard process, or come as a process option with additional cost. The requirements on the bipolar transistors and resistors set practical limits for the detection of current values in the μA-to-mA range.

Figure 8.2 shows a block diagram of the current sensing based completion detection system. The completion detection system consist of an asynchronous finite state machine (AFSM), a completion detection circuit, consisting of pulse generators and an AC-coupled amplifier, and a single PMOS transistor.

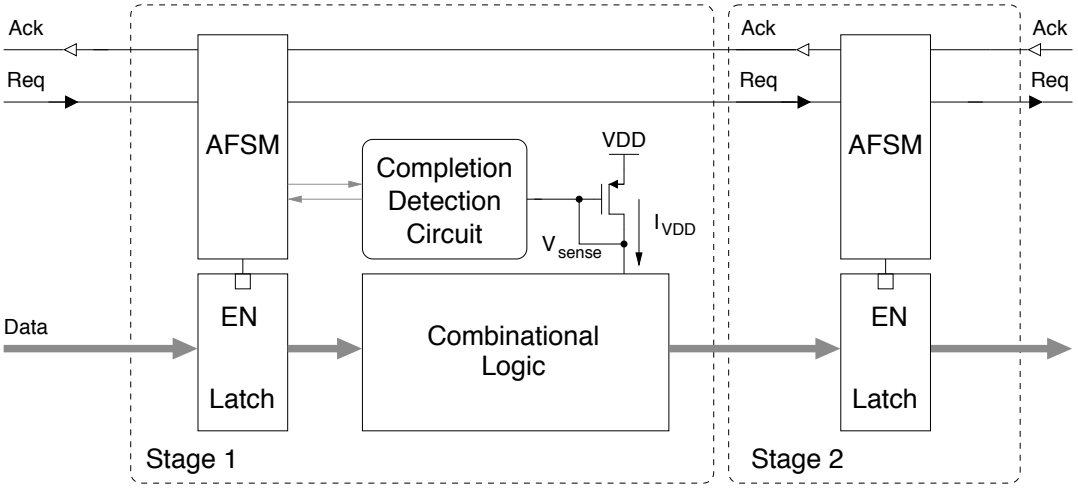


Figure 8.2: General block diagram of the completion detection system. The system consist of an asynchronous finite state machine, completion detection circuitry and a sensor transistor.

8.2 Current Sensing

To detect the operation phase of a circuit by the current sensing completion detection method, instantaneous current consumption of the circuit needs to be monitored. For this purpose, a simple circuit that acts as an ammeter is required. We propose a current sensing technique where the supply node (V_{DD}) of each combinational macro block in the asynchronous micro-pipeline is driven by a diode-connected low V_T PMOS transistor, see Figure 8.3. The current signal is sensed by the diode-connected

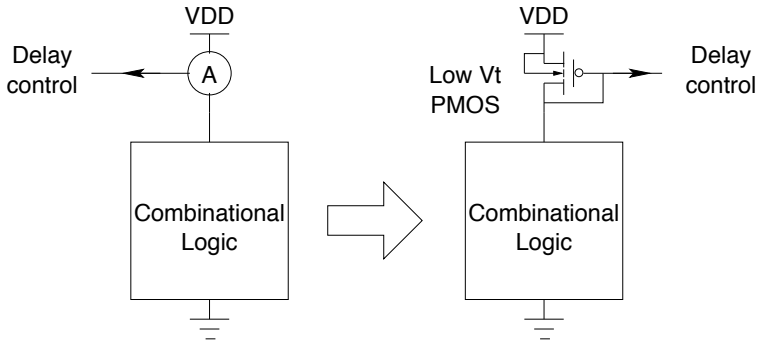


Figure 8.3: Low V_T PMOS current sensor used to detect the operation of the combinational block.

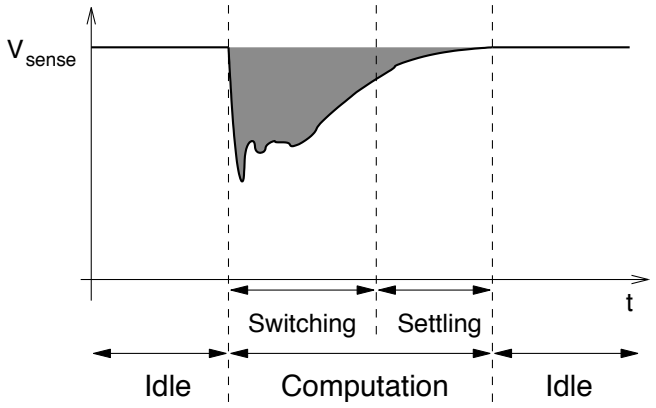


Figure 8.4: Computation signature detected as the temporary drop of the supply voltage at the drain node of the current sensor device. Two main operation regions are shown, computation and idle. The computation frame is divided into switching and settling regions.

low V_T PMOS transistor, converted to a voltage signal, and due to the sub-threshold operation of the MOS transistors, is compressed in the log domain.

While the combinational circuit is idle, i.e. is only *leaking*, the supply voltage is at its sub-V_T nominal value. As soon as the *computation* phase starts the combinational gates begin to draw current from the PMOS transistor and the internal nodes begin switching. The gate-to-source voltage (V_{sense}) of the PMOS transistor changes to accommodate the current change as shown in Figure 8.4. By detecting the change in this current, it is possible to detect the completion of the computation phase.

Although the main idea is simple enough, there are formidable challenges. First of all, as seen in Figure 8.4, there are two operation regions in a sensed signal. The first part is where actual *switching* takes place and is the time we intend to detect. The second part is due to settling of the supply node of the combinational logic block to its nominal value. All internal nodes that will settle to a logic high value follow the settling of the supply node. The settling time depends on the capacitance of the supply node of the logic block, and the resistance of the diode-connected PMOS transistor. In order to avoid excessive computation delay when compared to the standard implementation, the resistance of the diode connected transistor needs to be kept as small as possible. For this reason a low-V_T PMOS transistor with a high width to length ratio is used as the sensing element. Large size of the transistor also increases the capacitance at the supply node, but is negligible when compared to the supply node capacitance of the logic block.

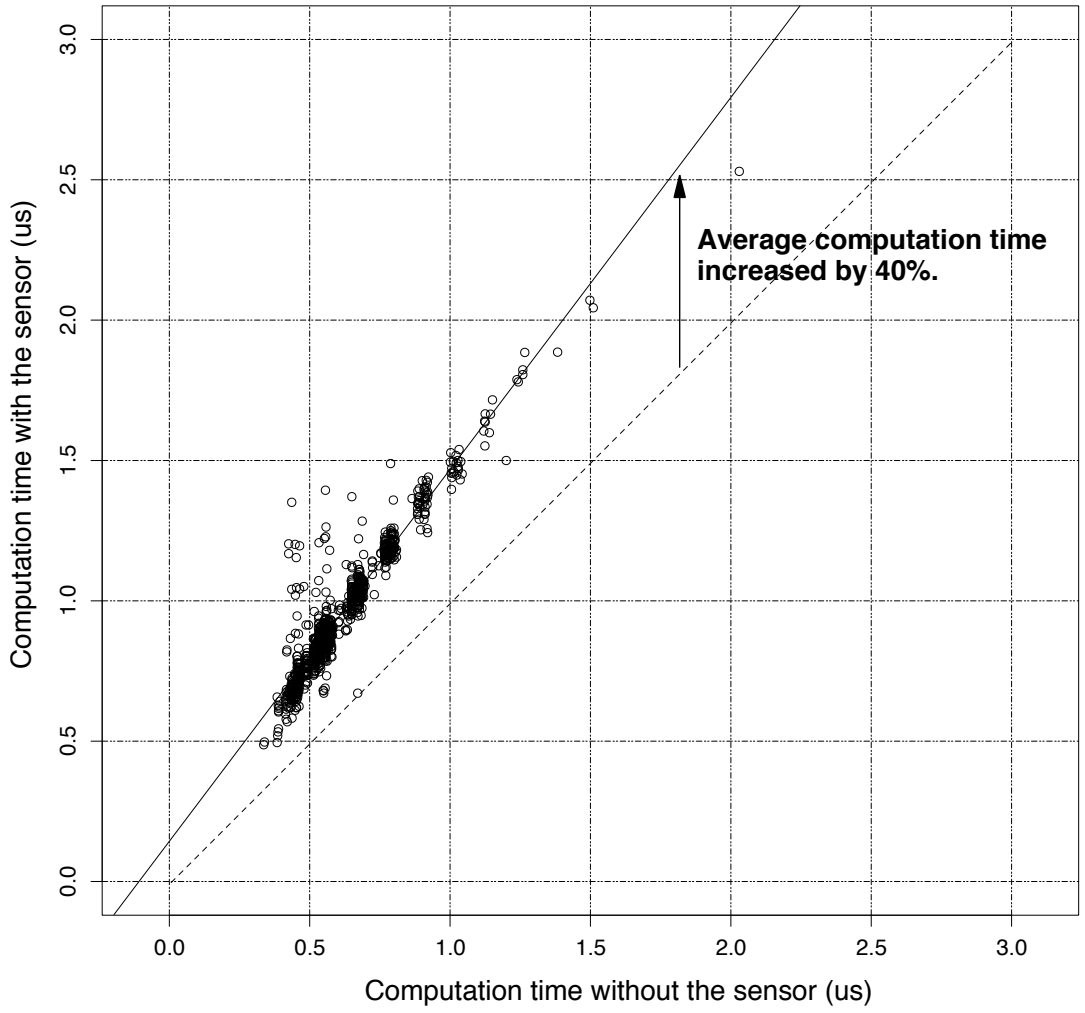
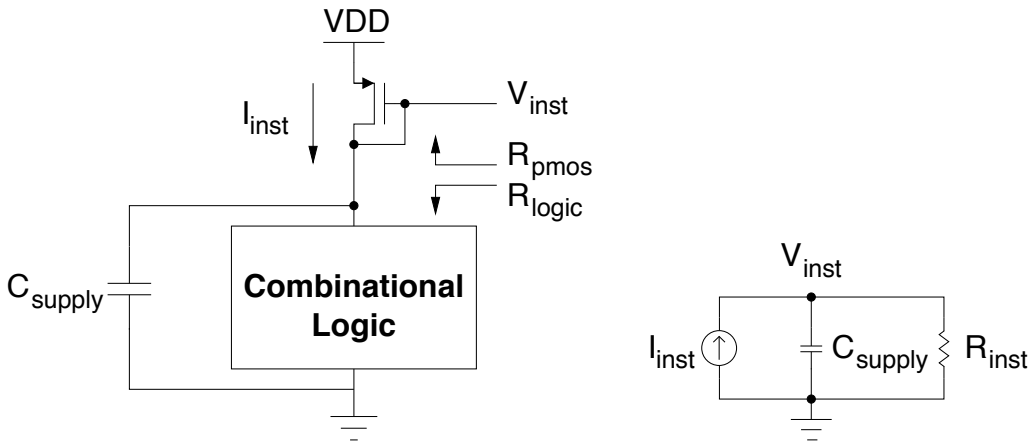


Figure 8.5: Comparison of computation time of a 16-bit adder with and without sensor transistor. Each data point corresponds to a randomly generated input vector.

As explained, the sensor transistor introduces extra delay to the overall computation process. The computation time comparison of an 16-bit ripple carry adder with and without the sensor transistor is shown in Figure 8.5. In the figure the x-axis shows the computation time of the adder block which is directly connected to the supply voltage and the y-axis shows the computation time of the adder which is connected to the sensor transistor. Each individual data point corresponds to a randomly generated input vector, resulting in a spread of computation (completion) times. As it is apparent from the figure, the adder with the sensor transistor takes about 40% more time for computation, on average. To reduce this timing overhead,



(a) Schematic (b) Small signal model

Figure 8.6: Schematic and the small signal model of the signal changes of the supply node.

the sensor transistor's aspect ratio is increased to reduce the resistance of the sensor transistor for faster operation and faster settling of the supply node. One disadvantage of a lower resistance is a lower amplitude of the sensed signal and greater need for amplification.

Another effect of the sensor transistor on the overall operation of the circuit is the low-pass filtering of the sensed current signal. Low-pass filtering of the sensor transistor is illustrated in Figure 8.6. C_{supply} is the total capacitance at the supply node of the combinational logic block, I_{inst} is the instantaneous current supplied by the sensor transistor, V_{inst} is the instantaneous sensed voltage signal, and R_{inst} is the instantaneous resistance seen at the supply node of the logic block. R_{inst} is defined as the parallel combination of R_{pmos} and R_{logic}, i.e, $R_{inst} = R_{pmos}//R_{logic}$, which are the resistance values looking into the PMOS transistor and the logic block, respectively. From the figure, the current to voltage transfer function of the circuit is specified as

$$\frac{V_{inst}}{I_{inst}} = \frac{R_{inst}}{1 + sC_{supply}R_{inst}}. \tag{8.1}$$

Equation (8.1) corresponds to a low-pass (LP) transfer function. The characteristics are controlled by the instantaneous resistance and capacitance at the supply node. The delay overhead due to the sensor transistor during settling is related to the low-pass filtering as well. During the settling phase of the combinational block, R_{logic} acts similar to an open circuit, hence R_{inst} increases in value slowing down the settling

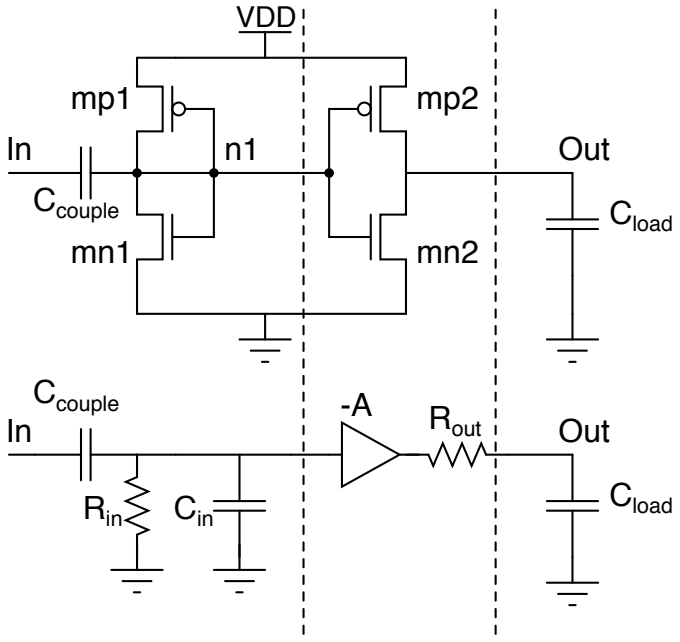


Figure 8.7: Schematic and the small signal model of the AC-coupled amplifier used to amplify the detected signal.

process. Furthermore, R_{pmos} increases as the current supplied by the sensor transistor decreases, which further slows down the settling process.

Another challenge in the implementation of completion detection circuitry involves the magnitude of the signals. The supply voltage in the sub-threshold regime is lower than 400 mV. Consequently, the change in the sensed signal is in the range of a few tens of mVs at most. After sensing the current signal and converting to a log-compressed voltage signal, amplification of this signal is necessary before feeding it into the completion pulse generator. For the amplification of the sensed signal we investigated two class AB CMOS inverter-amplifier topologies, which are explained in the following section.

8.3 Sensed Signal Amplification

One example of investigated AC-coupled amplifiers with its small signal model is presented in Figure 8.7. Diode-connected transistors mp1 and mn1 bias the transistors mp2 and mn2, which are acting as amplifiers, at the maximum gain point for a given size and DC level. In the small signal model C_{in} is the total capacitance at node n1, R_{in} the resistance at the same node, $-A$ the gain of the inverter, R_{out} the output

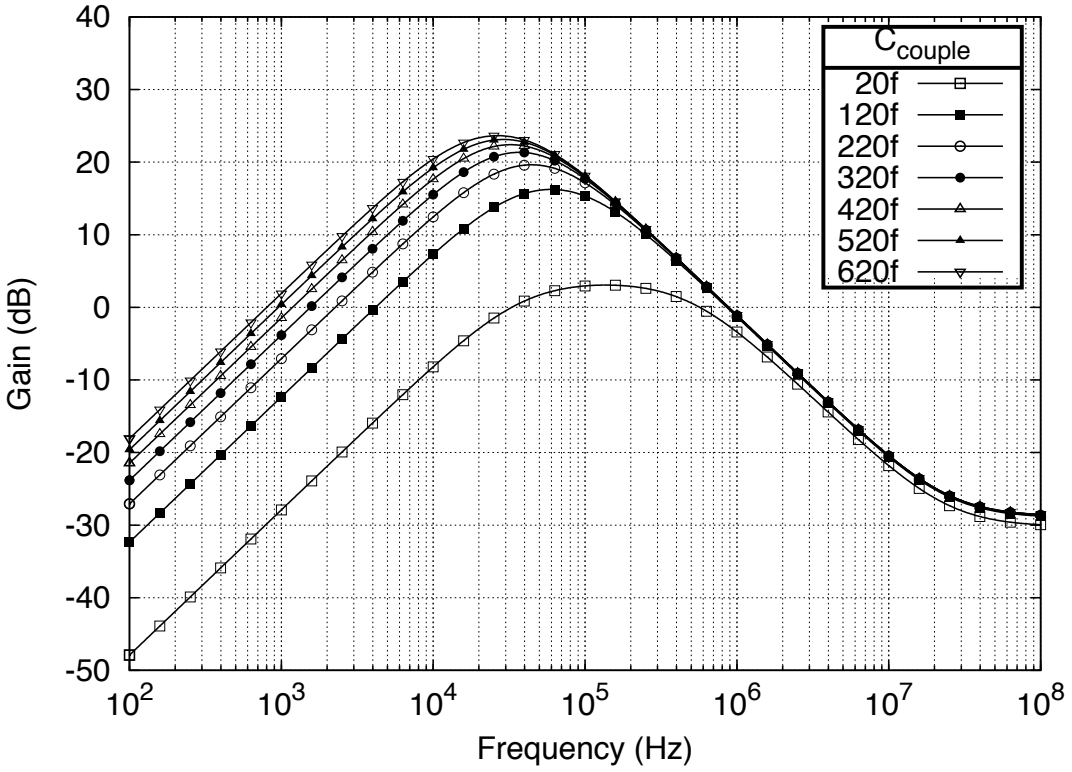


Figure 8.8: Frequency response of the AC-coupled amplifier shown in Figure 8.7 in a 0.18 μm CMOS process.

resistance of the inverter, and C_{load} the total capacitance consisting of the parasitic capacitance of the output inverter (amplifier) and the load capacitance. The input resistance R_{in} is given by

$$R_{in} = \frac{1}{g_{mn1} + g_{mp1}} \tag{8.2}$$

where g_{mn1} and g_{mp1} are the transconductances of the transistors mn1 and mn2, respectively.

$$\frac{V_{out}(s)}{V_{in}(s)} = \underbrace{\frac{sC_{couple}R_{in}}{1 + sR_{in}(C_{in} + C_{couple})}}_{\text{HP Response}} \cdot \underbrace{\frac{-A}{1 + sC_{load}R_{out}}}_{\text{LP Response}} \tag{8.3}$$

The transfer function of the amplifier is given by (8.3). As seen from the transfer function, the AC-coupled amplifier shows a band-pass (BP) response. The high-pass (HP) characteristic is due to the coupling capacitor and the input stage, while low-pass (LP) response with gain is the characteristic of the output stage. The important

Table 8.1: Maximum gain (dB) and lower and upper 3 dB cutoff frequencies (kHz) of the first AC-coupled amplifier for various values of C_{couple} and C_{load}.

		C_{load} (fF)								
		5			10			15		
		Gain	f_1	f_2	Gain	f_1	f_2	Gain	f_1	f_2
C_{couple} (fF)	20	7.1	51.70	619.74	3.1	33.34	561.21	0.3	24.65	535.95
	70	16.0	38.54	266.59	12.7	26.91	223.05	10.3	20.80	203.79
	120	19.1	31.45	194.57	16.2	22.95	155.73	14.1	18.23	138.48
	170	20.8	26.82	162.41	18.3	20.19	126.09	16.3	16.34	110.00
	220	21.9	23.51	143.89	19.6	18.10	109.15	17.9	14.88	93.82
	270	22.6	20.99	131.70	20.6	16.47	98.09	19.0	13.69	83.30
	320	23.2	18.99	123.05	21.3	15.13	90.24	19.8	12.71	75.87

parameters while designing the amplifier and optimizing the whole completion detection system (CDS) are the coupling capacitance, parasitic capacitance at node n1, the load capacitance and the gain of the amplifier. By changing the coupling capacitance, the first pole location and the amplitude of the signal at the input of the output amplifier may be modified. The frequency domain response of the amplifier for multiple C_{couple} values is shown in Figure 8.8. The main constraint limiting the coupling capacitance selection value is the available chip area. The load capacitance, affecting the low-pass response of the amplifier is as important as the coupling capacitance, and needs to be kept as small as possible not to degrade the frequency response of the amplifier. Maximum gain in dB, lower 3 dB cutoff frequency f_1, and upper 3 dB cutoff frequency f_2 of the AC-coupled amplifier for multiple values of C_{couple} and C_{load} are given in Table 8.1. It may be observed that with increasing load capacitance value, both the maximum gain and bandwidth of the amplifier degrades. Thus, when designing a CDS, the capacitive load of the AC-coupled amplifier is kept to a minimum. We designed the AC-coupled amplifier in a standard 0.18 μm process and the sizing of the transistors are presented in Table 8.2.

Sizing of the AC-Coupled amplifier transistors depend on the size of the sensor transistor as well. By changing the amplifier transistor sizes, the frequency response of the amplifier may be changed. Furthermore, a trade off between the gain required from the AC-coupled amplifier and the delay caused by the sensor transistor does exist. The sizing optimization depends on the implementation characteristics of the logic block. If lower timing overhead is required for lower energy operation, the sensor transistor's aspect ratio may be increased to reduce the voltage drop (sensed signal) on the sensor transistor. Due to a lower amplitude of the sensed signal, more gain is

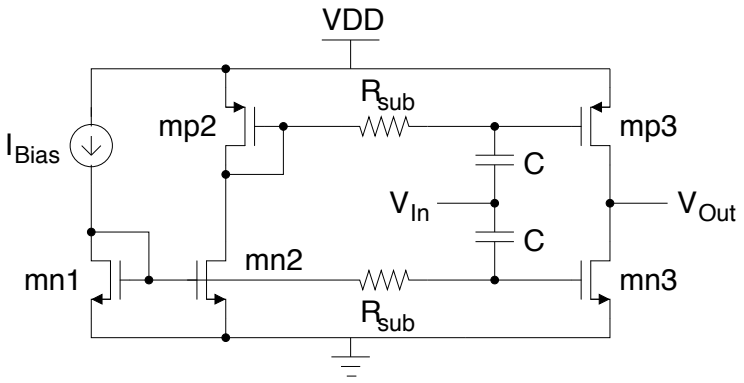


Figure 8.9: AC-coupled amplifier employing sub-threshold MOS resistances for biasing.

required in the AC-coupled amplifier before feeding the amplified signal to the pulse generation block. This results in higher power consumption. On the other hand, if greater delay caused by the sensor transistor (larger spikes in the supply node of the combinational logic block) can be tolerated, the gain, and thus the power consumption of the AC-coupled amplifier may be reduced.

A second AC-coupled amplifier that is suitable for sub-threshold operation is shown in Figure 8.9. This architecture was proposed in [83] by Bikumandla et al. The operation principle of this circuit is similar to the AC-coupled amplifier in Figure 8.7. Transistors mp3 and mn3 act as amplifiers. The biasing network of the circuit consists of transistors mn1, mn2 and mp2. MOS transistors operating in the sub-threshold regime, i.e., R_{sub}, act as high-valued resistors. The sensed signal by the low-V_T transistor is applied to the amplifier transistors mp3 and mn3 through the input coupling capacitances and is then AC amplified by the amplifier transistors. This AC amplified signal is then used to control the pulse-width of the generated completion

Table 8.2: Transistor sizes of the AC-coupled amplifier shown in Figure 8.7 (0.18 μm CMOS process).

Transistor	W (μm)	L (μm)
mn1	0.30	0.18
mp1	0.90	0.18
mn2	0.30	0.18
mp2	0.90	0.18

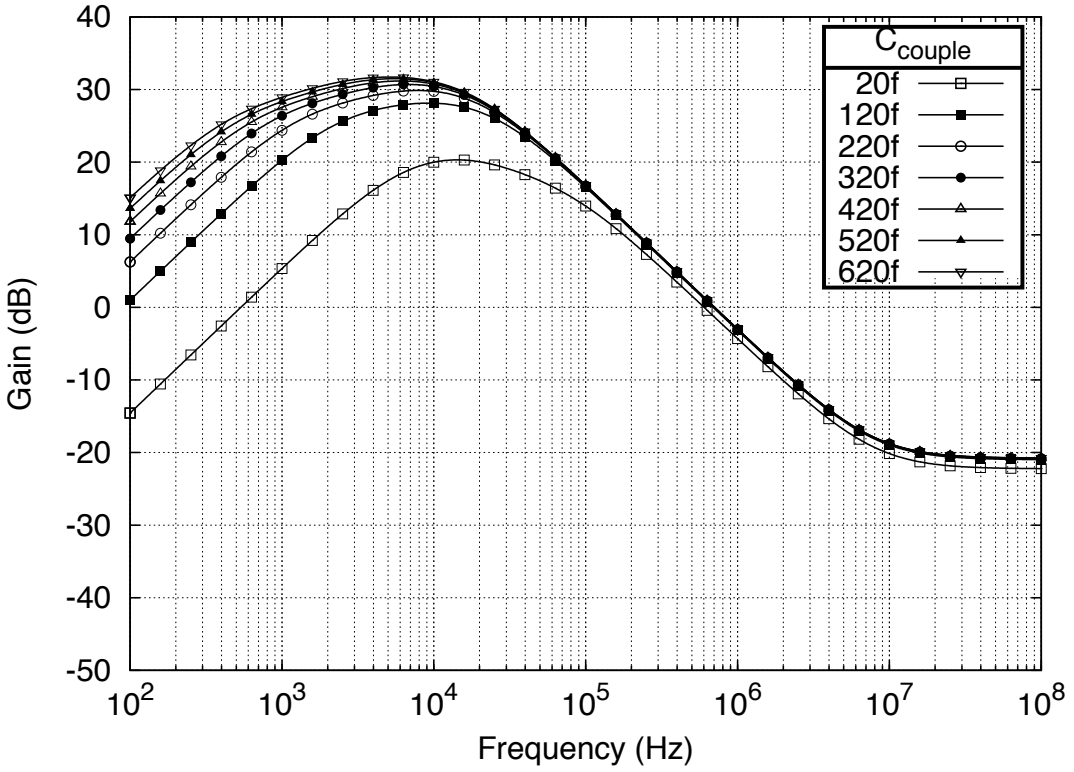


Figure 8.10: Frequency response of the AC-coupled amplifier shown in Figure 8.9.

Table 8.3: Maximum gain (dB), lower and upper 3 dB cutoff frequencies (kHz) of the second AC-coupled amplifier for various values of C_{couple} and C_{load}.

		C_{load} (fF)								
		5			10			15		
		Gain	f_1	f_2	Gain	f_1	f_2	Gain	f_1	f_2
C_{couple} (fF)	20	21.3	5.55	69.03	20.3	4.88	51.46	19.4	4.37	42.53
	70	27.0	3.25	47.85	26.2	2.87	35.42	25.6	2.60	28.45
	120	28.8	2.57	40.20	28.1	2.24	30.07	27.6	2.01	24.41
	170	29.8	2.20	36.12	29.2	1.91	26.94	28.7	1.71	21.92
	220	30.4	1.93	33.61	29.9	1.69	24.93	29.4	1.52	20.24
	270	30.8	1.73	31.91	30.3	1.53	23.54	29.9	1.38	19.05
	320	31.1	1.57	30.68	30.7	1.40	22.52	30.3	1.27	18.16

detection signal, similar to the previous AC-coupled amplifier. Frequency responses of the amplifier are presented in Figure 8.10.

The sub-threshold MOS resistor amplifier is also implemented in a standard CMOS 0.18 μm process. Transistor sizes are presented in Table 8.4.

Table 8.4: Transistor sizes of the AC-coupled amplifier shown in Figure 8.9 (0.18 μm CMOS process).

Transistor	W (μm)	L (μm)
mn1	1.00	0.30
mn2	1.00	0.30
mp2	3.00	0.30
mn3	1.00	0.30
mp3	3.00	0.30
R_{sub} (N)	0.24	0.18

8.3.1 Comparison of AC-Coupled Amplifiers

When the frequency responses of the two AC-coupled amplifier implementations are compared, i.e., Figures 8.8 and 8.10, it may be seen that the implementation with sub-threshold resistor transistors have higher maximum gain for the same coupling capacitance values. Another advantage of the second implementation is that for the lower frequency band, i.e., up to 1 kHz, second implementation has positive gain while the first implementation cannot realize any gain in the same range. This means for very slow changing signals, using the sub-V_T resistor biases implementation is crucial for obtaining gain. Both amplifiers are compared in Table 8.5 for their gains and frequency responses while driving a load of 5 fF with various coupling capacitance values. For all the C_{couple} values sub-V_T resistor biased amplifier has higher gain and lower 3 dB cutoff frequency, which makes it a better choice for very low speed and voltage applications.

Table 8.5: Maximum gain (dB) and lower and upper 3 dB cutoff frequencies (kHz) of the designed AC-coupled amplifiers for various values of C_{couple} and a load capacitance of 5 fF.

		Simple			Sub-V_T		
		Gain	f_1	f_2	Gain	f_1	f_2
C_{couple} (fF)	20	7.1	51.70	619.74	21.3	5.55	69.03
	70	16.0	38.54	266.59	27.0	3.25	47.85
	120	19.1	31.45	194.57	28.8	2.57	40.20
	170	20.8	26.82	162.41	29.8	2.20	36.12
	220	21.9	23.51	143.89	30.4	1.93	33.61
	270	22.6	20.99	131.70	30.8	1.73	31.91
	320	23.2	18.99	123.05	31.1	1.57	30.68

Table 8.6: Power consumption of the AC-Coupled amplifiers at $V_{DD} = 0.4\,V$.

Circuit	Power consumption (nW)	Drawn current (nA)
Simple	0.99	2.48
Sub-V_T MOS Resistors	1.47	3.69

One disadvantage of the implementation shown in Figure 8.9 is a higher current consumption as shown in Table 8.6. It may be noted that for the sub-V_T MOS resistor biased implementation, 1.30 nA out of the total current consumption is used for biasing the transistor mn1. In a system with multiple AC-coupled amplifiers this consumption is shared by multiple instances of the AC-coupled amplifier, reducing the total current drawn by the amplifier blocks.

8.4 Completion Signal Generation

In static CMOS digital circuits, the current drawn by a combinational block depends entirely on the input data switching probability for a fixed circuit structure. Sensing the instantaneous current consumption information, amplifying it and feeding this signal to a circuit that converts this information to a timing signal realizes completion detection operation. However, in the extreme case, where the input data does not change for two consecutive cycles, there will be no dynamic current drawn, and

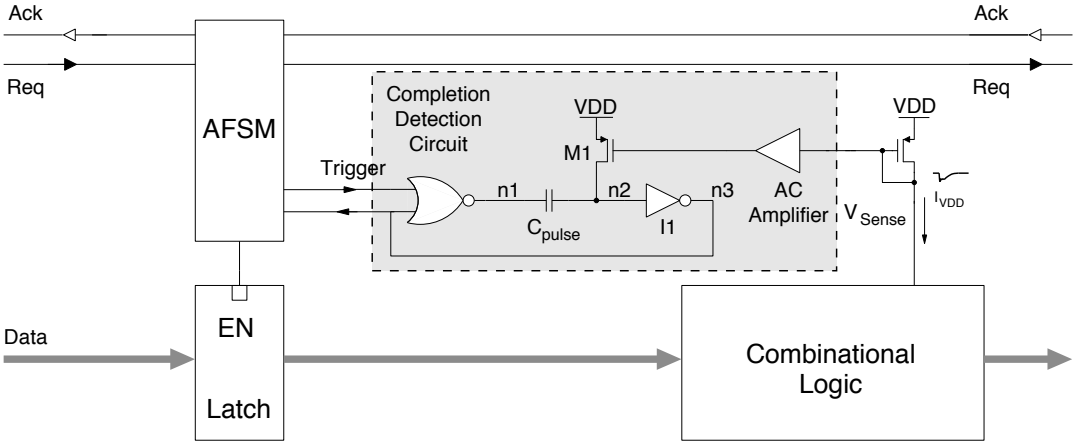


Figure 8.11: Principle of the completion detection system using a triggering circuit and a variable pulse generator.

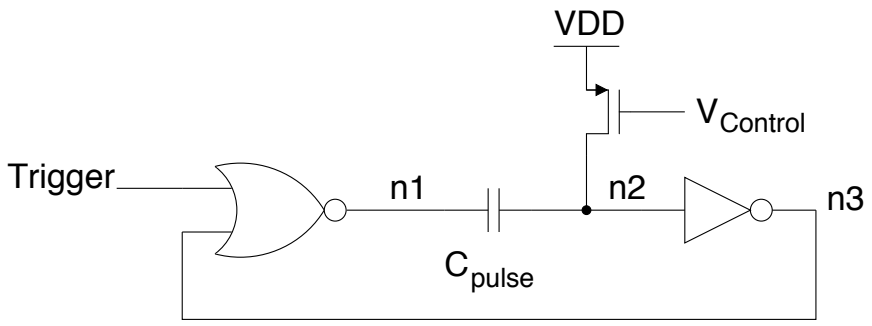


Figure 8.12: Basic monostable multivibrator with variable resistor.

the system will not be able to detect a change in the current. For this reason the completion detection system must provide an automatic *timeout* feature.

We address this problem by using a circuit with inherent *timeout* feature that consists of a trigger circuit and a variable length pulse generator as shown in Figure 8.11. We implemented the variable pulse generator as a monostable multivibrator as shown in Figure 8.12 [84]. In this monostable multivibrator implementation, a PMOS transistor acts as a variable resistor, whose resistance is modulated by the amplified sensed current signal. By modulating the instantaneous resistance of the PMOS transistor, the RC time constant of the multivibrator is modulated as well, resulting in a pulse whose width is proportional to the area under the current curve of the combinational block. The pulse width of the variable pulse generator is given as

$$T = C(R + R_{on}) \ln \left[\frac{R}{R + R_{on}} \frac{V_{DD}}{V_{DD} - V_{th}} \right], \tag{8.4}$$

where R is the average resistance of the PMOS transistor during pulse generation, R_{on} the resistance of the NOR gate, and V_{th} the switching threshold of the inverter. Assuming $R_{on} << R$ and $V_{th} = V_{DD}/2$, equation (8.4) is simplified to $T = 0.69RC$. This means, the generated pulse is compressed by a factor of 0.69 when compared to the actual computation time in the ideal case.

The trigger circuit inside the asynchronous finite state machine (AFSM) is activated with the arrival of new data to the combinational block, see Figure 8.11. This starts the pulse generator by setting nodes n1 and n2 to logic-0. At this point, the PMOS (M1), which acts as a variable resistor, starts charging node n2 until this node reaches the inversion threshold of the inverter (I1). This causes the node n3 to switch finalizing the completion detection pulse generation. The time required to switch node n2 is dictated by the value of the charging current I_{charge}, and the value

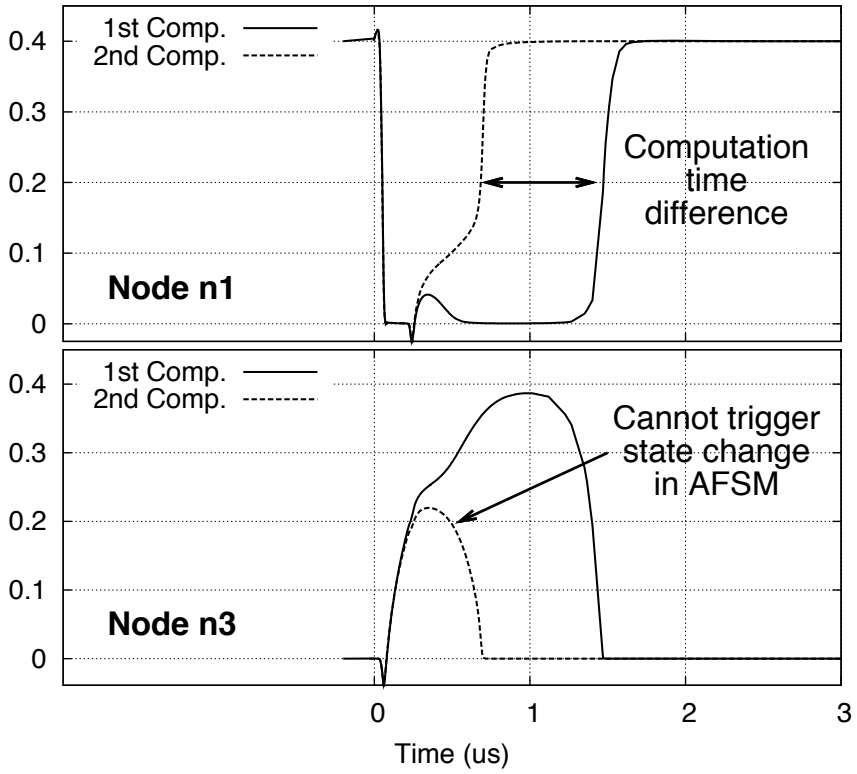
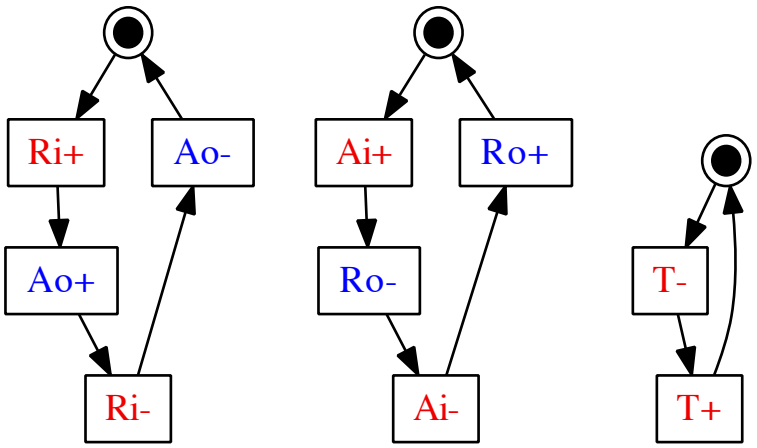


Figure 8.13: Variable pulse generator node voltages. The pulse width of the signal at node n1 is proportional to the control voltage, and hence, to the actual computation completion time of the combinational block.

of the pulse capacitor C_{pulse}. The charging current I_{charge} is inversely proportional to the change in V_{sense}, thereby generating a pulse that directly corresponds to the *switching* time of the combinational logic block. The correct computation completion signal generation is achieved by matching the time required to charge node n2 to the actual computation time.

In a standard implementation of a variable pulse generator, node n3 is used as the output. For some cases, the resistance of the PMOS transistor might be too low. A very fast switching occurs at node n2 and no switching occurs at node n3, resulting in no signal generation. To prevent this situation, the signal at node n1 was used as the output. Figure 8.13 shows the node voltage waveforms of the pulse generator for two different control voltage waveforms. The signal at n1 goes low when the trigger signal goes high, and if there is no change in the signal at n3, it goes high as the trigger signal goes low. This operation guarantees that a completion detection signal is generated for each trigger input, hence the automatic *timeout* feature. The minimum width



(a) Previous stage related.

(b) Following stage related.

(c) Completion detection related.

Figure 8.14: Behavior of the first AFSM at the borders. Signals related to the (a) previous stage, (b) following stage, and (c) CDS are shown.

of the completion detection pulse is thus set by the trigger pulse width, setting the minimum delay of the *Request* signal of the preceding stage.

8.5 Designing Asynchronous Finite State Machines

A finite state machine (FSM) is a behavioral representation of a system composed of a finite number of states. The FSM changes states based on the current state and/or inputs to the system. In synchronous FSMs, the time is discretized by introducing the notion of cycles, which is the time that a system takes for switching from one state to another. In the hardware systems, the notion of cycle is implemented with a periodic signal, *clock*, that defines the time instants that a system changes state [85]. On the other hand, in asynchronous FSMs (AFSMs), due to the lack of a common signal that defines the instants for a state change, asynchronous system changes state only based on the current state and present inputs. In AFSMs, the decisions are affected by input changes without a regulating body, i.e., a clock signal, so all input signals employed for the decision needs to be stable. Due to only input event based state change, the AFSMs are susceptible to glitches, and may end up in wrong states if the glitches at the inputs are not removed. Therefore, synthesis of AFSMs is more involved than their synchronous counterparts.

Different classes of self-timed circuits are described in Chapter 2. While designing the AFSMs for a completion detection system, we employed speed-independent (SI) assumptions. As mentioned, AFSM design is not as straight-forward as synchronous FSM design. Popular hardware description languages, e.g., Verilog and VHDL, and industrial logic synthesis engines do not support design and synthesis of AFSMs. Therefore, an academic tool, i.e., *Petrify*, based on signal transition graphs (STG) is used [86]. An STG is an event-based model that is represented by a directed graph [87]. An STG is also a simplified form of Petri-Nets [88–90]. Example STGs are shown in Figure 8.14. In the STGs, the boxes represent a signal transition. A + and a − represent up and down transitions, respectively. Red labels represent the input signals and blue labels represent the output signals. Large dot inside a circle (place), which is called a token, represents the current state of the system. Except the initial marking related places, all other places are not shown in the STGs.

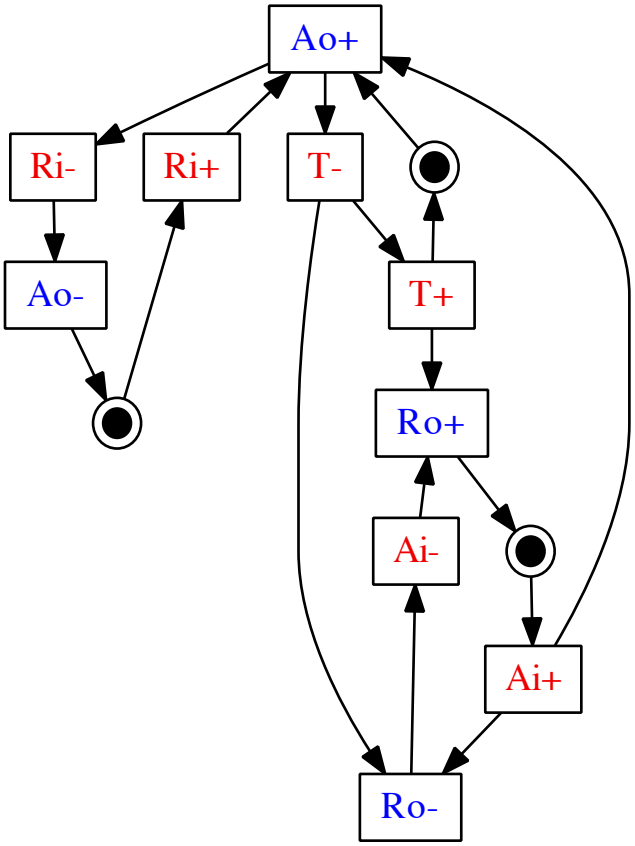
To be able to synthesize and verify the AFSMs with Petrify, they need to be represented in a form that is understood by the tool. The textual description of the AFSM for the completion detection circuit is shown in Listing 8.1. This textual format is used by SIS, a logic synthesis tool developed at UC Berkeley [91]. Petrify first verifies this description works correctly and is synthesizable, and then synthesizes the description into a set of Boolean equations and gate-level netlists.

The asynchronous communication protocol between consecutive pipeline stages of the completion detection circuit is a 4-phase bundled data protocol based on [92]. The AFSM that implements this protocol has been modified to include the completion detection circuit described in Section 8.4. Figure 8.14 shows how the designed circuit behaves at the borders, i.e., with respect to the driving circuit and the driven circuit. In the figure Ri, Ro, Ai and Ao are input request, output request, input acknowledge and output acknowledge signals, respectively. The T is the signal coming from the analog part of the completion detection circuit, i.e., sensing and pulse generation circuitry.

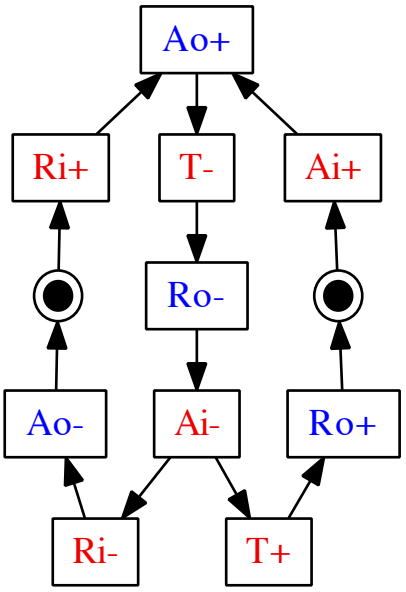
The STG of the textual description in Listing 8.1 is presented in Figure 8.15. This definition is further simplified by specifying the timing assumptions. In the listing, the line

```
.slow T+ Ri-
```

specifies the events T+ and Ri- as slow events. This means that the signals generated by these events are slow enough, such that the internal nodes of the circuit will have settled to their final values before a new event occurs. These specifications are made based on the following observations:



(a) Initial STG



(b) Simplified STG

Figure 8.15: STG for the first version of the completion detection AFSM where all the signals are inter-connected. (a) initial description STG and (b) simplified STG are shown.

```
.model comProt_v10
.inputs Ri Ai T
.outputs Ro Ao
.graph
Ri+ Ao+
Ao+ Ri-
Ri- Ao-
Ao- Ri+
Ro+ Ai+
Ai+ Ro-
Ro- Ai-
Ai- Ro+
Ao+ T-
T- T+ Ro-
T+ Ro+ Ao+
Ai+ Ao+

.marking {<Ao-,Ri+> <T+,Ao+> <Ro+,Ai+>}
.slow T+ Ri-
.end
```

Listing 8.1: Textual representation of the AFSM shown in Figure 8.15

- Event T+ is coming from the completion detection circuitry and the minimum pulse width generated is roughly equivalent to the delay of 20 inverter gates, which is much greater than the nodes take for settling.
- From Figure 8.15a, there is another event between events Ri+ and Ri-, meaning enough settling time for the internal nodes of the circuit.

Following the timing assumption, the simplified version of the starting STG is presented in Figure 8.15b. The simplification is done by Petrify based on the timing assumptions presented.

The circuit generated by Petrify and its gate level implementation with embedded completion detection circuitry is shown in Figure 8.16. The state diagram of the system is given in Appendix D Figure D.1 as reference. After the synthesis of the circuit, extensive HSPICE simulations were run to verify the functionality of the AFSM over process corners as well.

Furthermore, we explored another version of the AFSM to remove the triggering circuit in Figure 8.16. In this implementation, the latch signal is not generated by an autonomous pulse generator as in the previous implementation, but is generated from

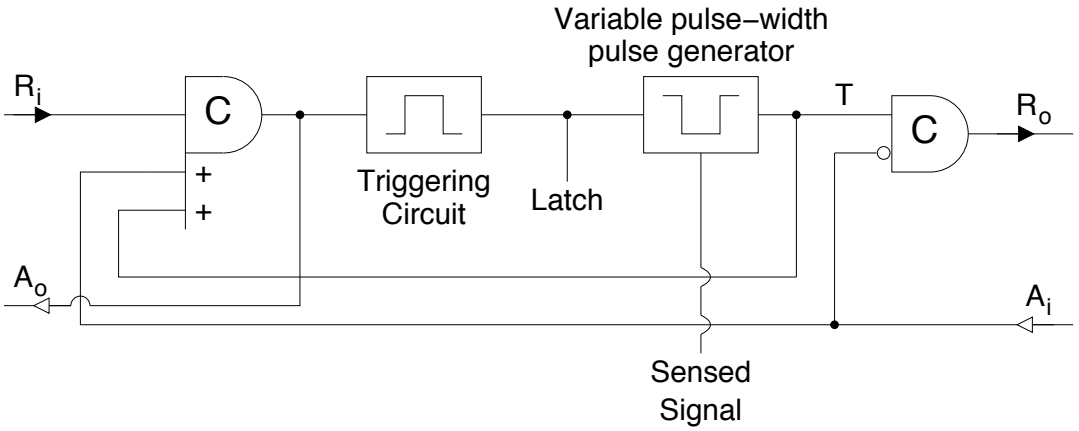
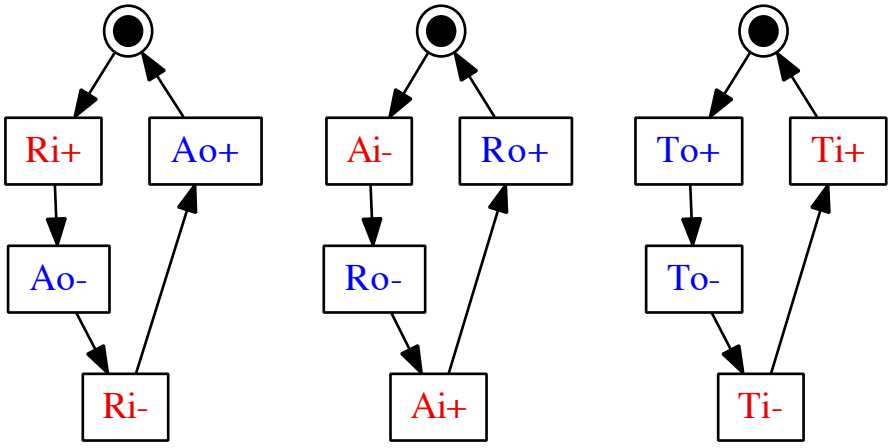


Figure 8.16: AFSM including the pulse generating completion detection circuitry.



(a) Previous stage related.

(b) Following stage related.

(c) Completion detection related.

Figure 8.17: Behavior of the second AFSM at the borders. Signals related to the (a) previous stage, (b) following stage, and (c) CDS are shown.

the state information and the input signals. Figure 8.17 shows how the second version of the AFSM behaves at the borders. In addition to the signals defined in Figure 8.14, two internal signals, i.e., nx and ny, are defined. These signals are required to solve complete state coding (CSC) conflicts. Moreover, from these signals a latch control signal for controlling the positive level latches is generated. The event related to the completion detection circuitry is divided into one input and one output signal, Ti and To, respectively. This is required since the triggering of the variable pulse generator is realized by the AFSM, and the state change of the AFSM is triggered directly by

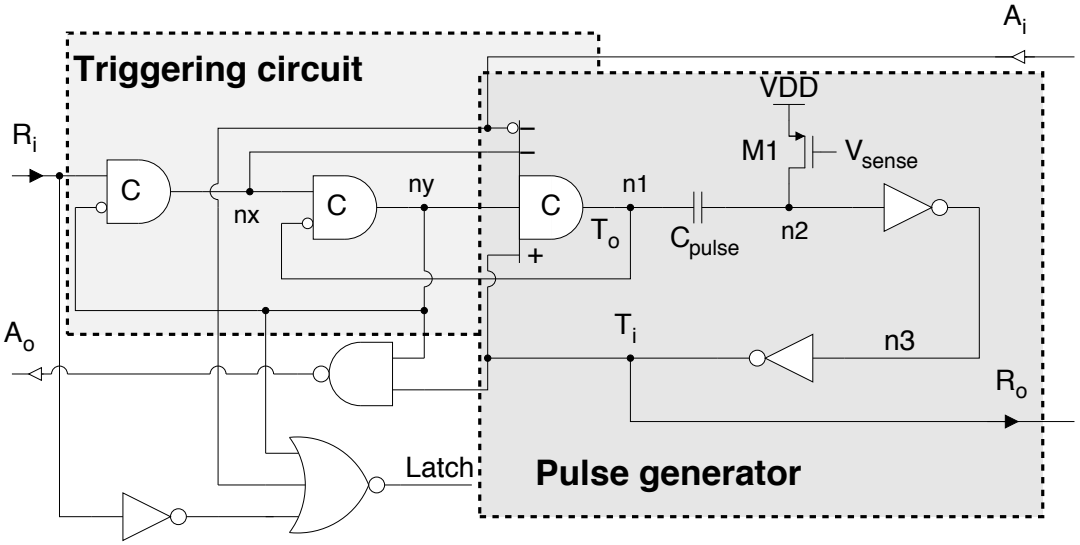
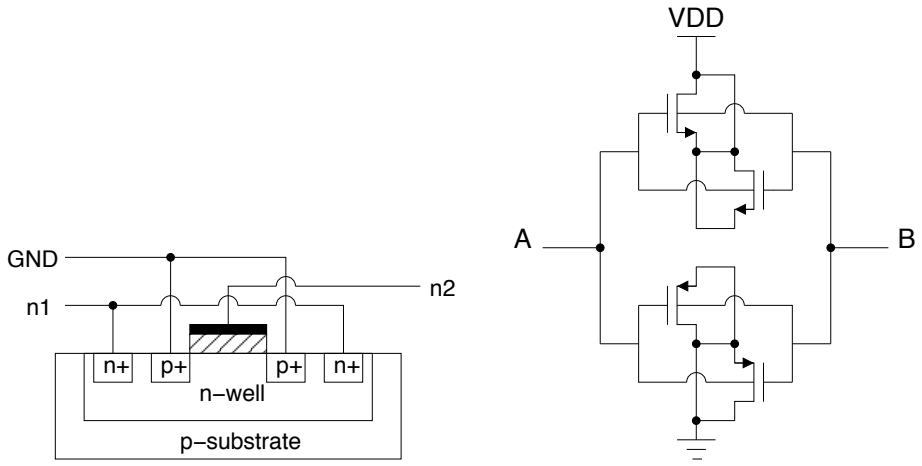


Figure 8.18: Second AFSM including the pulse generating completion detection circuitry. Function of the triggering circuitry in the previous version is embedded inside the AFSM.

the completion detection circuit. Hence, signals related to the completion detection circuit are separated.

The textual description of the second AFSM is given in Appendix D Listing D.1, and its STG is shown in Figure D.2a. Compared to the previous version of the AFSM, STG of the second AFSM is more complex. This is due to added extra signals. A slow operating environment assumption is made during the synthesis of this circuit. With the specified timing assumptions, the simplified STG is shown in Figure D.2b, and the state diagram representation of the STG is presented in Appendix D Figure D.3 for visualizing the state-based operation of the circuit.

We designed and optimized the circuit based on Petrify synthesis results. Circuit equations generated by Petrify was further simplified by merging the function of logic and Muller-C gates. The designed circuit is presented in Figure 8.18. The triggering circuit realization which acts as the autonomous pulse generator in Figure 8.16 and the variable-width pulse generator are emphasized. The triggering circuit is realized using the AFSM design methodology while the pulse generator is again based on the monostable multivibrator circuit. Only change in the variable pulse generator is the replacement of NOR gate in Figure 8.12 with a Muller-C element. Again,



(a) Cross section. (b) Schematic.

Figure 8.19: Parallel compensated depletion mode MOS transistor based capacitor. (a) Cross section of a depletion mode PMOS, and (b) parallel compensated depletion mode MOSCAP schematic are shown.

correct operation of the implemented circuit was verified with HSPICE simulations over process corners and varying input signal timings.

8.5.1 Coupling and Variable Pulse Generator Capacitors

In our implementation of the completion detection circuitry we used metal-insulator-metal (MIM) capacitors as the coupling capacitor that is connected to the AC-coupled amplifier. This choice was made to be able to couple the sensed signal as strongly as possible to the amplifier regardless of process variations. However, we have used MOS-based capacitors to realize the pulse capacitor C_{pulse}. In this way, we tried to compensate for the process-voltage-temperature (PVT) variations of the combinational circuit. Using a MOSFET based capacitor, the capacitor value is made more dependent on the PVT variations, thus affecting the generated pulse-width of the variable pulse generator. Consequently, this improves immunity of the circuit against such PVT variations. One disadvantage of MOS-based capacitors is their non-linearity. In the parallel compensated configuration presented in Figure 8.19 [93], all MOS transistors remain in the depletion mode for all input and output voltage combinations .

To check the behavior of the depletion mode capacitance in the sub-threshold regime, we ran HSPICE simulations. The capacitance values seen between the

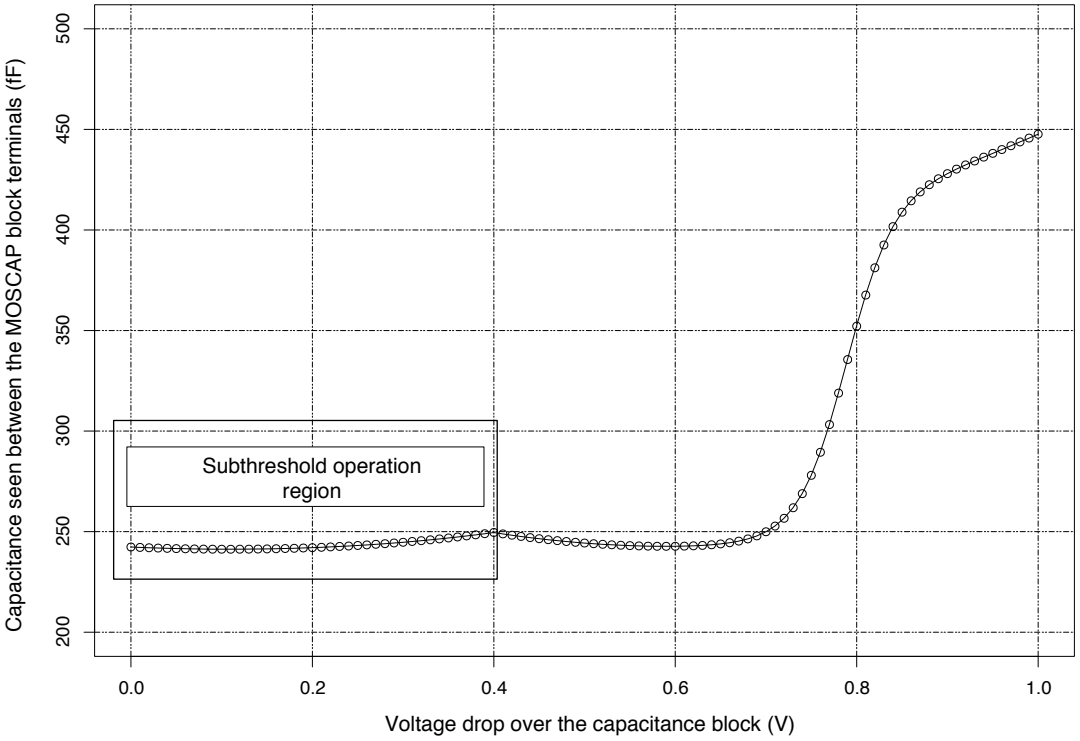


Figure 8.20: Simulation of the MOSCAP capacitor.

MOSCAP terminals for varying voltage drop values is shown in Figure 8.20. Sub-threshold operation region is marked in the figure and as seen, within the operation range of a few hundred mVs this configuration gives a capacitance value that varies less than 4%. While this value is by no means spectacular, for the pulse generation circuit where the current is averaged over time, it is sufficient enough.

8.5.2 Muller-C Element Implementation

One of the main problems of sub-V_T operation is implementing state-holding elements. Such circuits usually depend on a feedback loop to preserve the desired state. High leakage currents in the sub-threshold operating regime may result in state loss in some of these circuits. In asynchronous circuit design, the main state holding primitive is the Muller-C element [94]. The correct operation of the Muller-C element is crucial for any asynchronous circuit. There are several published Muller-C implementations in the literature. We have compared four separate Muller-C implementations operating in the sub-threshold regime against each other: Dynamic and static versions

Table 8.7: Comparison of Muller-C elements operating at $V_{DD} = 0.4\,\mathrm{V}$.

	Dynamic	Martin	Sutherland	Van Berkel
Reference	[95]	[96]	[95]	[97]
Average Current [pA]	31.3	138.9	39.7	36.9
Total Energy [fJ]	12.5	59.5	15.5	14.8
Delay [ns]	63.2	101	67.9	45.4
EDP [pJ.ns]	0.79	6.01	1.05	0.67
State retention	No	Yes	Yes	Yes

suggested in [95], a dynamic implementation with feedback suggested by Martin [96], and the one designed by Van Berkel for low-power operation [97].

All four circuits were compared in terms of their power consumptions, and their operation speeds. The simulation results for a supply voltage of 0.4 V are given in Table 8.7. It may be observed that the dynamic Muller-C element is not able to retain its state at this supply voltage. Of the remaining alternatives the Van Berkel Muller-C element consumes the least amount of power and is also the fastest implementation. The standard cell library of the target technology has been enhanced by implementing the Van Berkel Muller-C element for placement and routing.

8.6 Design Flow

The design flow for CDS implementation is shown in Figure 8.21. The shaded area represents the design flow for the AFSM that includes the completion detection circuitry. Previously explained design of AFSM is integrated with the analog part of the completion detection circuit in this flow. Final design consists of three parts:

- The combinational logic block.
- Completion detection circuit with AFSM.
- Latches/registers that separate combinational pipeline stages.

Combinational circuits are generated following a standard synthesis flow. The functionality is described using hardware description languages and synthesized into a gate level netlist by Synopsys Design Compiler using a standard cell library supplied by the foundry. All logic blocks are designed such that they only consist of combinational gates. The synthesized circuit is placed and routed in Cadence SoC Encounter, a standard back-end tool, and the combinational block Graphic Data System (GDSII)

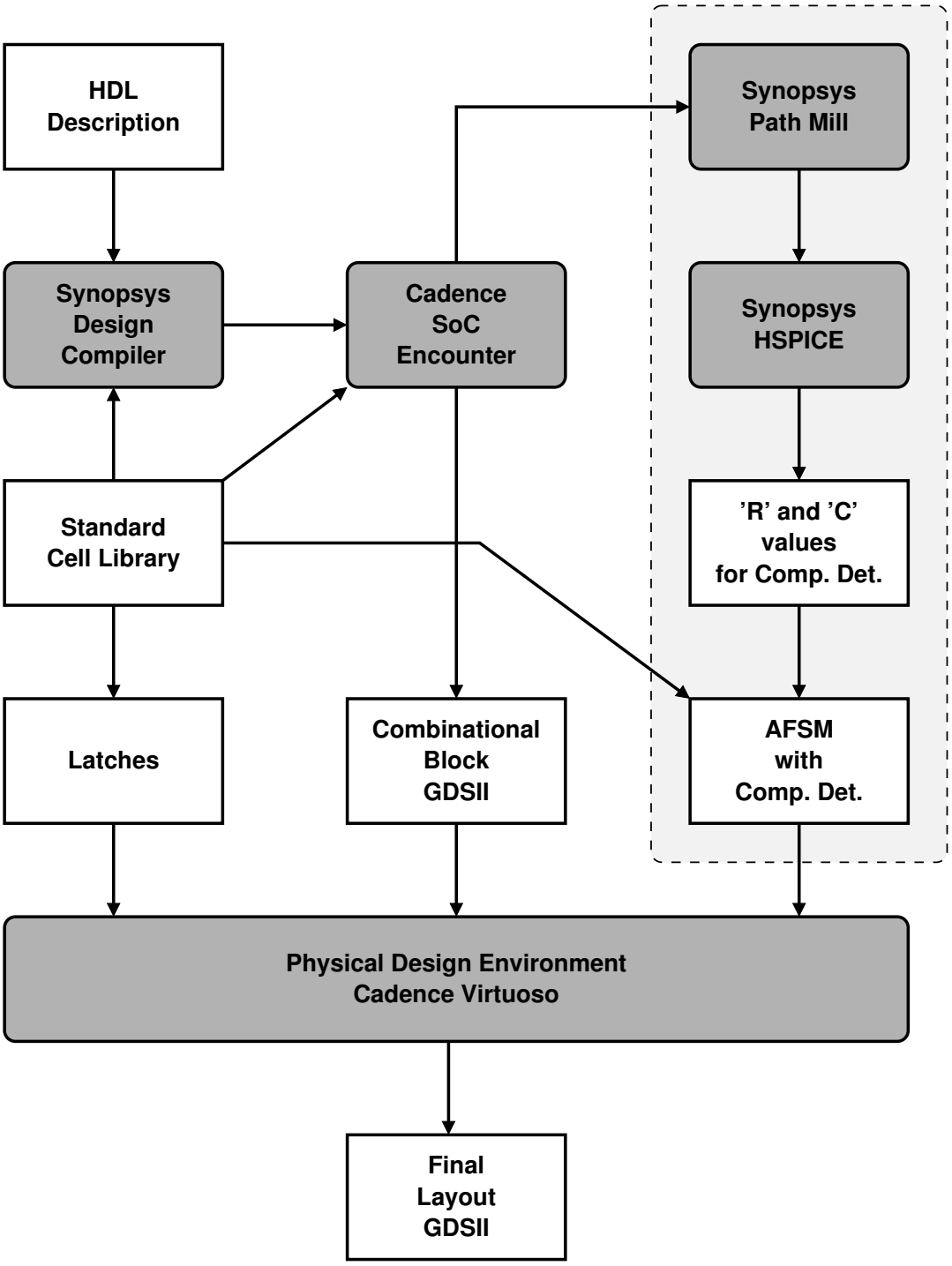


Figure 8.21: Design flow for CDS implementation.

file is generated for importing into the Cadence Virtuoso Physical Design Environment.

Design of the completion detection circuitry depends on transistor level simulations. First, transistor level critical path of the combinational logic circuit is extracted using Synopsys PathMill. On the extracted critical path, the delay of the circuit for sub-threshold operation is simulated using HSPICE. From this simulated critical path delay, values of the pulse generator capacitor C_{pulse}, and the required maximum resistance of the PMOS resistor transistor and its sizing are calculated. For all possible capacitance-resistance pairs, it needs to be guaranteed that the maximum pulse-width generated according to equation (8.4) is greater than the critical path of the combinational circuit for functionally correct operation.

The AFSM in the completion detection system is a relatively small circuit and was manually implemented by placing and connecting standard cells from the library. After completing the design of the variable pulse generator and the AFSM, the blocks that make up the completion detection circuit, i.e., sensor transistor, AC-coupled amplifier, AFSM and the pulse generation circuits, are integrated in Physical Design Environment.

Assuming a design that consists of multiple combinational stages is implemented, the latches/registers that separate the combinational logic stages may be connected to the combinational circuit in either Cadence SoC or Physical Design Environments. In the final step of implementation, designed completion detection system, latches/registers from the standard cell library to separate the combinational stages (if not implemented in Cadence SoC), and the combinational logic block are manually connected in the Physical Design Environment. Furthermore, if required, level converter circuitry for converting the IO-pad signals to sub-threshold values are connected to the rest of the circuitry in this step. An example implementation following the described flow is presented in Figure 8.22. SubBytes operation of the Advanced Encryption Standard (AES) is implemented following the current sensing completion detection methodology. Input latches are implemented in Cadence SoC in a separate power domain than the combinational circuit. Details of power domain separation will be explained in Chapter 9. Level down and up converters are also implemented to be able to convert the IO-pad signals to sub-threshold supply values and vice-versa. The circuit in the figure has two completion detection circuits, one for controlling the input latches and one for controlling the output latches. Thus, we are able to test the correct operation of the completion detection circuit as in a pipelined implementation.

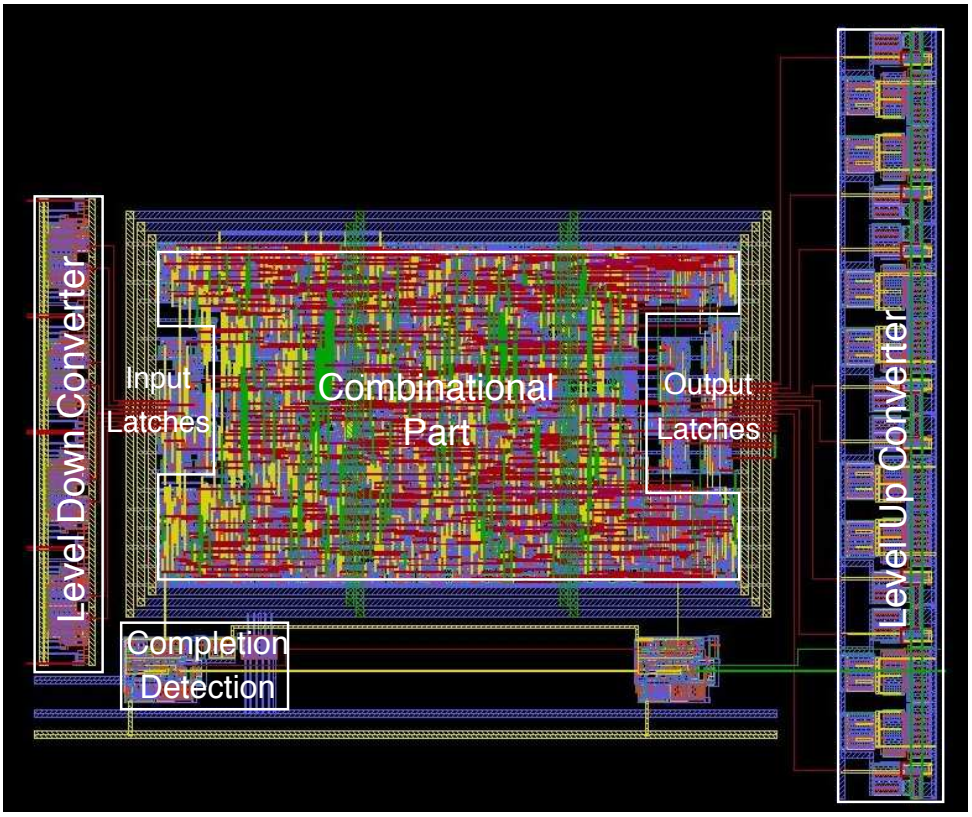


Figure 8.22: Layout implementation of a single stage for testing.

8.7 Results

The current sensing completion detection concept is applied to multiple circuits. Test circuits include, an 8x8 multiplier, an 8-bit adder, a 16-bit adder, a look-up table (LUT) implementing the SubBytes operation of the Advanced Encryption Standard (AES), and a LUT containing randomly distributed entries.

To demonstrate the feasibility of the sub-threshold completion detection circuit, and to make accurate energy dissipation measurements, a test chip containing different circuits implemented using standard synchronous logic, asynchronous micro-pipelines with conventional matched delays, and finally asynchronous micro-pipelines with the current sensing based completion detection circuits were implemented and taped-out in a conventional 0.18 μm CMOS process, see Figure 3.6. Due to last-minute process option changes, i.e., unavailability of low-V_T transistors, we could not measure the performance of the current sensing completion detection system. Therefore, we are presenting the results of post-layout simulations instead.

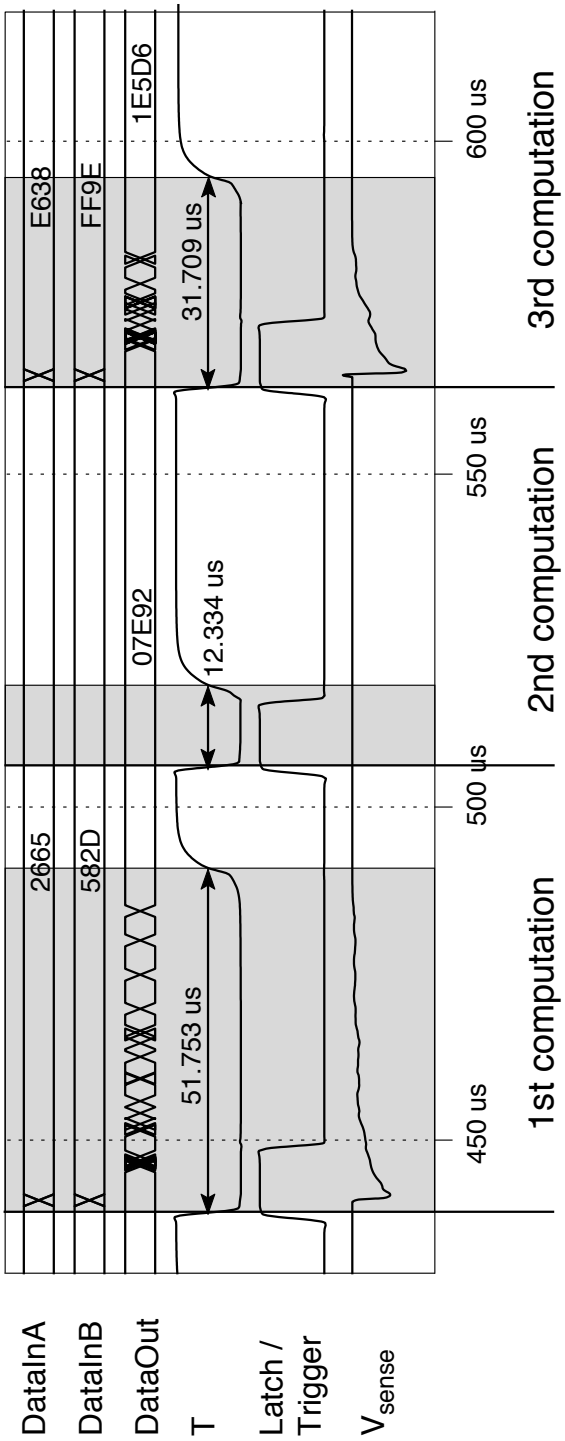


Figure 8.23: Operation of the completion detection circuitry at V_{DD} = 220mV. Three consecutive computations are of the 16-bit adder are shown. The time difference between the computations and the automatic timeout feature are emphasized.

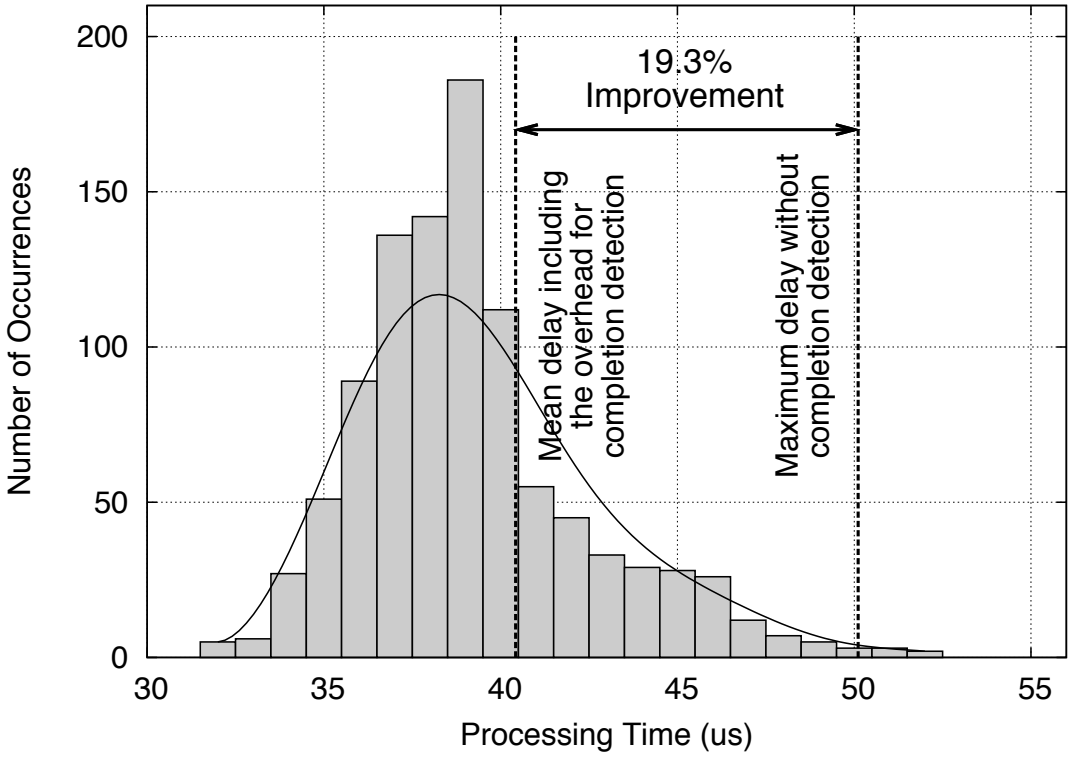


Figure 8.24: Speed improvement in the operation of the 16-bit adder. Both the mean delay generated by the completion detection circuitry and the critical path delay are shown.

Extensive post-layout simulations have been performed on the completion detection system for all the implemented circuits. The waveforms in Figure 8.23 show the operation of the completion detection system for three consecutive cycles operating at $V_{DD} = 0.22\,V$. The supply voltage value of 0.22 V is the energy-minimum supply voltage for synchronous operation of the 16-bit adder. This value is used for all the timing related simulations. The *switching* phase of the first computation continues for a long time, i.e., 51.8 μs. During the second computation, the input data does not change, and consequently no *switching* activity within the combinational block is observed. However, due to the timeout feature, a fast completion detection signal is generated by the circuit, i.e., 12.3 μs. The width of this fast signal is set by the pulse-width of the *latch/trigger* signal as explained before. The last operation is much shorter than the first one and the sensed voltage signal V_{sense} settles more quickly to its final value, resulting in a shorter pulse, i.e., 31.7 μs, compared to the first generated pulse.

Table 8.8: Synchronous and asynchronous energy dissipation comparison

Implementation	Supply voltage	Energy dissipation
Synchronous	0.22 V	31.48 pJ
Asynchronous	0.16 V	26.12 pJ

Figure 8.24 shows the *switching* delay distribution extracted from the post-layout simulation of the 16-bit adder with completion detection for a large number of input vectors while operating with a supply voltage of 0.22 V. It is seen that the mean value is slightly larger than 40 μs. On the same graph critical path delay of the same combinational circuit is shown as well. This figure clearly shows the potential benefit of the completion detection circuit for coarse grained pipeline stages. Including the delay overhead of the completion detection circuit and the sensor transistor, the average gain in throughput is still 19.3%.

In Sections 3.1 and 3.2 it is shown that energy-minimum supply voltages for synchronous and asynchronous operations differ. Furthermore, we investigated the energy reduction due to asynchronous operation with CDS at a lower supply voltage. Table 8.8 compares two implementations: *Synchronous* corresponds to the case where the adder is operating at the synchronous energy-optimum supply voltage value. The *Asynchronous* case shows the implementation with completion detection including the overhead for the completion detection circuitry while working at the asynchronous energy-optimum supply voltage. The energy dissipation numbers given are for equal number of computations while operating on the same input data. For the 16-bit adder circuit, the synchronous energy-minimum supply voltage is 0.22 V while for the asynchronous case the same value is 0.16 V. Due to the asynchronous operation and the lowering of the energy-minimum supply voltage value, asynchronous operation reduces the energy dissipation by 17% for the same computational load.

Furthermore, simulations for characterizing the performance of the completion detection circuit for different combinational circuits were run. Table 8.9 compares two implementations: Matched delay corresponds to the case where the asynchronous

Table 8.9: Comparing matched delay and completion detection at VDD=0.4 V

Test Block	Matched delay	Completion det.	Improvement
AES LUT	0.99 μs	0.82 μs	16.9%
8x8 Multiplier	1.33 μs	1.04 μs	21.8%
8-bit Adder	1.03 μs	0.79 μs	23.3%

Table 8.10: Comparison of the energy dissipation for constant throughput

	Matched delay	Completion detection			
Test Block	E_{core}	E_{core}	Improv.	V_{DD}	Overhead
AES LUT	50.06 pJ	45.5 pJ	9.1%	0.392 V	9.5%
8x8 Multiplier	19.3 pJ	17.9 pJ	7.3%	0.390 V	12.2%
8-bit Adder	3.95 pJ	3.68 pJ	7.3%	0.385 V	41.3%

micro-pipeline is implemented using a matched delay, similar to a synchronous implementation. In this case the number shows the worst case delay of the combinational logic block. The Completion det. column shows the implementation with completion detection including the overhead for the completion detection circuit. The number given here is the average delay over a large number of sample inputs. Both implementations are operated using a supply voltage of VDD=0.4 V. This table shows the potential of the completion detection circuit for coarse grained pipeline stages. Speed improvements of up to 23% are achieved by using completion detection circuits, hence reducing leakage energy dissipation correspondingly.

The energy dissipation of the asynchronous circuit with *Matched delay* and *completion detection* is compared in Table 8.10. In this table the *Matched delay* based asynchronous micro-pipeline component is supplied with VDD=0.4 V. The power supply of the circuit using completion detection was adjusted such that both circuits have the same throughput. Thus, it is possible to compare the energy per throughput performance of both implementations. In the table E_{core} is the energy dissipation of the combinational logic block in both cases. The overhead is the energy dissipated by the completion detection circuit. Note that the matched delay or synchronous implementation will have a similar overhead. It is clear from the overhead numbers that the completion detection technique presented in this chapter is more suitable for coarse grain pipelines. Therefore, by trading the delay improvement for lower energy dissipation, energy savings up to 9% are achieved for the same throughput in smaller circuits. Moreover, corner simulations were run for post-layout circuits and the completion detection circuit functioned without any loss of performance for all the corner cases except the slow-slow corner. In corners other than the slow-slow one, completion detection circuit successfully tracked the variation in processing time of the logic block thanks to the MOS-based capacitors and PMOS resistance transistor. On the other hand for the slow-slow corner high correlation of the generated pulse width with the actual processing time worsens due to the slow AC-coupled amplifier response and slower charging of n2 in Figure 8.11.

Chapter 9

Implementation of a Self-timed Cardiac Event Detector

This chapter presents the methodology and the implementation results of a current sensing completion detection (CSCD) system on a real-world circuit implementation, i.e., a digital event detector for cardiac pacemakers. Digital event detector is a more complex circuit by orders of magnitude when compared to the test circuits in Chapter 8. Thus, a different implementation flow will be employed to realize the completion detection circuit. Furthermore, due to the fact that the event detector cannot be divided into simple combinational logic-only pipeline stages with ease, another methodology for current sensing is proposed.

9.1 Power Domain Separation

The current sensing completion detection concept is applicable for sensing the current that is drawn by the combinational circuitry. In circuits where the majority of the gates are combinational, the same power domain may be used for both combinational and sequential elements. Due to the majority of the combinational gates, the current drawn by the whole circuit may be sensed with the sensor transistor without saturating the following AC-coupled amplifier.

On the other hand, the R-wave event detector includes a substantial amount of sequential gates, i.e., registers. Separated current waveforms for the combinational logic and the registers are shown in Figure 9.1. Waveform of the current drawn by the sequential elements has sharp and instantaneous changes, whereas the current waveform of the combinational logic part is spread over time with a smaller amplitude.

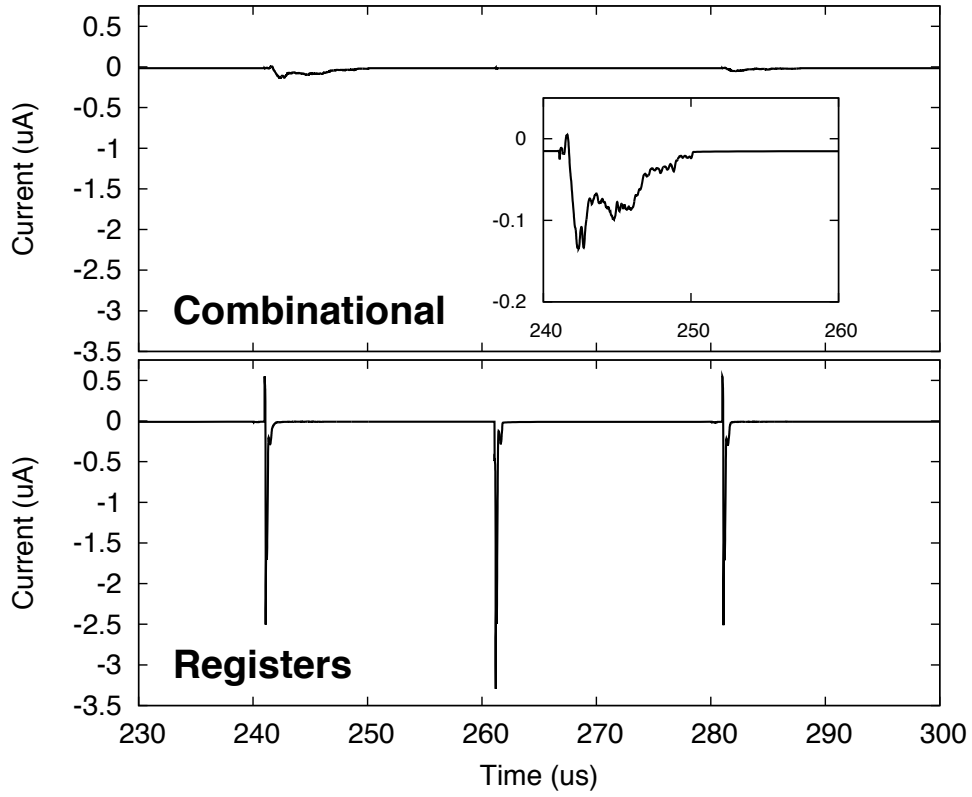
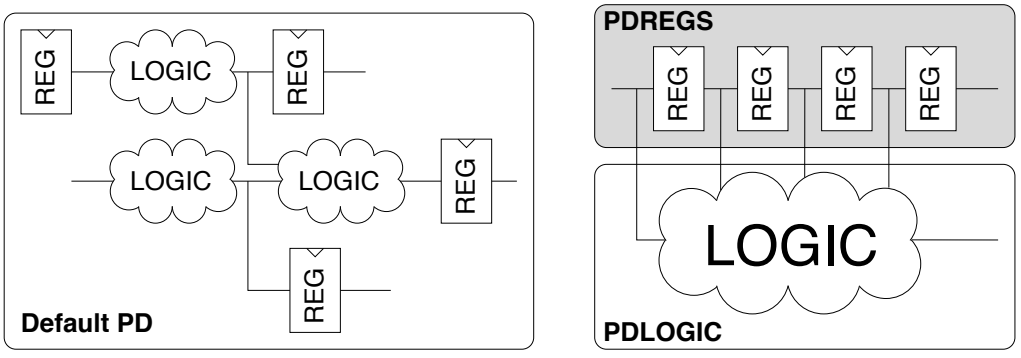


Figure 9.1: Separated current waveforms of the combinational and sequential gates of the R-wave event detector.



(a) Single power domain. (b) Separate power domains.

Figure 9.2: Concept of power domain separation for completion detection circuit implementation. (a) Single and (b) multiple power domain implementations are shown.

Hence, separation of the combinational and memory elements of a complex circuit are crucial for proper operation of the current sensing completion detection system.

Asynchronous implementation of the digital R-wave event detector presented in this chapter employs the separation of the power domains as shown in Figure 9.2. The power domain separation process is automated and incurs little overhead in terms of routing and area as will be explained next.

9.2 De-synchronization Flow For Complex Circuits

The design flow to convert the synchronous version of the R-wave event detector is shown in Figure 9.3. The steps in the de-synchronization flow are as follows: After verifying the HDL description of the circuit (synchronous) is working properly, it is synthesized using Synopsys Design Compiler using the standard cell library provided by the foundry. After the circuit has been synthesized as in standard synchronous design flow, the memory elements in the design, e.g., latches and flip-flops, are grouped as a separate module using

```
group -design_name REGDOM [all_registers]
```

command in Design Compiler. This command creates a new module called `REGDOM` in the design and puts all the memory elements inside it. Later in the placement and routing process this new module will be assigned a different power supply than the combinational gates. Following the separation of logic and memory elements, final gate level netlist is written, placed, and routed in Cadence SoC Encounter. As mentioned, for only sensing the logic gates' current usage, the logic gates and memory elements need to be assigned and connected to different power domains. This is realized by creating multiple supply domains in SoC Encounter using the Common Power Format (CPF) flow [98]. The complete CPF definition file is given in Appendix C.

In the CPF flow two power domains, `PDLOGIC` and `PDREGS`, are created and assigned different power nets, `VDDLOGIC` and `VDDREG`, respectively. No voltage shifting occurs between different power domains in our design so both power domains are assigned a common ground note, i.e., `VSS`. Two versions of the R-wave event detector, i.e., synchronous and asynchronous, are shown in Figure 9.4. The asynchronous version, which is basically the sequential and combinational elements separated, is slightly bigger than the synchronous version of the design. The total area of the circuits are $19425\mu m^2$ and $21000\mu m^2$, for the synchronous and asynchronous versions,

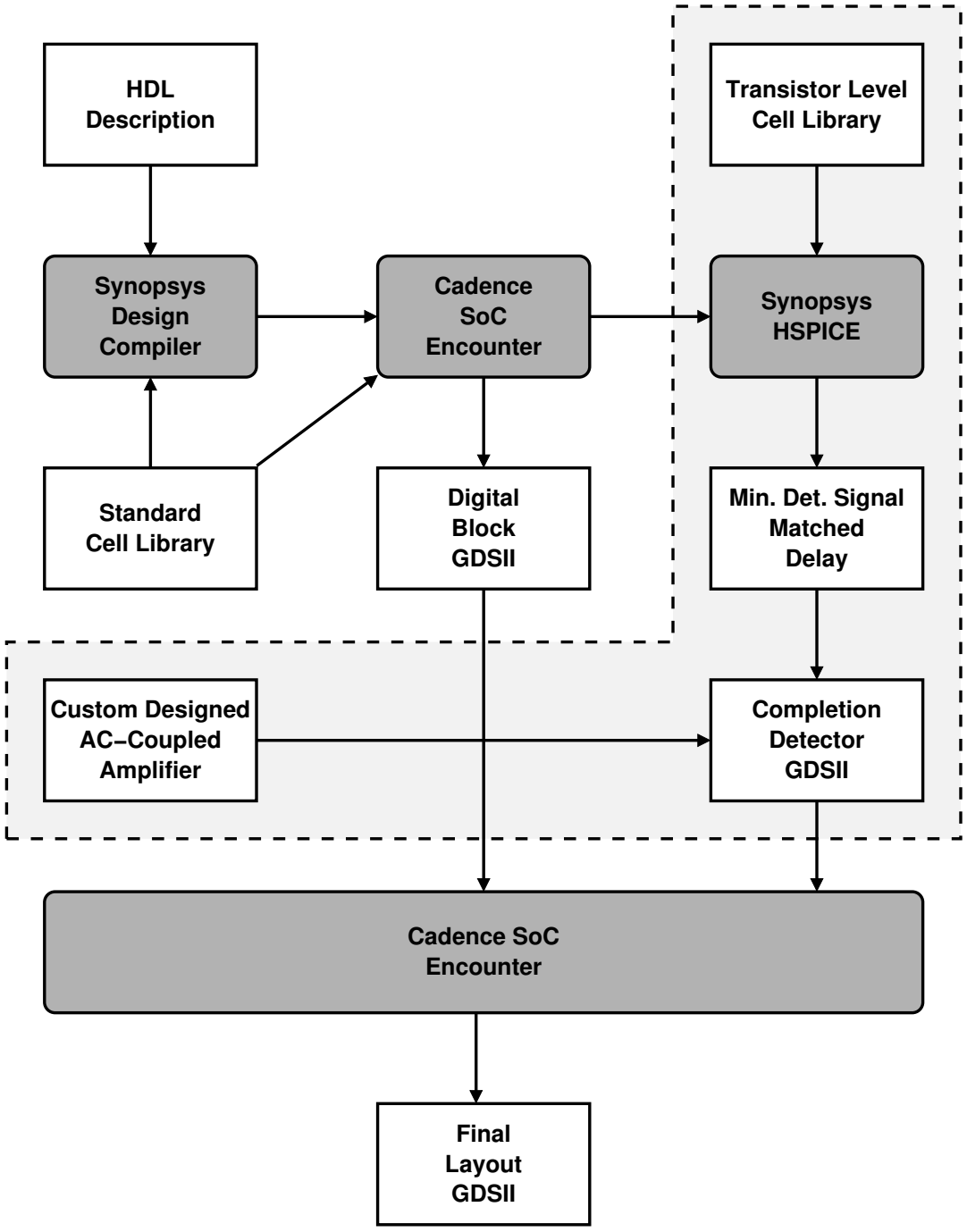
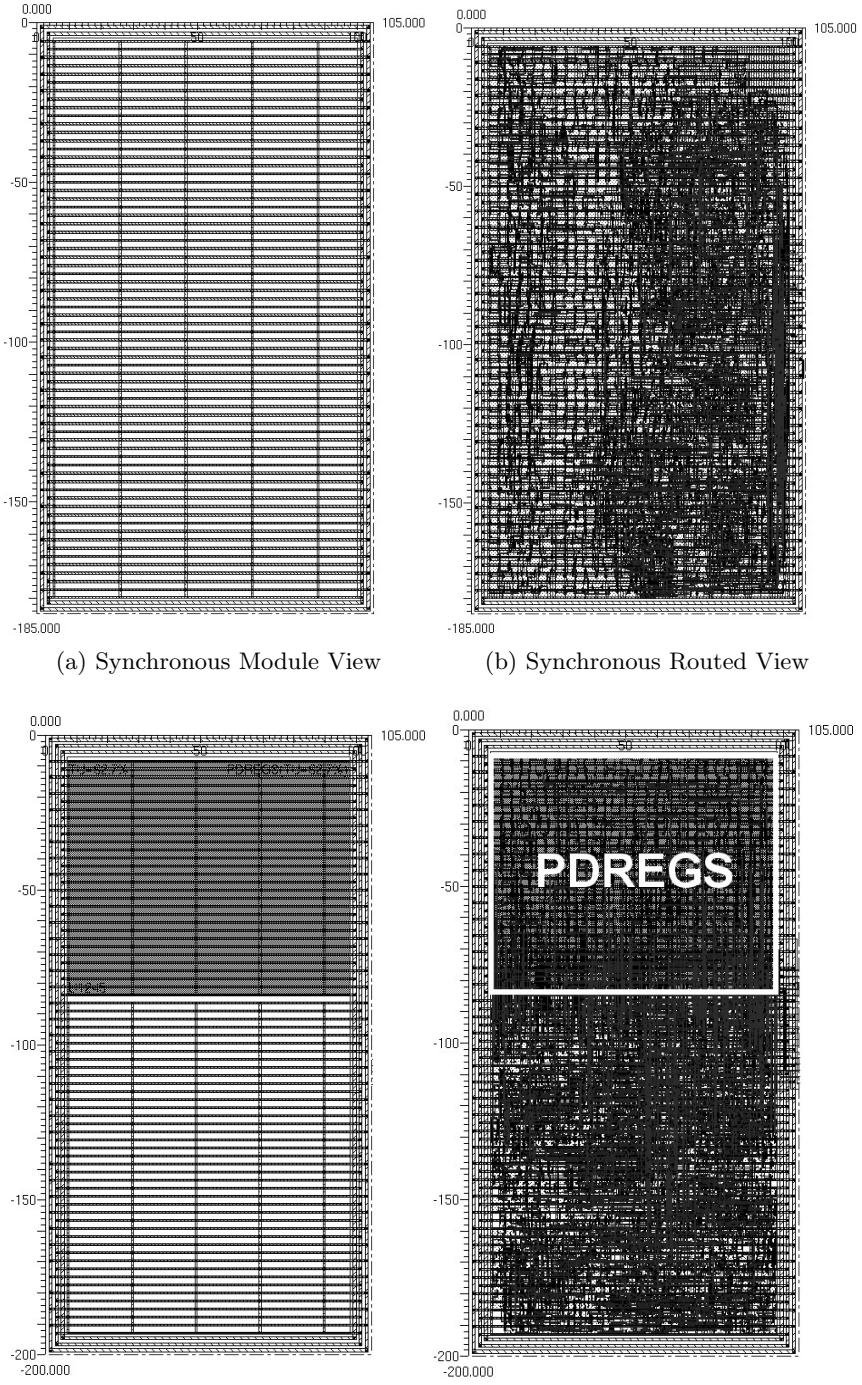


Figure 9.3: Design flow for test chip implementation.



(a) Synchronous Module View (b) Synchronous Routed View

(c) Asynchronous Module View (d) Asynchronous Routed View

Figure 9.4: Visual comparison of synchronous and asynchronous versions of the cardiac event detector in terms of routing and placement.

respectively. This increase, 8.2%, in area occurs due to a single row required for dividing different power domains and routing overhead due to sub-optimal placement of the memory elements in the circuit. Area increase due to the CSCD system is $897\mu m^2$, i.e., 4.6%. After the placement and routing, a Graphic Data System (GDSII) file is generated for the routed block.

To estimate the correct size of the parallel pulse generator, SPICE level simulations are run. From the routed block a Verilog file is generated, and is converted to HSPICE netlist format using custom scripts. This netlist file is instantiated in the HSPICE testbench with the sensing transistor and the AC-Coupled amplifier. Transient simulations are run to find the minimum signal amplitude that may be sensed. By applying this simulated minimum signal value to the AC-coupled amplifier, the width of the required pulse that will operate in parallel to the AC-coupled amplifier is found. According to the simulation results, a minimum detectable signal matched delay circuit is designed. The GDSII file for this design may be created in two ways: It may be either (i) fully custom or (ii) standard ASIC design flow may be applied for generating the layout and the GDSII file. After the design of the matched pulse generator is completed, layout of the AC-coupled amplifier is combined with the pulse circuitry and a GDSII file for the whole completion circuit is generated. Finally, both the digital and analog (completion detection) GDSII files combined, placed and routed in SoC encounter and a final layout GDSII is generated for the tape-out.

9.2.1 Routing Capacitance Overhead

Separating power domains result in both area overhead and increased routing density as observed in Figure 9.4. A substantial increase in the routing capacitance results in switching energy overhead. We investigated effects of power domain separation on the length of the routing wires of the digital event detector circuit. Figure 9.5 shows the wire length distributions after placement and routing of the circuits in Cadence SoC Encounter. The wire length distributions of two implementations are compared, and it is seen that the increase in the wire lengths for the asynchronous case is not significant. Maximum wire-length in the whole distribution has increased as expected, but overall distribution of the wire-lengths is the same with the same mean value. This shows that de-synchronization and separation of the power-domains do not incur significant routing overhead.

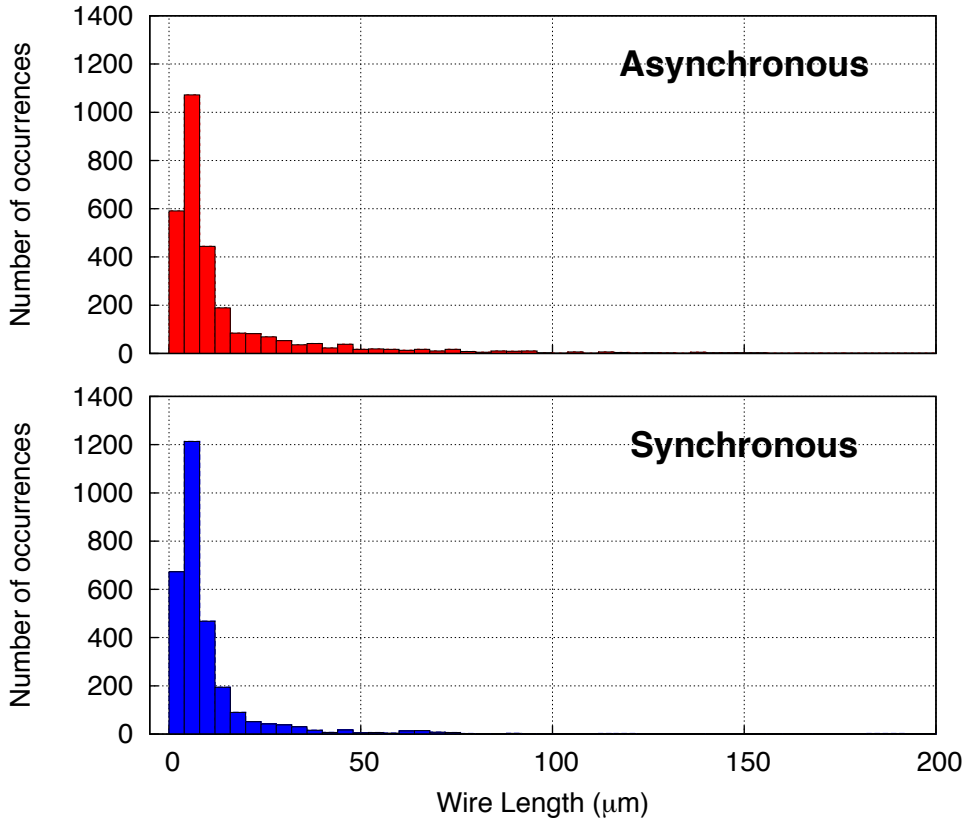


Figure 9.5: Routing capacitance distribution for synchronous and asynchronous implementations of the event detector.

9.2.2 Completion Detection Circuit for Digital R-wave Event Detector

The completion detection circuit used to implement an asynchronous R-wave event detector is presented in Figure 9.6. This implementation is similar to the implementation previously shown in Figure 8.16. Both implementations share the same STG and the communication protocol shown in Figure 8.15. Unlike the previous circuit implementation, we removed the variable pulse-width generator. This is possible due to an increased complexity, and hence higher current drawn by the circuit, where the sensed signal is strong enough to drive the AC-coupled amplifier to voltage values close to the supply voltage. Thus, it is possible to trigger the digital circuitry, i.e., AFSM, using the amplified analog signal.

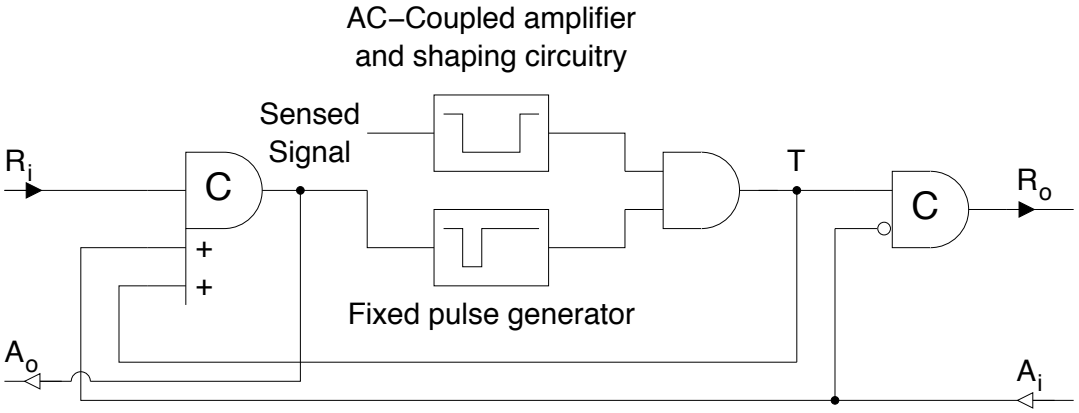


Figure 9.6: AFSM including the pulse generating completion detection circuitry.

A fixed pulse generator in parallel to the AC-Coupled amplifier and shaping circuitry is implemented to generate a pulse signal for two cases. Firstly, no combinational switching occurring, and secondly minimal amount of switching occurring in the circuit that cannot be amplified by the AC-coupled amplifier for converting to a logic level. Thus, this fixed pulse generator both realizes the *time-out* feature and guarantees correct operation for the cases where the sensed signal is not strong enough. The sizing of the fixed pulse generator is done based on the HSPICE simulation results as explained in the previous section.

9.3 Test Chip Implementation

A test chip to verify the functionally correct operation of the asynchronous R-wave event detector and different architectures presented in Chapter 7 was taped-out. Top level implementation of the chip is presented in Figure 9.7. The test chip includes both synchronous and asynchronous versions of the unfolded R-wave event detector as well as both pipelined and unpipelined versions of folded by six event detector. Asynchronous implementation of the event detector uses the circuit shown in Figure 9.6. Both AC-coupled amplifiers presented in Chapter 8 were simulated with the transistor level implementation of the event detector circuitry. In this test chip fabrication, first version of the AC-coupled amplifier was used.

In a complex circuit such as the R-wave event detector, there may be many paths which have delays equal, or close to the critical path delay of the circuit. It can be argued that, current sensing completion detection may not be as effective as the case where there is a single dominating critical path. Figure 9.8 shows the normalized

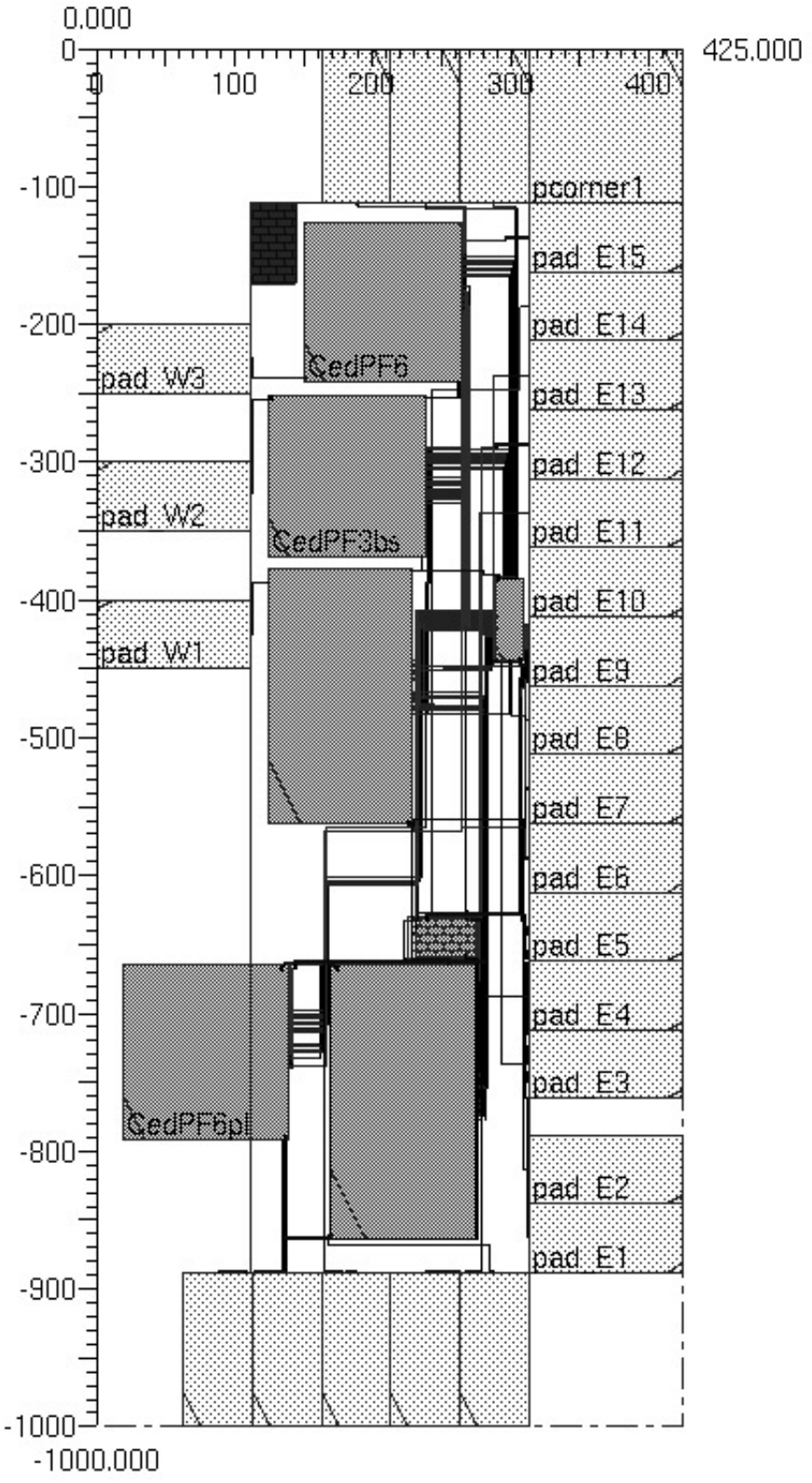


Figure 9.7: Top level of SVCED, i.e., Sub-V_T Cardiac Event Detectors.

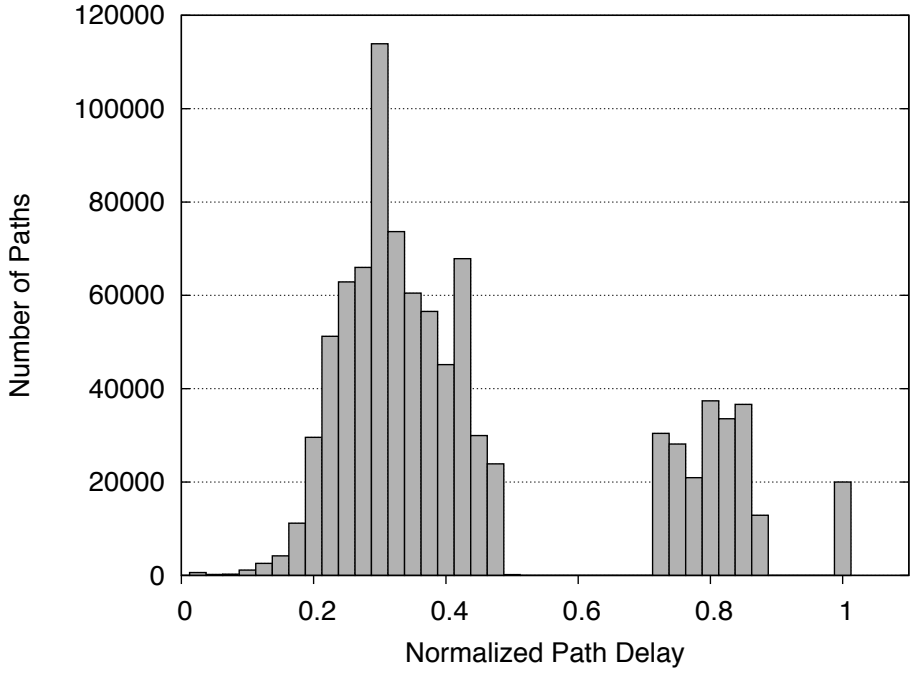


Figure 9.8: Normalized timing paths of the reference circuit. All path values are normalized to the critical path delay.

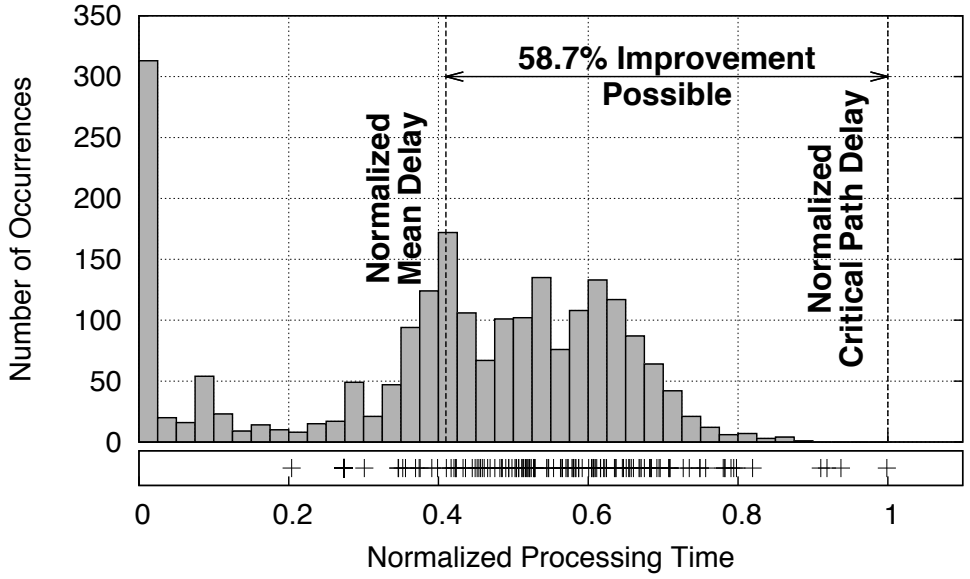


Figure 9.9: Data processing time distribution histogram of the event detector. All processing time values are normalized to the critical path delay.

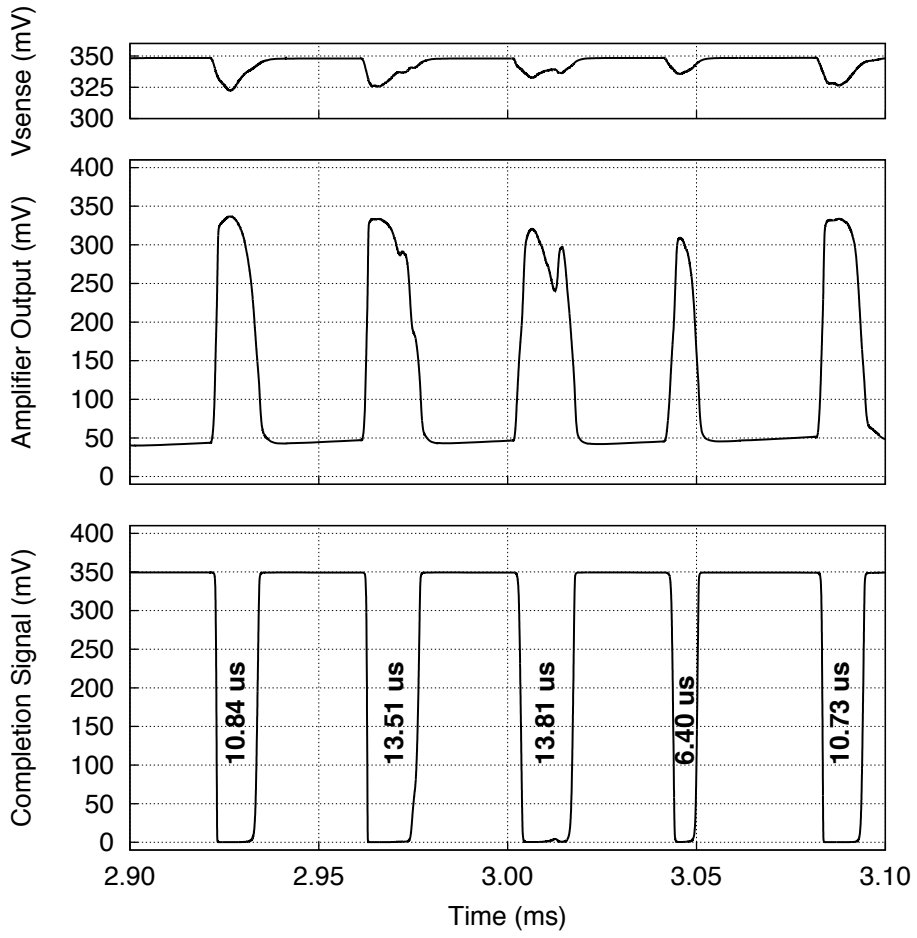


Figure 9.10: Transient simulation results of the completion detection circuit at a supply voltage of 350 mV.

path delay distribution of the reference design for all the paths in the circuit. All path delays are normalized to the critical path delay of the circuit. In the reference design, there are more than 20000 paths that are close to the critical path value. Therefore, to see the possible gain in asynchronous operation, processing time of the circuit while processing real data needs to be investigated.

The histogram in Figure 9.9 shows the normalized processing time of the circuit while processing real-life data from an electrogram database [5]. The computation time data presented in the histograms is obtained by processing the power waveforms generated by Synopsys PrimeTime for 2200 data samples. All the processing time values are normalized to the critical path delay as in the previous case. High number of low processing value occurrences are due to repeated processing of the same data

Table 9.1: Energy minimum operating voltage and respective minimum energy dissipation of the reference circuit implemented in a commercial 65 nm process for different operation modes.

Operation	EMV (mV)	Energy (fJ)
Synchronous	330.6	973.4
Asynchronous	296.1	809.7

or due to the periods where the data at the input of the reference design does not change. Based on the processing times, a speed improvement of 58.7% is possible while operating asynchronously with a completion detection circuit.

HSPICE simulation results of the completion detection circuit for 200 data samples are shown below the histogram in Figure 9.9. According to the results of HSPICE simulations, completion detection circuit implemented using the presented flow results in 52.58% throughput improvement when compared to the synchronous case while operating with a supply voltage of 0.35 V. In our implementation, pulse-width generated by minimum detectable current signal is 20.7% of the critical path. Transient simulation results of the completion detection circuit while operating at a supply voltage of 350 mV are presented in Figure 9.10. In the figure variable width pulses generated by the circuit are shown.

The improvement in the throughput results in moving of the EMV. This improvement is calculated by setting μ_d in equation (3.12) to 0.47, i.e., $1-0.53$. Change in EMV and reduction in energy dissipation resulting due to asynchronous operation is presented in Table 9.1. By trading the throughput improvement and moving to a lower EMV, energy dissipation of the same circuit is reduced by 16.8%. Energy dissipation of the completion detection circuit is 18.36 fJ, which is 2.3% of the total energy dissipation in asynchronous mode.

Chapter 10

Conclusions

In this thesis, various methods to enhance energy efficiency of sub-V_T digital circuits are discussed. Separate conclusions may be drawn for different topics presented.

10.1 Summary of Contributions

A High Level Sub-V_T Energy Model

- A high level energy estimation model for sub-threshold operation was developed.
- The model is developed based on asynchronous specifications, and is later extended for synchronous operation.
- Model parameters are obtained from standard cell synthesis results and switch level (synthesized HDL) simulations.
- The proposed model reduces the simulation time by 270x with respect to transistor level simulations for ISCAS85 benchmark circuits. The model uses back-annotated toggle information, and, thus, realistic use-cases may be simulated.
- The model is easily extensible to explore architectural modifications.
- Model accuracy is confirmed by measurements. Measured error is between -4.79% and 3.83% for different operating conditions in a $0.18\,\mu$m CMOS process.

Optimum Process Selection for Energy Minimization

- Modern CMOS processes, i.e., 180, 130, 90 and 65 nm, are compared for their energy efficiency.

- Unlike previously published work, special low-power process options are taken into consideration.
- It has been shown that with the correct choice of process options, migrating to smaller feature sized processes is beneficial in terms of energy efficiency.
- By choosing the correct process options, energy dissipation reduction of 21.4% is possible while moving from one technology node to another.

Architectural Energy Reduction

- Effects of simple architectural changes, such as pipelining and parallelism, are shown both analytically and through simulation results.
- It is shown that for register heavy circuit implementations, pipelining enhances energy efficiency.
- Folding is shown to reduce energy dissipation for a very low speed application, i.e., R-wave event detector operating at 1 kHz. Folding reduces the area by 31% while improving the energy efficiency by 14.4%.

Asynchronous Sub-V_T Operation

- Energy reduction by asynchronous operation in sub-V_T regime is shown by analytical derivation, numerical simulations and by applying asynchronous operation to benchmark circuits.
- A current sensing completion detection system for sub-V_T operation is proposed and implemented.
- Different flows for de-synchronization and implementation of the completion detection using commercial EDA tools are presented.
- Energy efficiency and/or speed enhancement due to completion detection is evaluated on various basic circuits.
- Proposed completion detection idea is employed to implement an asynchronous version of an R-wave cardiac event detector in a 65 nm process.
- Asynchronous implementation increases throughput by 52.58%. When this speed increase is traded for better energy efficiency, energy dissipation is reduced by 16.8%.

10.2 Conclusions and Future Directions

In this work we explored different methods to improve the energy efficiency of sub-V_T digital CMOS circuits. The work presented explores energy efficiency enhancement in different levels of the design process. In this section we provide possible future work for improving the results presented in this thesis.

High level energy model presented in Chapter 3 does not take process variation related effects into consideration. For large circuits, random dopant fluctuation variations average and do not affect the EMV or energy dissipation significantly. On the other hand, they cause functional failure and increase delay variation of the circuit. Thus, future work may concentrate on improving the energy model to take process variations into consideration. Furthermore, the energy model may be improved for design for manufacturability, which is a topic not covered for sub-V_T operation yet.

In the area of architectural energy reduction, simple improvements like serialization may be explored for energy efficiency. Furthermore, algorithmic simplifications, extension of above-threshold power reduction techniques for sub-V_T operation are strong candidates for reducing energy dissipation. Combining the energy model and different architectural changes on a parametric framework, it is also possible to automate the analysis and optimization of digital sub-threshold systems, providing a complete flow from algorithms to optimum circuit implementation.

Asynchronous sub-V_T operation has been shown to enhance the energy efficiency of digital systems. One major improvement possible in the design of the completion detection circuit is the ability to track PVT variations. Although for the variable pulse generator case, MOSCAPs are used for tracking PVT variations, they do not provide complete immunity to PVT effects. If a system with complete PVT immunity is realized, it is expected to reduce energy dissipation even more when compared to the synchronous case, as delay variation and functional failure is more pronounced for sub-threshold operation.

Appendix A

Lambert-W Function

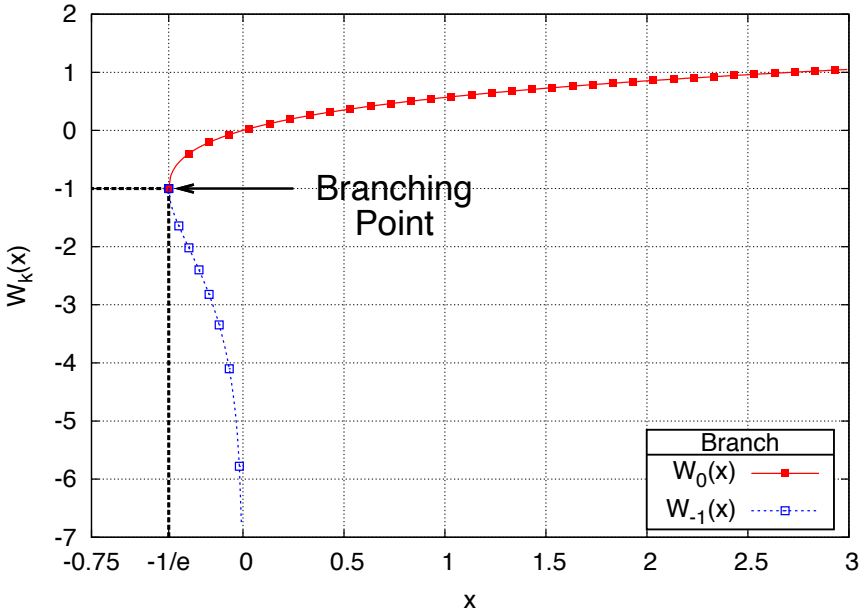


Figure A.1: Plot of *Lambert W* function showing the real valued branches.

In mathematics, the *Lambert W* function, $W = LambertW(x)$, gives the multi-valued solution to the equation

$$We^W = x \tag{A.1}$$

where e^W is the natural exponential function and W is an arbitrary complex number.

The plotted function is shown in Figure A.1. W function has a branch point of order 2 at $x = -1/e$ and has two real solutions in the range $(-1/e, 0)$. W function assumes complex values for $x < -1/e$. In general, the branch satisfying $-1 \leq W(x)$

is denoted by $W_0(x)$ and the branch satisfying $W(x) \leq -1$ by $W_{-1}(x)$. The branch defined by $W_0(x)$ is called the *principal branch* of the W [63].

Appendix B

Log-Normal Distribution

The probability density function (PDF) of a log-normal distribution is characterized by the mean and the standard deviation of the variable's logarithm, μ and σ, respectively. The PDF of the log-normal distribution is given by

$$f(x, \mu, \sigma) = \frac{e^{-(\ln x - \mu)/(2\sigma^2)}}{x\sigma\sqrt{2\pi}} \tag{B.1}$$

and the mean and variance of the log-normal variable X are given by

$$E(X) = e^{\sigma^2/2+\mu} \tag{B.2}$$

$$Var(X) = e^{\sigma^2+2\mu}(e^{\sigma^2} - 1) \tag{B.3}$$

Appendix C

CPF Definition File

```
###################################################
#       Technology part of the CPF
###################################################
# define the library sets
define_library_set -name libSet \
        -libraries \
        {PAR/TIM/CORE65LPHVT_nom_1.20V_25C.lib}

###################################################
#       Design part of the CPF
###################################################
set_design QRSdetectorAsync
# create power domains
create_power_domain -name PDLOGIC -default
create_power_domain -name PDREGS -instances {U1245}

# create nominal conditions
create_nominal_condition -name allV -voltage 1.2

# create power mode
create_power_mode -name PM \
        -domain_conditions {PDLOGIC@allV PDREGS@allV} \
        -default
```

```
update_nominal_condition -name allV \
        -library_set libSet

# declare power and ground nets
create_ground_nets -nets VSS
create_power_nets -nets VDDLOGIC -voltage 1.2
create_power_nets -nets VDDREG -voltage 1.2

# create global connections
create_global_connection -domain PDLOGIC \
        -net VSS -pins gnd
create_global_connection -domain PDLOGIC \
        -net VDDLOGIC -pins vdd
create_global_connection -domain PDREGS \
        -net VSS -pins gnd
create_global_connection -domain PDREGS \
        -net VDDREG -pins vdd

# add implementation info for power domains
update_power_domain -name PDLOGIC \
        -internal_power_net VDDLOGIC
update_power_domain -name PDREGS \
        -internal_power_net VDDREG
end_design
```

Listing C.1: CPF file for seperating power domains.

Appendix D

AFSM Design Related

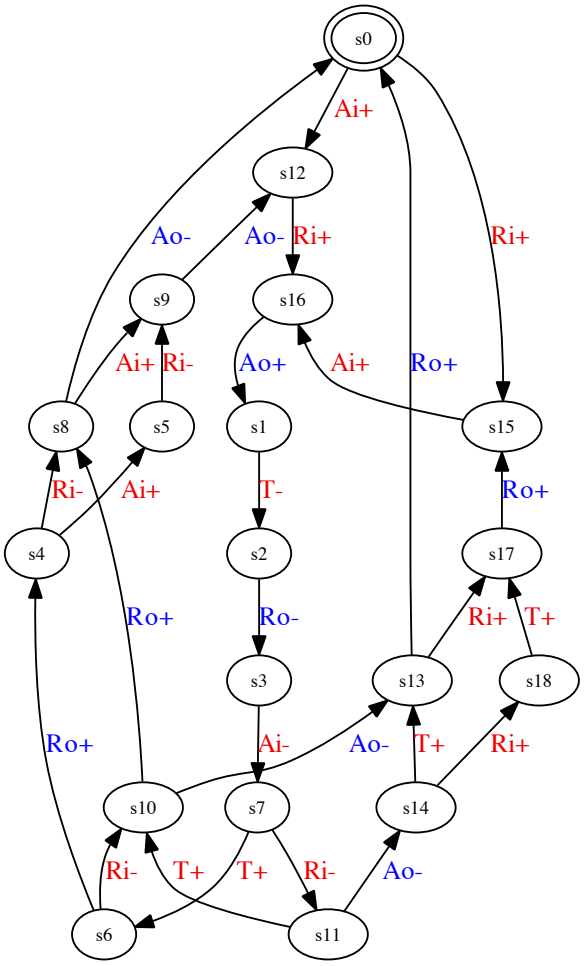
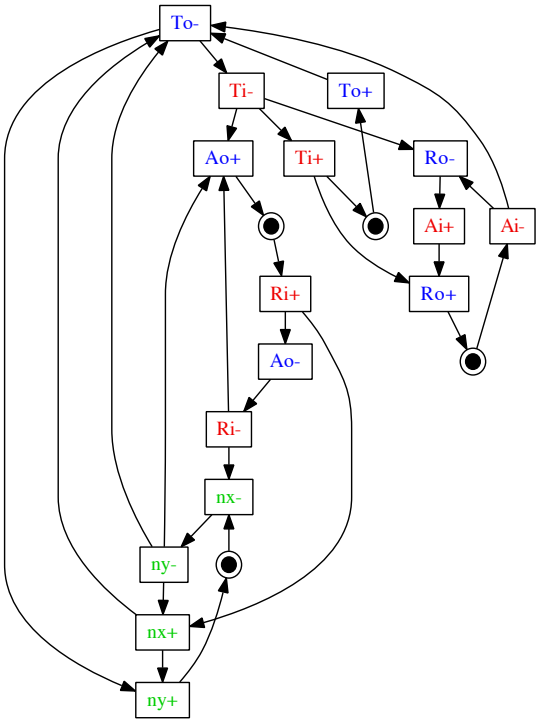


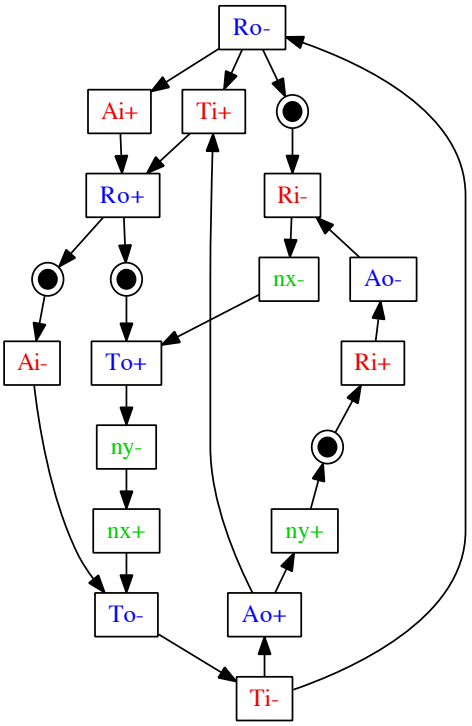
Figure D.1: State diagram of the first version of the implemented AFSM.

```
.model comProtMerged_v5

# Signals
.inputs Ri Ai Ti
.outputs Ro Ao To
.internal nx ny

# Petri net
.graph
Ri- Ao+ nx-
Ao+ Ri+
Ri+ Ao- nx+
Ao- Ri-
Ro- Ai+
Ai+ Ro+
Ro+ Ai-
Ai- Ro- To-
To- Ti- ny+
Ti- Ti+ Ro-
Ti+ To+ Ro+
To+ To-
nx+ ny+ To-
ny+ nx-
nx- ny-
ny- nx+ To- Ao+
Ti- Ao+

.marking {<Ao+,Ri+> <Ro+,Ai-> <Ti+,To+> <ny+,nx->}
.slowenv
.end
```

Listing D.1: Textual representation of the AFSM shown in Figure D.2a



(a) Initial definition



(b) Simplified version

Figure D.2: STG for the second version of the completion detection AFSM where all the signals are inter-connected. (a) Initial definition, and (b) simplified version are shown.

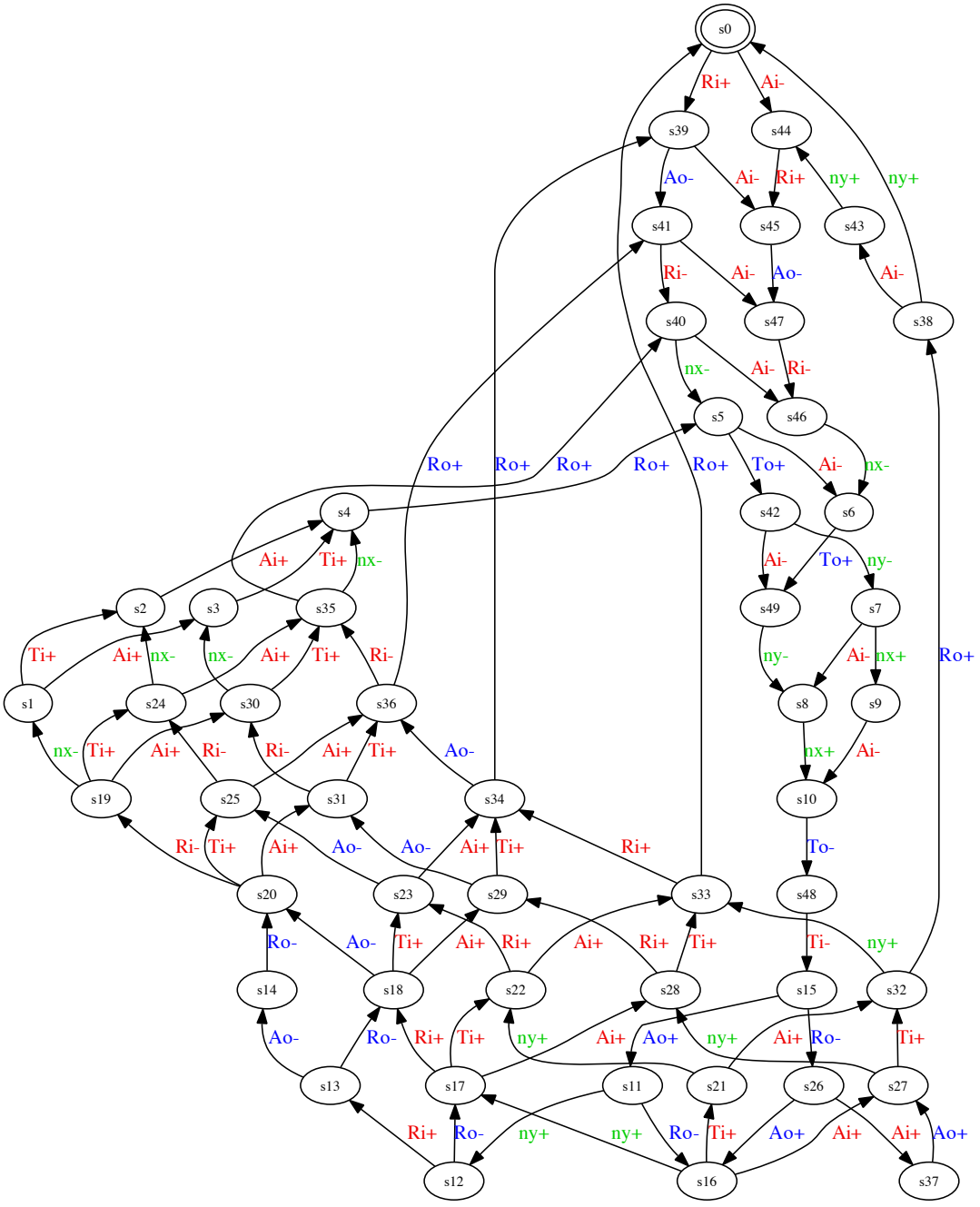


Figure D.3: State diagram of the second version of the implemented AFSM.

Bibliography

[1] J. Sparso and S. Furber, Eds., *Principles of Asynchronous Circuit Design - A Systems Perspective*. Kluwer Academic Publishers, 2001.

[2] S. Haddad and W. Serdijn, *Ultra Low-Power biomedical Signal Processing. An analog Wavelet Filter Approach for Pacemakers*. Springer Verlag, 2009.

[3] A. Guyton and J. Hall, *Textbook of Medical Physiology. 11th Edition*. Philadelphia: WB Saunders Co, 2006.

[4] C. J. Love, *Cardiac Pacemakers and Defibrillators, Second Edition*. Landes Bioscience, 2006.

[5] J. Rodrigues, "Development and implementation of cardiac event detectors in digital cmos," Ph.D. dissertation, Lund University, 2005.

[6] J. Rodrigues, O. C. Akgun, A. de la Calle, P. Acharya, Y. Leblebici, and V. Owall, "Energy dissipation reduction of a cardiac event detector in the sub-vt domain by architectural folding," in *Proceedings of International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS) 2009 (in press)*, 2009.

[7] E. Seevinck, F. List, and J. Lohstroh, "Static-noise margin analysis of MOS SRAM cells," *IEEE Journal of Solid-State Circuits*, vol. 22, no. 5, pp. 748–754, 1987.

[8] S. Roundy, P. Wright, and J. Rabaey, "A study of low level vibrations as a power source for wireless sensor nodes," *Computer Communications*, vol. 26, no. 11, pp. 1131–1144, 2003.

[9] J. Hennessy and D. Patterson, *Computer architecture: a quantitative approach*. Morgan Kaufmann, 2007.

[10] M. Pedram, "Tutorial and Survey Paper Power Minimization in IC Design: Principles and Applications," *ACM Transactions on Design Automation of Electronic Systems*, vol. 1, no. 1, pp. 3–56, 1996.

[11] G. Téllez, A. Farrahi, and M. Sarrafzadeh, "Activity-driven clock design for low power circuits," in *Proceedings of the 1995 IEEE/ACM international conference on Computer-aided design*. IEEE Computer Society Washington, DC, USA, 1995, pp. 62–65.

[12] M. Takahashi, M. Hamada, T. Nishikawa, H. Arakida, T. Fujita, F. Hatori, S. Mita, K. Suzuki, A. Chiba, T. Terazawa *et al.*, "A 60-mW MPEG4 video codec using clustered voltage scaling withvariable supply-voltage scheme," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 11, pp. 1772–1780, 1998.

[13] S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, and J. Yamada, "1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS," *IEEE Journal of Solid-state circuits*, vol. 30, no. 8, pp. 847–854, 1995.

[14] B. Calhoun, A. Wang, and A. Chandrakasan, "Device sizing for minimum energy operation in subthreshold circuits," in *Custom Integrated Circuits Conference, 2004. Proceedings of the IEEE 2004*, 2004, pp. 95–98.

[15] E. Vittoz, *Low-Power Electronics Design*. CRC Press LLC, 2004, ch. 16.

[16] B. G. Streetman and S. Banerjee, *Solid State Electronic Devices 5th Edition*. Prentice Hall, 2000.

[17] Y. Tsividis, *Operation and Modeling of the MOS Transistor*. Oxford University Press, USA, 1999.

[18] R. Swanson and J. Meindl, "Ion-implanted complementary MOS transistors in low-voltage circuits," *Solid-State Circuits, IEEE Journal of*, vol. 7, no. 2, pp. 146–153, 1972.

[19] E. Vittoz and J. Fellrath, "CMOS analog integrated circuits based on weak inversion operations," *IEEE Journal of Solid-State Circuits*, vol. 12, no. 3, pp. 224–231, 1977.

[20] E. Vittoz and O. Neyroud, "A low-voltage CMOS bandgap reference," *IEEE Journal of Solid-State Circuits*, vol. 14, no. 3, pp. 573–579, 1979.

[21] C. Enz and E. Vittoz, "CMOS low-power analog circuit design," *Designing Low Power Digital Systems, Emerging Technologies (1996)*, pp. 79–133, 1996.

[22] E. Seevinck, E. Vittoz, M. du Plessi, T. Joubert, and W. Beetge, "CMOS translinear circuits for minimum supply voltage," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, no. 12, pp. 1560–1564, 2000.

[23] H. Soeleman and K. Roy, "Ultra-low power digital subthreshold logic circuits," in *Low Power Electronics and Design, 1999. Proceedings. 1999 International Symposium on*, 1999, pp. 94–96.

[24] H. Soeleman, K. Roy, and B. Paul, "Robust ultra-low power sub-threshold dtmos logic," in *Low Power Electronics and Design, 2000. ISLPED '00. Proceedings of the 2000 International Symposium on*, 2000, pp. 25–30.

[25] J. Kao, M. Miyazaki, and A. Chandrakasan, "A 175-mv multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1545–1554, 2002.

[26] C. Kim and K. Roy, "Dynamic v/sub t/ sram: a leakage tolerant cache memory for low voltage microprocessors," in *Low Power Electronics and Design, 2002. ISLPED '02. Proceedings of the 2002 International Symposium on*, 2002, pp. 251–254.

[27] A. Wang, A. Chandrakasan, and S. Kosonocky, "Optimal supply and threshold scaling for subthreshold cmos circuits," in *VLSI, 2002. Proceedings. IEEE Computer Society Annual Symposium on*, 2002, pp. 5–9.

[28] C.-I. Kim, H. Soeleman, and K. Roy, "Ultra-low-power DLMS adaptive filter for hearing aid applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 6, pp. 1058–1067, 2003.

[29] C. Enz, F. Krummenacher, and E. Vittoz, "An analytical mos transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications," *Analog Integrated Circuits and Signal Processing*, vol. 8, pp. 83–114, July 1995.

[30] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," in *Proceedings of the 41st Annual Conference on Design Automation*. ACM New York, NY, USA, 2004, pp. 868–873.

[31] B. Calhoun and A. Chandrakasan, "Characterizing and modeling minimum energy operation for subthreshold circuits," in *Low Power Electronics and Design, 2004. ISLPED '04. Proceedings of the 2004 International Symposium on*, 2004, pp. 90–95.

[32] B. Calhoun, A. Wang, and A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *Solid-State Circuits, IEEE Journal of*, vol. 40, no. 9, pp. 1778–1786, 2005.

[33] O. Akgun and Y. Leblebici, "Energy Efficiency Comparison of Asynchronous and Synchronous Circuits Operating in the Sub-Threshold Regime," *J. Low Power Electronics*, vol. 3, no. 3, pp. 320–336, 2008.

[34] B. Paul, A. Raychowdhury, and K. Roy, "Device optimization for digital subthreshold logic operation," *IEEE Transactions on Electron Devices*, vol. 52, no. 2, pp. 237–247, 2005.

[35] A. Raychowdhury, B. Paul, S. Bhunia, and K. Roy, "Computing with subthreshold leakage: device/circuit/architecture co-design for ultralow-power subthreshold operation," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 11, pp. 1213–1224, 2005.

[36] S. Hanson, M. Seok, D. Sylvester, and D. Blaauw, "Nanometer device scaling in subthreshold circuits," in *Design Automation Conference, 2007. DAC'07. 44th ACM/IEEE*, 2007, pp. 700–705.

[37] N. Weste and D. Harris, *CMOS VLSI Design.* Pearson Addison Wesley, NY, USA, 2005.

[38] A. Raychowdhury, S. Mukhopadhyay, and K. Roy, "A feasibility study of subthreshold SRAM across technology generations," in *2005 IEEE International Conference on Computer Design: VLSI in Computers and Processors, 2005. ICCD 2005. Proceedings*, 2005, pp. 417–422.

[39] J. Chen, L. Clark, and T. Chen, "An ultra-low-power memory with a subthreshold power supply voltage," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 10, pp. 2344–2353, 2006.

[40] J. P. Kulkarni, K. Kim, and K. Roy, "A 160 mv robust schmitt trigger based subthreshold SRAM," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 10, pp. 2303–2313, Oct. 2007.

[41] B. Calhoun and A. Chandrakasan, "A 256-kb 65-nm sub-threshold SRAM design for ultra-low-voltage operation," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 3, pp. 680–688, 2007.

[42] B. Zhai, S. Hanson, D. Blaauw, and D. Sylvester, "Analysis and mitigation of variability in subthreshold design," in *Proceedings of the 2005 international symposium on Low power electronics and design*. ACM New York, NY, USA, 2005, pp. 20–25.

[43] J. Kwong and A. Chandrakasan, "Variation-driven device sizing for minimum energy sub-threshold circuits," in *Proc. of the 2006 International symposium on Low power electronics and design*. ACM New York, NY, USA, 2006.

[44] B. Calhoun and A. Chandrakasan, "Ultra-dynamic voltage scaling (UDVS) using sub-threshold operation and local voltage dithering," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 1, pp. 238–245, 2006.

[45] Y. Ramadass and A. Chandrakasan, "Minimum energy tracking loop with embedded DC-DC converter enabling ultra-low-voltage operation down to 250 mV in 65 nm CMOS," *IEEE Journal of Solid State Circuits*, vol. 43, no. 1, p. 256, 2008.

[46] A. Wang and A. Chandrakasan, "A 180-mV subthreshold FFT processor using a minimum energy design methodology," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 1, pp. 310–319, 2005.

[47] S. Hauck, "Asynchronous design methodologies: An overview," *Proceedings of the IEEE*, vol. 83, no. 1, pp. 69–93, 1995.

[48] C. Van Berkel, M. Josephs, and S. Nowick, "Applications of asynchronous circuits," *Proceedings of the IEEE*, vol. 87, no. 2, pp. 223–233, 1999.

[49] V. Tiwari, D. Singh, S. Rajgopal, G. Mehta, R. Patel, and F. Baez, "Reducing power in high-performance microprocessors," in *Design Automation Conference, 1998. Proceedings*, 1998, pp. 732–737.

[50] L. Clark, E. Hoffman, J. Miller, M. Biyani, Y. Liao, S. Strazdus, M. Morrow, K. Velarde, and M. Yarch, "An embedded 32-b microprocessor core for low-power and high-performance applications," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 11, pp. 1599–1608, 2001.

[51] L. Nielsen and J. Sparso, "Designing asynchronous circuits for low power: an IFIR filter bankfor a digital hearing aid," *Proceedings of the IEEE*, vol. 87, no. 2, pp. 268–281, 1999.

[52] P. Beerel and M. Roncken, "Low Power and Energy Efficient Asynchronous Design," *J. Low Power Electronics*, vol. 3, no. 3, pp. 234–253, 2007.

[53] M. Renaudin, "Asynchronous circuits and systems: a promising design alternative," *Microelectron. Eng.*, vol. 54, no. 1-2, pp. 133–149, 2000.

[54] A. Taubin, J. Cortadella, L. Lavagno, A. Kondratyev, and A. Peeters, "Design automation of real-life asynchronous devices and systems," *Foundations and Trends in Electronic Design Automation*, vol. 2, no. 1, pp. 1–133, 2007.

[55] A. Martin, "The limitations to delay-insensitivity in asynchronous circuits," California Institute of Technology, Tech. Rep., 1990.

[56] J. Cortadella, A. Kondratyev, L. Lavagno, and C. Sotiriou, "Desynchronization: Synthesis of asynchronous circuits from synchronous specifications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 10, pp. 1904–1921, 2006.

[57] K. Fant, S. Brandt, T. Inc, and M. St Paul, "NULL Convention Logic TM: a complete and consistent logicfor asynchronous digital circuit synthesis," in *Application Specific Systems, Architectures and Processors, 1996. ASAP 96. Proceedings of International Conference on*, 1996, pp. 261–273.

[58] M. E. Dean, D. L. Dill, and M. Horowitz, "Self-timed logic using current-sensing completion detection (CSCD)," in *Computer Design: VLSI in Computers and Processors, 1991. ICCD '91. Proceedings., 1991 IEEE International Conference on*, Oct. 1991, pp. 187–191.

[59] E. Grass and S. Jones, "Asynchronous circuits based on multiple localised current-sensing completion detection," in *Asynchronous Design Methodologies, 1995. Proceedings., Second Working Conference on*, May 1995, pp. 170–177.

[60] H. Lampinen and O. Vainio, "Circuit design for current-sensing completion detection," in *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, vol. 2, May-Jun 1998, pp. 185–188.

[61] ——, "Dynamically biased current sensor for current-sensing completion detection," in *Circuits and Systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on*, vol. 4, May 2001, pp. 394–397.

[62] C. Myers, *Asynchronous circuit design.* Wiley-Interscience, 2004.

[63] R. Corless, G. Gonnet, D. Hare, D. Jeffrey, and D. Knuth, "On the LambertW function," *Advances in Computational Mathematics*, vol. 5, no. 1, pp. 329–359, 1996.

[64] A. Robbins, *Effective awk programming.* O'Reilly & Associates, Inc. Sebastopol, CA, USA, 2001.

[65] M. Hansen, H. Yalcin, J. Hayes, D. Syst, and I. Kokomo, "Unveiling the ISCAS-85 benchmarks: a case study in reverseengineering," *IEEE Design & Test of Computers*, vol. 16, no. 3, pp. 72–80, 1999.

[66] N. Lotze, M. Ortmanns, and Y. Manoli, "A study on self-timed asynchronous subthreshold logic," in *Computer Design, 2007. ICCD 2007. 25th International Conference on*, Oct. 2007, pp. 533–540.

[67] R. Elmqvist, J. Landegren, S. Petersson, A. Senning, and G. W. Ollson, "Artificial pacemaker for treatment of adams-stokes syndrome and slow heart rate," *Am Heart J*, no. 65, pp. 731–748, 1963.

[68] M. Astrom, "Detection and classification in electrocardiac signals," Ph.D. dissertation, Lund University, 2003.

[69] A. Poularikas, *The transforms and applications handbook*, 2nd ed., 2000.

[70] M. Åström, S. Olmos, and L. Sörnmo, "Wavelet-based event detection in implantable cardiac rhythm management devices," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 3, March 2006.

[71] K. Parhi, *VLSI Digital Signal Processing.* Wiley, 1999.

[72] B. Parhami, *Computer Arithmetic.* 198 Madison Avenue, NY, USA: Oxford University Press, 2000.

[73] J. Rodrigues, L. Olsson, T. Sörnmo, and V. Öwall, "Digital implementation of a wavelet-based event detector for cardiac pacemakers," *IEEE Transactions on Circuits and Systems I: Regular Papers,*, vol. 52, no. 12, pp. 2686–2698, Dec. 2005.

[74] M. Seok, D. Sylvester, and D. Blaauw, "Optimal technology selection for minimizing energy and variability in low voltage applications," in *ISLPED '08: Proceeding of the thirteenth international symposium on Low power electronics and design.* New York, NY, USA: ACM, 2008, pp. 9–14.

[75] K. A. Bowman, S. G. Duvall, and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 2, pp. 183–190, Feb. 2002.

[76] D. Bol, R. Ambroise, D. Flandre, and J. Legat, "Analysis and minimization of practical energy in 45nm subthreshold logic circuits," in *IEEE International Conference on Computer Design, 2008. ICCD 2008*, 2008, pp. 294–300.

[77] A. Chandrakasan, S. Sheng, and R. Brodersen, "Low-power CMOS digital design," *Solid-State Circuits, IEEE Journal of*, vol. 27, no. 4, pp. 473–484, 1992.

[78] V. Sze and A. Chandrakasan, "A 0.4-V UWB baseband processor," *Proceedings of the 2007 international symposium on Low power electronics and design*, pp. 262–267, 2007.

[79] E. Brunvand, "Low latency self-timed flow-through FIFOs," in *Proceedings of the 16th Conference on Advanced Research in VLSI (ARVLSI'95).* IEEE Computer Society Washington, DC, USA, 1995.

[80] N. Kim, T. Kgil, K. Bowman, V. De, and T. Mudge, "Total power-optimal pipelining and parallel processing under process variations in nanometer technology," in *Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, 2005, pp. 535–540.

[81] V. Srinivasan, D. Brooks, M. Gschwind, P. Bose, V. Zyuban, P. Strenski, P. Emma, I. Center, and N. Yorktown Heights, "Optimizing pipelines for power and performance," *Microarchitecture, 2002.(MICRO-35). Proceedings. 35th Annual IEEE/ACM International Symposium on*, pp. 333–344, 2002.

[82] *Design Compiler Register Retiming Reference Manual*, C-2009.06 ed., 2009.

[83] M. Bikumandla, J. Ramirez-Angulo, C. Urquidi, R. G. Carvajal, and A. J. Lopez-Martin, "Biasing cmos amplifiers using mos transistors in subthreshold region." *IEICE Electronic Express*, vol. 1, no. 12, pp. 339–345, 2004.

[84] A. S. Sedra and K. C. Smith, *Microelectronic Circuits*, 4th ed. Oxford University Press, 1998.

[85] J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavagno, and A. Yakovlev, "Hardware and petri nets: Application to asynchronous circuit design," *Lecture Notes in Computer Science*, pp. 1–15, 2000.

[86] M. Kishinevsky, J. Cortadella, and A. Kondratyev, "Asynchronous interface specification, analysis and synthesis," in *Proceedings of the 35th annual conference on Design automation*. ACM New York, NY, USA, 1998, pp. 2–7.

[87] T. Chu, C. Logic, and C. Fremont, "CLASS: a CAD system for automatic synthesis and verification of asynchronous finite state machines," in *System Sciences, 1993, Proceeding of the Twenty-Sixth Hawaii International Conference on*, vol. 1, 1993.

[88] A. Semenov, A. Koelmans, L. Lloyd, and A. Yakovlev, "Designing an asynchronous processor using Petri nets," *IEEE Micro*, vol. 17, no. 2, pp. 54–64, 1997.

[89] R. Zurawski and M. Zhou, "Petri nets and industrial applications: A tutorial," *IEEE Transactions on Industrial Electronics*, vol. 41, no. 6, pp. 567–583, 1994.

[90] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[91] E. Sentovich, K. Singh, L. Lavagno, C. Moon, R. Murgai, A. Saldanha, H. Savoj, P. Stephan, R. Brayton, and A. Sangiovanni-Vincentelli, "SIS: A system for sequential circuit synthesis," Citeseer, Tech. Rep., 1992.

[92] S. Furber and P. Day, "Four-phase micropipeline latch control circuits," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 4, no. 2, pp. 247–253, 1996.

[93] J. Sauerbrey, T. Tille, D. Schmitt-Landsiedel, and R. Thewes, "A 0.7-v MOSFET-only switched-opamp /spl sigma//spl delta/ modulator in standard digital CMOS technology," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 12, pp. 1662–1669, Dec. 2002.

[94] D. E. Muller and W. S. Bartky, "A theory of asynchronous circuits," in *Proceedings of an International Symposium on the Theory of Switching*. Harvard University Press, 1959, pp. 204–243.

[95] I. Sutherland, "Micropipelines," *Communications of the ACM*, vol. 32, no. 6, Jul. 1989.

[96] A. J. Martin, "Formal program transformations for vlsi circuit synthesis," pp. 59–80, 1990.

[97] M. Shams, J. Ebergen, and M. Elmasry, "A comparison of cmos implementations of an asynchronous circuits primitive: the c-element," in *ISLPED '96: Proceedings of the 1996 international symposium on Low power electronics and design*. Piscataway, NJ, USA: IEEE Press, 1996, pp. 93–96.

[98] *Cadence Common Power Format Language Reference*, Version 1.0 Extended ed., June 2008.

Curriculum Vitae

Omer Can Akgun

Address

Chemin de Somais, 17
1009, Pully
Switzerland
Phone: +41-76-326.1649
Email: omercan.akgun@epfl.ch

Personal Details

Gender: Male
Date of birth: 18th of June, 1980
Place of birth: Istanbul, Turkey
Present Citizenship: Turkish

Education

01/2005-11/2009 **Swiss Federal Institute of Technology**
Lausanne, Switzerland
Doctor of Philosophy in Electrical Engineering,
Thesis title: *Energy Efficiency Enhancement of Sub-threshold Digital CMOS - Modeling, Technology Selection, and Architectural Exploration*
Thesis Defense Date: 4 November 2009

05/2009-06/2009 **Lund University**
Lund, Sweden
Visiting PhD Student - Researcher

10/2008-03/2009 **Technical University of Denmark**
Copenhagen, Denmark
Visiting PhD Student

09/2002-03/2004 **The Ohio State University**
Columbus, OH, USA
Master of Science in Electrical Engineering
Thesis title: *Design Approaches for Low-Power Reconfigurable ADCs*

09/1998-06/2002 **Istanbul Technical University**
Istanbul, Turkey
Bachelor of Science in Electronics and Telecommunication Engineering
Thesis title: *Integrated Circuit Implementation of Voltage Controlled LC Tank Oscillators*

Publications

• *Energy Dissipation Reduction of a Cardiac Event Detector in the Sub-Vt Domain By Architectural Folding*, J. Rodrigues, O. C. Akgun, P. Acharya, A. D. L. Calle, Y. Leblebici and V. Owall. In Proceedings of PATMOS 2009, to appear.

• *A Current Sensing Completion Detection Method for Asynchronous Pipelines Operating in the Sub-threshold Regime*, O. C. Akgun, F. K. Gurkaynak and Y. Leblebici. International Journal of Circuit Theory and Applications, 37(2):203-220, 2009. Invited paper to special issue.

• *Energy Efficiency Comparison of Asynchronous and Synchronous Circuits Operating in the Sub-Threshold Regime*, O. C. Akgun and Y. Leblebici. Journal of Low Power Electronics, 3:320-336, 2008.

• *A 1.8V 12-bit 230-MS/s Pipeline ADC in 0.18um CMOS Technology*, T. Liechti, A. Tajalli, O. C. Akgun, Z. Toprak, Y. Leblebici. In IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), pages 21-24. IEEE, 2008.

• *Current Sensing Completion Detection for Subthreshold Asynchronous Circuits.*, O. C. Akgun, Y. Leblebici, and E. A. Vittoz. In European Conference on Circuit Theory and Design, 2007.

• *Design of Completion Detection Circuits for Self-timed Systems Operating in Sub-threshold Regime.*, O. C. Akgun, Y. Leblebici, and E. A. Vittoz. In IEEE 3rd Conference on Ph.D. Research in Microelectronics and Electronics, Bordeaux, pages 241-244, 2007.

• *Weak Inversion Performance of CMOS and DCVSPG Logic Families in Sub-300mV Range.*, O. C. Akgun and Y. Leblebici. In IEEE International Symposium on Circuits and Systems, Kos, pages 1251-1254, 2006.

• *An Active-RC Reconfigurable Lowpass-Polyphase Tow-Thomas Biquad Filter.*, A. Yamazaki, A. Ravindran, O. Akgun, and M. Ismail. In 47th Midwest Symposium on Circuits and Systems, pages 57-60, 2004.

• *A Reconfigurable Pipelined ADC with Linear Power-Speed Scaling*, O. C. Akgun, A. Savla and M. Ismail, accepted for publication at Journal of Analog Integrated Circuits and Signal Processing, Kluwer Academic Publishers.