2012

Mixed-Signal Carry Look-Ahead Adder with Constant Power for Cryptographic Applications

Ashley Novak

Mixed-Signal Carry Look-Ahead Adder with Constant Power for Cryptographic Applications

by

Ashley Novak

A Thesis
Submitted to the Faculty of Graduate Studies through the
Department of Electrical and Computer Engineering in Partial Fulfillment
of the Requirements for the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada
2012

Mixed-Signal Carry Look-Ahead Adder with Constant Power for Cryptographic Applications

by

Ashley Novak

APPROVED BY:

Dr. M. Mirhassani, Advisor
Electrical and Computer Engineering

Dr. H. Wu
Electrical and Computer Engineering

Dr. D. Ting
Mechanical Automotive and Materials Engineering

, Chair of Defense

University of Windsor

June, 2012

Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyones copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

Due to the ubiquity of electronic communication systems in consumers' lives, it is necessary to ensure that the sensitive information being transmitted is not accessible by malicious parties. Because of advancements in technology, it is now possible to easily steal data from these electronic systems, even if they are protected by a strong encryption algorithm. These security threats, known as Side Channel Attacks, have exposed weaknesses in the hardware architectures of the systems meant to be secure.

This research explores a novel method of designing a crypto processor component, the adder, which allows it to produce minimal side channel information, rendering it less vulnerable in terms of hardware. The results show that it is possible to maintain a competitively low power consumption, as compared to conventional architectures, all while providing a method to greatly improve data security systems.

I would like to dedicate this work to my family and friends. I thank you for the support and good advice, and will cherish the company that I was granted on some of the all-nighters that I enjoyed completing this work. Thank you.

Acknowledgments

I will always be grateful to my supervisor, Dr. Mirhassani, for advice, both academic and in life.

Thank you also to my other committee members, Dr. H. Wu and Dr. D. Ting, for their support in this research.

Contents

Declaration of Originality iv

Abstract v

Dedication vi

Acknowledgments vii

List of Figures xi

List of Tables xiii

List of Abbreviations xiv

1 Introduction 1

1.1 History of Side Channel Attacks 2

1.2 Overview of Research, Motivation 4

1.3 Organization of Thesis 5

2 Cryptography and Cryptanalysis 6

2.1 Types of Encryption Schemes 6

2.1.1 Secret-Key Cryptography 7
2.1.2 Public-Key Cryptography 10
2.2 Security Weaknesses 16
2.2.1 Physical Attacks 17
2.2.2 Side Channel Attacks 18
2.3 Countermeasures: Hardware Versus Software 19
2.4 Summary 20

3 Countermeasures: Circuit Architecture 21
3.1 Masking vs. Hiding 22
3.2 Circuit Architectures 23
3.2.1 Random Switching Logic 24
3.2.2 Sense Amplifier Base Logic 24
3.2.3 Wave Dynamic Differential Logic 25
3.2.4 Masked Dual-Rail Pre-charged Logic 25
3.2.5 Dual-Rail Random Switching Logic 26
3.2.6 Multiple-Valued Source-Coupled Logic 27
3.3 Summary 27

4 Proposed Circuit Architecture 29
4.1 Montgomery Multiplication 30
4.2 Mixed Signal Carry Look-Ahead Adder 35
4.3 Characteristics of the Proposed DAC 37
4.4 Characteristics of the Proposed Carry Generator 41
4.5 Characteristics of the Proposed ADC 45

5 Results **49**

5.1 Digital-to-Analog Converter 50

5.2 Mixed Signal Carry Generator 54

5.3 Analog-to-Digital Converter 58

5.4 Comprehensive Circuit Results 61

6 Conclusion and Recommendations **63**

References **66**

Vita Auctoris **70**

List of Figures

1.1 The Principle of Cryptography 2

1.2 Side Channel Attack 3

2.1 Secret-key Cryptography 8

2.2 Encryption and Decryption using 3DES 10

2.3 Public-key Cryptography 11

2.4 Data Encryption and Decryption Using RSA 12

2.5 Signature Encryption and Decryption Using RSA 14

3.1 Countermeasure Techniques: (a)Masking (b)Hiding 23

4.1 Modular Multiplication Using Montgomery Method 32

4.2 An Adder as Research Focus 34

4.3 Block Diagram of the Proposed Full Circuit 36

4.4 Signal Propagation of Proposed Design 37

4.5 Gate Level Diagram of the DAC and Carry Generator blocks 39

4.6 Analog Equivalent Signal and its Complement 40

4.7 Block Diagram of Proposed ADC Design 45

5.1 Transistor Level Diagram of One a One Bit Conversion 51
5.2 4-bit Input A: Constant Value 53
5.3 4-bit Input B: Incrementing Value 53
5.4 Power Consumption of two 32bit Digital to Analog Conversions (64 bits) 54
5.5 Static versus Dynamic Logic 55
5.6 Transistor Level Diagram of the MV-CMDL Carry Generator 57
5.7 Power Consumption of the Carry Generator block 58
5.8 Transistor Level Diagram of a 4-bit ADC 59
5.9 Power Consumption of two 32-bit Analog to Digital Conversions (64 bits) 61
5.10 Overall Circuit Power Consumption 62

List of Tables

4.1 Modular Multiplication Using Classical Evaluation Method 31

4.2 Logic for Signal Generation 43

4.3 Logic for the Generation of the Two Least Significant Bits 47

5.1 G_i and P_i Generation Logic 56

5.2 Function of the $SELECT$ signal 60

5.3 2-bit DAC Conversion Equivalents 60

5.4 Results Comparison 62

List of Abbreviations

ADC	Analog-to-Digital Converter
AES	Advance Encryption Standard
ASIC	Application-Specific Integrated Circuit
CLA	Carry Look-ahead Adder
DAC	Digital-to-Analog Converter
DES	Data Encryption Standard
DPA	Differential Power Analysis
DRSL	Dual Rail Random Switching Logic
MDPL	Masked Dual Rail with Pre-charge Logic
MV-CML	Multiple-value Current Mode Logic
RSA	Rivest, Shamir and Adleman
RSL	Random Switching Logic
SABL	Sense Amplifier Base Logic
SPA	Simple Power Analysis
WDDL	Wave Dynamic Differential Logic

Chapter 1

Introduction

There is evidence of cryptography dating as far back as 4000 years ago [2]. Since then, people have been searching for increasingly complicated methods of disguising data from all but the intended recipients. Beginning with hieroglyphics in the ancient Egyptian era, continuing to digital communication that is favored today, the study and application of cryptography has an inexhaustible demand for development fueled by the equally enduring field of cryptanalysis (deciphering hidden data). Figure 1.1 shows a general representation of cryptography principles. A typical message is encoded based on an algorithm to disguise the data, sent over a channel to the recipient, then decrypted by this recipient to reveal the original message.

Just as traditional methods for securing data involved applying mathematical algorithms, familiar approaches to deciphering encoded messages were performed us-

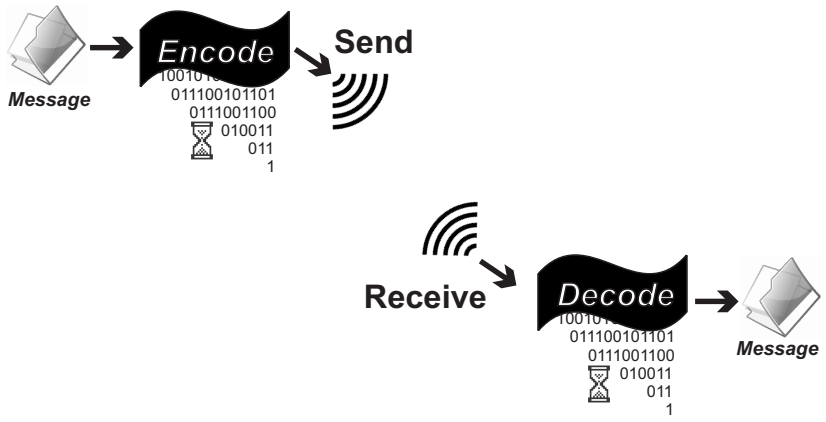

Figure 1.1: The Principle of Cryptography

ing equally or often additionally complicated algorithms. As technology progresses, computational devices used to secure and transmit data become increasingly fast. This same rapid technology allows for quicker decoding as well as more sophisticated attacks that may be accomplished without direct access to the encrypting device.

1.1 History of Side Channel Attacks

Side Channel Attacks may be considered a highly covert assault on electronic systems. It is a sophisticated means of exploiting crypto system hardware weaknesses and revealing critical data. This type of attack, contrary to the long-established software hacking method, is aimed at the physical implementation of the system itself, the hardware. Side Channel Attacks pose a major threat to data security, and to counter measure these attacks, new and innovative specifically designed processing hardware

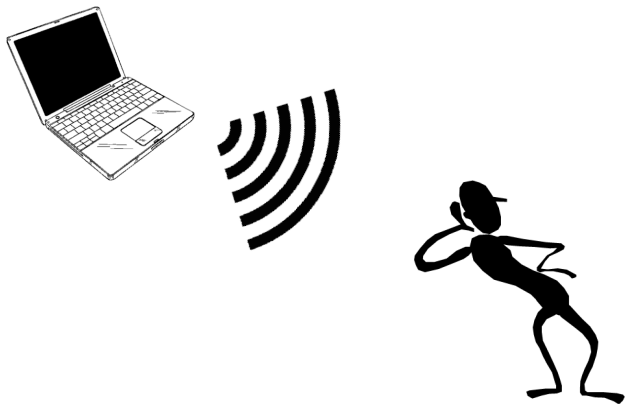

Figure 1.2: Side Channel Attack

is required to mitigate such attacks.

Variations of the Side Channel Attack have been reported since 1965 when the British intelligence agency, MI5, observed the sounds made by the Egyptian embassys rotor-cipher machine, and subsequently were able to decipher their messages [35]. Since then, the Side Channel Attack had not been considered as a major threat against the securities of modern technology, until the last decade when it started regaining recognition. Modern advancements in technology allow the speed and precision necessary for highly sensitive equipment to exist, and thus be misused to overcome barriers of privacy.

Side Channel Attacks focus on imperceptibly accessing information directly from the hardware itself, rather than algorithmic or brute force attacks targeted at software, as the simplified depiction in Figure 1.2 implies. Such information as power consumption, electromagnetic radiation and timing patterns are recorded using elab-

orate measuring devices and then analyzed to deduce the encryption key. Once the key is discovered, all further communications which are encrypted using this key are easily exposed.

1.2 Overview of Research, Motivation

The variety of applications in which electronic systems are being used is continually increasing. We have come to rely greatly on electronics to further enhance the convenience of daily life concerning such areas as communication, banking, and transportation among others. These applications sometimes involve extremely sensitive information, and other times have a significant correlation to human safety. Due to the increase in known security breach methods, research for more secure electronic systems, especially pertaining to the circuitry level, is becoming popular in academia [12] [10] [6].

The focus of this research is to present a novel circuit design approach to be used in cryptographic processors for the purpose of securing data by minimizing side channel information leakage. As new and improved algorithms are generally the focus in the field of cryptography, it is essential that hardware specialists explore methods of circuit implementation which may rise to the challenge posed by the ingenuity of new highly sensitive measuring equipment.

This thesis will demonstrate the capability of designing a crypto processor able to withstand Side Channel Attacks in the form of Power Analysis. It aims to minimize side channel information as a whole, if not eliminate it, and to reduce the availability of data to be acquired then subsequently analyzed, leading to a breach of security.

The novel crypto processor design approach, presented in this research, combines the advantages of using analog signals along with a new dual-rail multiple valued analog arithmetic, aimed at suppressing the dependence of the power consumption spurs on the data.

1.3 Organization of Thesis

Chapter 1 begins with an introduction to cryptography, as well as a brief description of the side channel attack, which is the main focus of the proposed hardware design presented in this thesis. Chapter 2 elaborates on the fields of cryptology and cryptanalysis, giving examples of popular methods and algorithms studied. Chapter 3 discusses current publicized hardware architectures intended to thwart side channel attacks. Chapter 4 illustrates, by applying examples, Montgomery Multiplication as well as its benefit, explains the theory of the Multiple-Valued Current Mode Logic Carry Look-Ahead adder, then presents the novel hardware design of the adder. Chapter 5 presents the results followed by the conclusion and recommendations in Chapter 6.

Chapter 2

Cryptography and Cryptanalysis

There are two opposing fields of research regarding information security: Cryptology and Cryptanalysis. Cryptology is concerned with securing sensitive information with the aid of increasingly complex algorithms and ASIC design methods. Conversely, the equally enduring study of Cryptanalysis aims to breach, or crack, these security systems. In this chapter, several of the most important and widely used cryptographic algorithms are reviewed.

2.1 Types of Encryption Schemes

Information security is one of the paramount criteria when designing or conceptualizing many electronic systems. When considering algorithmic security, there are typ-

ically two basic types of encryption schemes: Secret-Key Cryptography and Public-Key Cryptography [30].

The main difference between these two schemes lies in the keys, and their distribution. In Secret-Key Cryptography, a single key is used by two parties to both encrypt and then decrypt a message. In Public-Key Cryptography, there are two sets of keys, one public, used for encryption, and one secret, used for decryption.

There are many different encryption algorithms that have been developed over the years for data security. As computer speeds increase, security algorithms must increase in complexity to avert attackers and guard data. After further describing the aforementioned encryption schemes, examples of each are illustrated and explained.

2.1.1 Secret-Key Cryptography

Secret-Key Cryptography, also referred to as Symmetric-Key Cryptography, uses the same key for the encryption as for the decryption of a transmitted message [30] [15]. For this reason, the key must be kept private and secured from the possession of malicious parties. Figure 2.1 illustrates the general process of data encryption and decryption using a Secret-Key Algorithm. The difficulty in this approach lies within the delivery of the secret key. Under the classification of Secret-Key Cryptography, the cipher schemes can further be divided among one of two categories: stream ciphers or block ciphers [15].

In a stream cipher there exists a constant communication regarding the secret-key, which allows it to be continually changed. In this scheme, encryption is only performed on a single word of data at a time. In contrast, block ciphers use the same key

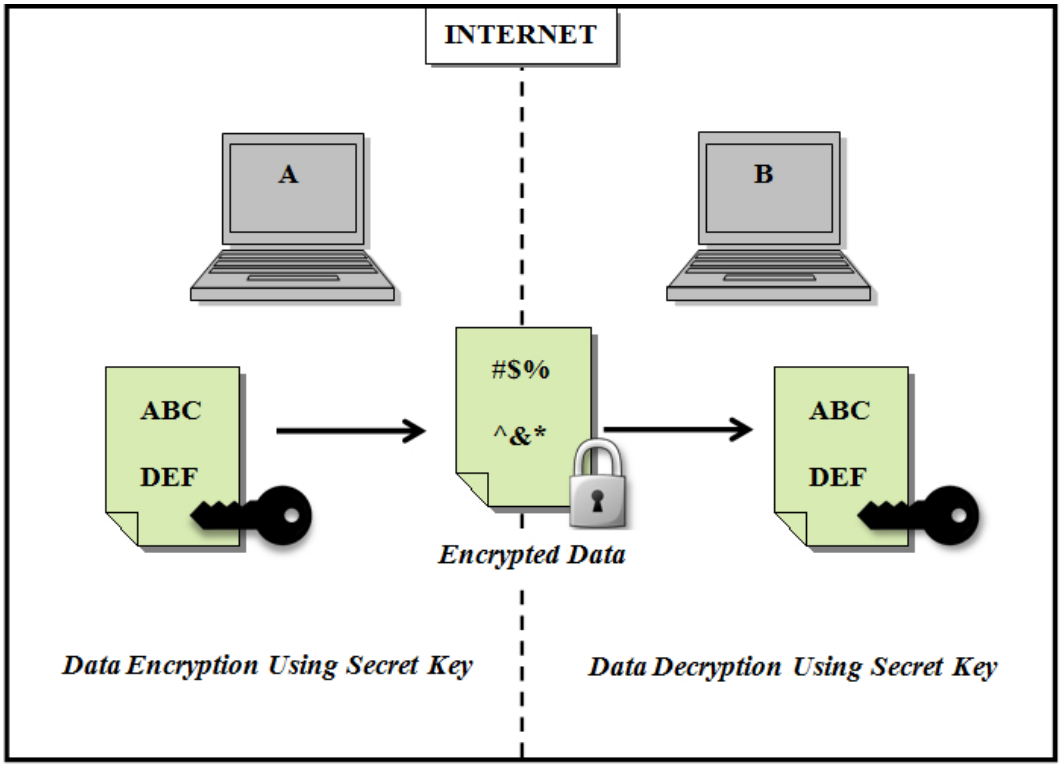

Figure 2.1: Secret-key Cryptography

to encrypt whole blocks of data at once using the same key. Examples of Secret-Key Cryptography include: Data Encryption Standard (DES) [7] and Advanced Encryption Standard (AES) [24], which are both still used today. These schemes are reviewed briefly in the next sections.

Data Encryption Standard

The Data Encryption Standard (DES) [7] was first developed in the IBM laboratories in the early 1970s, and is regarded as playing the substantial role in the advancement of Cryptography. This algorithm uses a key length of 56-bits operating on 64-bit

blocks. As it is a Secret-Key Algorithm, this same 56-bit key is used to both encrypt and decrypt the data.

After a request was made by the National Bureau of Standards in 1973 for an algorithm to protect the governments unclassified data, DES was submitted and accepted in 1977. It was designated an official Federal Information Processing Standard in the United States. After this designation, and the publication of this algorithm, it incurred great academic investigation and scrutiny due to its relatively short key size.

In response to this argument, 3DES [7] was a suggested replacement for the initial algorithm. The general configuration of this algorithm is shown in Figure 2.2. This modification to the algorithm essentially uses each one of 3 different keys. It first encrypts using key 1, decrypts using key 2, and then encrypts the data using key 3, to be sent over the channel. Decryption of this ciphertext is done in the exact opposite order with the corresponding keys.

Advanced Encryption Standard

The Advanced Encryption Standard (AES) [24] is another Secret-Key Algorithm which became the official successor to the previously used 3DES as of December of 2001. Its development was also primarily for governmental applications and was created by two Belgian Cryptography experts Joan Daemen and Vincent Rijmen.

In contrast to the very short key size employed by DES, AES allows a variable key length of 128, 192, or 256 bits with equivalent block lengths, and thus proved more successful against brute-force attacks. Among its advantages, the Advanced Encryption Algorithm has significantly improved efficiency in terms of processing time

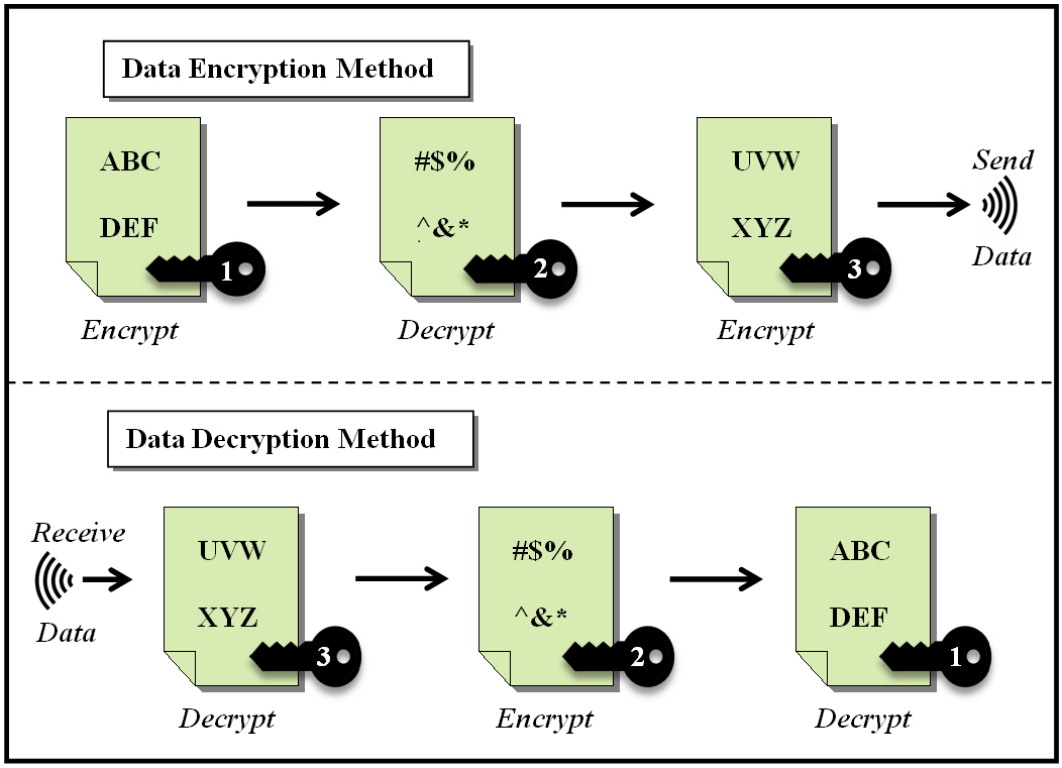

Figure 2.2: Encryption and Decryption using 3DES

[5] as well as greater security strength than that of the Data Encryption Standard.

2.1.2 Public-Key Cryptography

Public-Key Cryptography, also referred to as Asymmetric-Key Cryptography [30] [15], is a dual key system which uses one key for encrypting a message and a second key to decrypt this message.

The key used for data encryption is a public key that is visible to anybody who wishes to use it, however, the key used for decryption is known only by the recipient of the encoded message. Figure 2.3 shows the general concept behind this scheme.

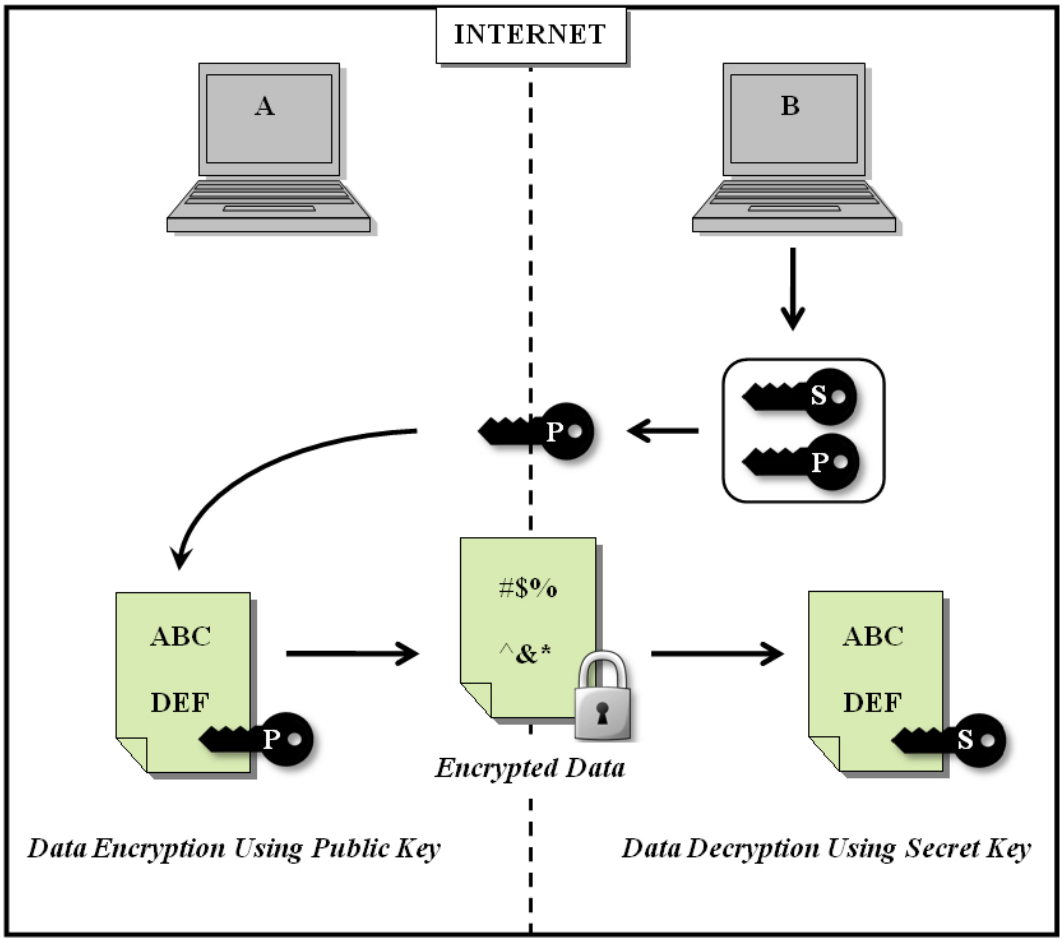

Figure 2.3: Public-key Cryptography

The reasoning exploited for the effectiveness of the Public-Key Cryptography scheme is based on the existence of one-way functions. These are functions of which the inverse is nearly impossible to compute, thus they are ideal for the use in encryption algorithms. Of the Public-Key Algorithms, RSA [26] (named after its creators) was one of the first, and remains the most popular. The RSA algorithm, due to its continued popularity, is the assumed algorithm for which our proposed novel hardware design was created.

RSA Algorithm

RSA is a Public-Key Algorithm created by, and named after, the MIT scholars Rivest, Shamir and Adleman [26]. RSA requires the use of one key to encrypt sensitive data and a second different key to decrypt. The key strength of this algorithm is its computational complexity. The RSA algorithm has the useful property that the keys are commutative; this means, either of the two different keys (secret or public) may be used to encrypt the data, while the opposite may be used to decrypt the data, this is useful in sending a Signature.

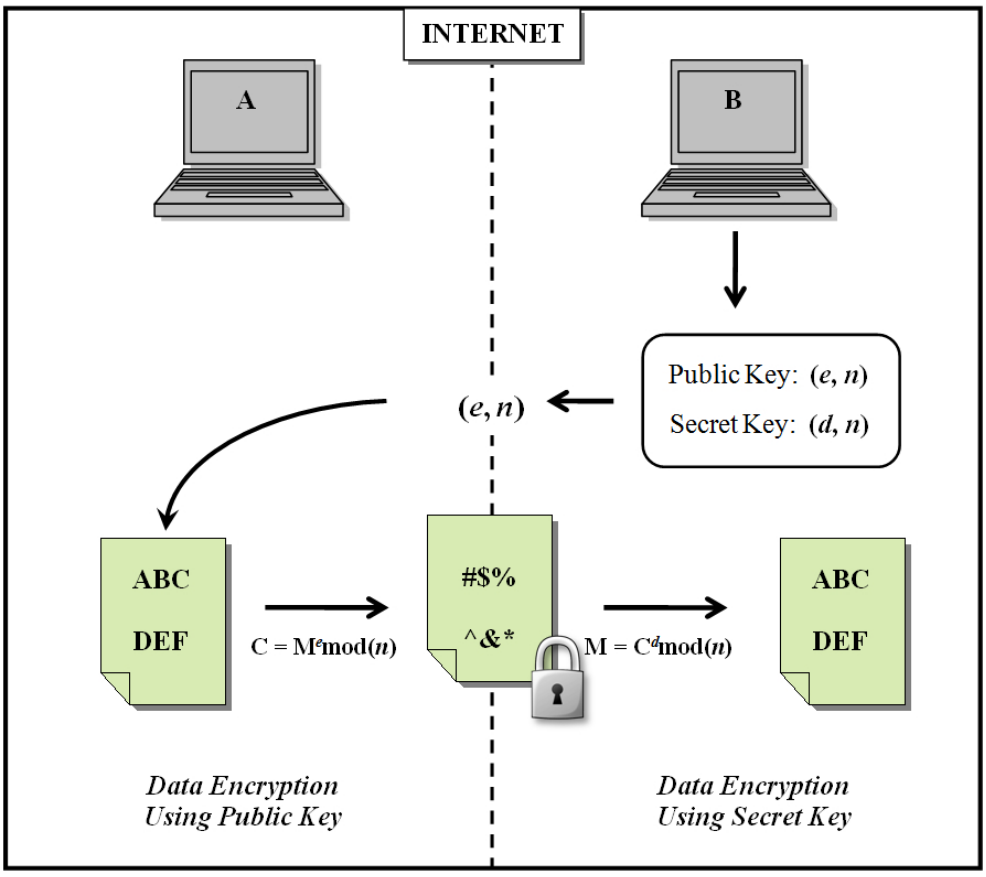

Figure 2.4: Data Encryption and Decryption Using RSA

In order to explain the RSA algorithm, we consider two parties: one which sends an encrypted message (Sender), and one who is intended to receive this message (Receiver). The idea behind this Public-Key Algorithm is that the party that will be receiving sensitive data, Receiver, will generate two keys; one private key that is used to decrypt the data and is seen only by the Receiver, and one public key that is used to encrypt the data, see Figure 2.4. This algorithm may also be used, as previously stated, to send an electronic signature, as seen in Figure 2.5. The private-key holder may encode a signature and send it, and the public-key (meaning everyone) is able to decode this signature. This authentication process works on the premise that; if the public-key can decode the signature, the signature must have only been encrypted using the secret-key. Since there is only one party who has possession of this secret-key, he must be the one who wrote the signature. This signature concept is illustrated in 2.5.

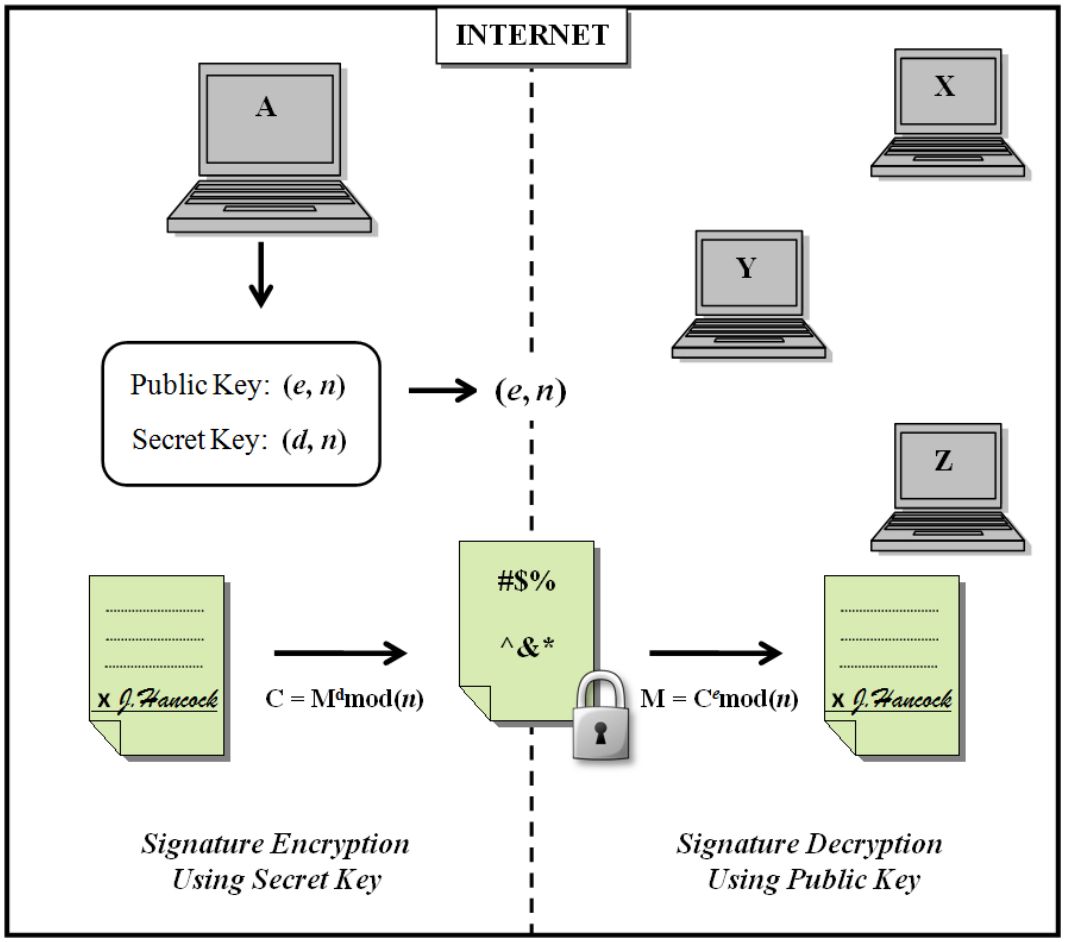

Figure 2.5: Signature Encryption and Decryption Using RSA

The Receivers public key is readily available for anyone to access, and is used to encrypt a message to be sent over a network to the Receiver. Only the secret-key can be used to decrypt this data, thus, only the Receiver will have access to the decoded message. Basically, this Public-Key algorithm ensures that the encrypted data remain secured as long as the private key is kept secret.

RSA is still the most widely used and accepted Public-Key Encryption scheme to date since its development in 1977. Due to its continued popularity and reliability,

the hardware design presented in this thesis was based on the RSA scheme.

To obtain the RSA keys, the following steps should be taken. The first step is to choose two prime numbers, p and q, such that their product, n, is of the adequate length, generally 1024 bits, (n=pq). The large bit length of n provides the mathematical complexity that leads to the infeasibility of evaluation by malicious parties.

Once p and q are chosen, the parameter ϕ is obtained as follows:

$$\phi = (p-1)(q-1) \tag{2.1}$$

At this point, the public key may be completed by choosing an integer, e, such that $1 < e < \phi$ and the greatest common denominator between e and ϕ is 1.

The resulting public key is (e, n). The secret exponent, d, is chosen such that $1 < d < \phi$ and it satisfies Equation (2.2):

$$ed = 1(mod\phi) \tag{2.2}$$

The resulting private key is (d, n). The parameter n is referred to as the modulus, e, the encryption exponent, and d, the secret exponent or the decryption exponent.

The secret key remains hidden to anyone but the party who generated the keys, which is an efficient and effective way to decode the cipher text. The encryption algorithm is demonstrated as in Equation (2.3) below, and the decrypt as in (2.4).

$$C = M^e mod(n) \tag{2.3}$$

$$M = C^d mod(n) \tag{2.4}$$

In this algorithm, C represents the ciphertext, or in the encrypted message, M is the original message, and (e, n) is the public-key, (d, n) is the secret key. Figure 2.4 presents the RSA algorithm.

Since n, the modulus, is typically chosen to be at least 1024 bits long, calculating a modular exponentiation, as is found in the RSA algorithm, becomes a very arduous and lengthy process. In the next section, weaknesses of encryption algorithms are reviewed.

2.2 Security Weaknesses

With technology development, more complex and innovative methods for cryptanalysis are emerging. There are weaknesses in every cryptological method, whether it is in its computational strength, its logical operation, or its hardware implementation. The attacks on these systems may be done through software, physically, or through Side Channel Attacks [14] [19]. Though it is infeasible to know all of the conceivable methods of breaching the data security measures, it is possible to design systems with the capability of avoiding the attacks that are currently in play.

There are several different approaches that may be followed to breach data security, such as:

- Software Attacks: These attacks exploit the algorithmic weaknesses, software implementation faults, or protocol vulnerabilities in the communication channel.

- Fault Generation: This method employs knowledge of the systems normal conditions, so that after generating a fault, the attacker may gain access to this system.

- Microprobing: This requires direct access to the device to be able to measure and observe by way of sophisticated tools, as well as manipulate the system.

- Side Channel Attacks: These are performed by monitoring analog characteristics of a system without requiring direct access to the device.

In the past, software or algorithmic attacks were the predominant method of infiltration and security breaches. Within the past decade however, Physical and Side Channel Attacks have become an increasingly threatening means of acquiring critical data.

There are generally two categories in which to classify these types of attacks: invasive and non-invasive. Invasive attacks require direct access to the hardware and are performed using probes and high tech machinery. Generally, the invasive approach is expensive in terms of both equipment and time required to execute, however, the information that may be extracted is greatest using this method. Non-invasive attacks are done using tools to remotely monitor the device, and then exceedingly intelligent methods are used to infer the coded data.

2.2.1 Physical Attacks

Physical attacks are an example of an invasive attack [19]. They require direct access to the hardware itself, including the depackaging of the chip, allowing the attacker

to specifically probe the system to obtain critical information. Though this type of attack is relatively expensive and complex to orchestrate, it is useful in obtaining the necessary details of a device to be able to design less expensive, non-invasive, subsequent attacks on similar devices. A simple example of a physical attack is to connect an external wire to a data bus to eavesdrop on data transfers.

2.2.2 Side Channel Attacks

Side Channel Attacks are classified as non-invasive assaults. This means, physical contact with the device is not required. Attacks are performed using sensitive measuring devices that are able to obtain side channel information while being at a distance from the crypto processor device. Modern Side Channel Attacks can be performed through several different approaches, taking advantage of such information as: power consumption [25], operation timing [17], electromagnetic emissions [11], sound [29], and vibrations.

Side Channel Attacks are accomplished by making assumptions on sensitive parameters based on observations made through measuring various side channel data. Side channel data is defined as the activity of the system which produces information discernible to measuring equipment.

The research presented herein focuses on defending against Power Analysis Attacks. There are predominantly two types of power attacks: Simple Power Analysis (SPA) [34] and Differential Power Analysis (DPA) [18].

The SPA strike is easier to execute compared to the DPA, however, it is more time consuming with an increased amount of measuring that must be carried out.

This method requires the observation of the power trace of the system, related to the switching of the transistors in the CMOS circuit design. These measurements are then directly interpreted to reveal operation information and other critical data.

The DPA is a more complex and effective method of revealing secret parameters that requires advanced measuring tools and algorithms of a higher degree of complexity involving statistical analysis. This attack has the advantage of being able to extract useful information from crypto processors among electrically noisy environments due to its signal processing and error correcting properties.

2.3 Countermeasures: Hardware Versus Software

Cryptography is the study of how to better disguise sensitive information, or otherwise keep it from being observed by unwanted parties. Generally, it is simpler to implement algorithmic defenses, though these face the affliction of progressively innovative algorithmic attacks. Contrarily, hardware implementations are able to provide a more robust protection after a more challenging design process.

There are countless encryption algorithms which are used to secure data, a few of which have been described in section 2.2, however, these algorithms do not protect against Side Channel Attacks. These attacks, staged against the hardware of a system, require special design consideration during the implementation of the system, rather than a stronger algorithmic scheme. There are primarily two approaches to thwart such Side Channel Attacks: Masking method and the Hiding Technique [27].

In brief, Masking [8] is a method in which the side channel information is disguised by applying a randomized mask, or intermediate data. Hiding is performed by keeping

a constant power consumption, and therefore eliminating, or greatly reducing the side channel information to be observed.

2.4 Summary

This chapter discussed the types of encryption and presented examples of encryption algorithms. Specific security weaknesses were listed and defined, though the Side Channel Attack is of primary concern here.

This research focuses on the security of data beyond the algorithmic protection measures. The implementation of a secure system is achieved by purposefully designing the crypto processors circuitry to impede the external attacks known as Side Channel Attacks. The Public-Key Encryption algorithm of RSA is assumed and the hardware design of secure adder is implemented for its application.

The primary goal of this research was to defend against Differential Power Analysis, though in removing most of the side channel information, nearly all types of side channel attacks may be thwarted.

Chapter 3

Countermeasures: Circuit Architecture

This chapter will begin by defining the different approaches to securing data through circuit implementation, meaning, different defences against Side Channel Attacks. Then, the state of the art in circuit architectures that are currently being researched as countermeasures to Side Channel Attacks, as well as their advantages and disadvantages are presented.

3.1 Masking vs. Hiding

When referring to Power Analysis Attacks, there are two categories of countermeasures to be considered, Masking and Hiding [27]. These two techniques used to obstruct such Side Channel Attacks, vary greatly in concept and approach.

The Masking method [8] functions by applying a randomized mask to the intermediate data, anticipating that it may be measured by an outsider, which renders the observed power consumption values themselves to be irrelevant and inadequate toward obtaining the underlying secret message or key. The major disadvantage of this technique is that it is effective against Simple Power Analysis; however Differential Power Analysis Attacks or Timing Attacks may easily overcome this scheme.

The Hiding technique [23] concentrates on removing the dependency of glitches in the power consumption on the intermediate data. In other words, the aim of the Hiding scheme is to avoid creating any side channel information at all. Hiding can be accomplished in the time domain by randomizing the time of occurrence of a specific operation, or in the amplitude domain by minimizing the effect of the operations on the overall power consumption. By maintaining a constant power consumption, any quantity measured from the system is independent of secret values or intermediate information. The biggest disadvantage of this scheme is that there is a greater overall power consumption. The Hiding technique however, is a more secure technique that is capable of defending against attacks of a higher degree of complexity such as the Differential Power Analysis. Figure 3.1 illustrates the difference between the two countermeasure techniques of Masking and Hiding.

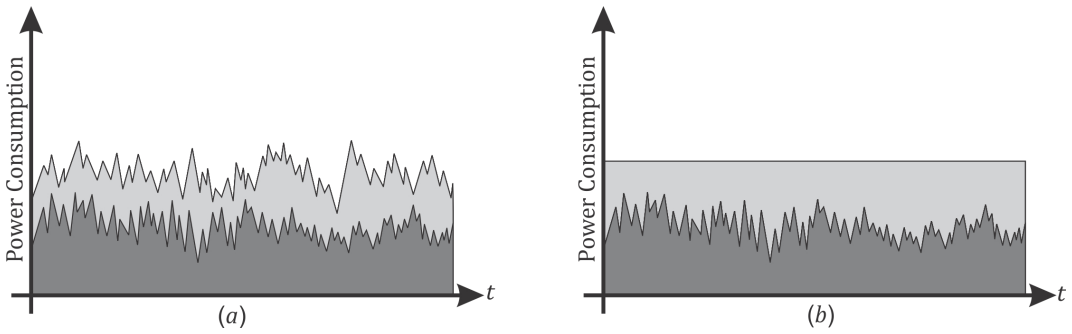

Figure 3.1: Countermeasure Techniques: (a)Masking (b)Hiding

3.2 Circuit Architectures

Since Side Channel Attacks began as a focus of researchers, there have been several proposed circuit design approaches to impede such attacks. These methods employ either the masking technique, or the hiding technique, previously described. There are many examples that may be presented of each of these two countermeasure classes. Two widely recognized security schemes, from which many others have derived, are the Random Switching Logic (RSL) [32], a Masking technique, and Wave Dynamic Differential Logic (WDDL) [33], a Hiding technique. These will be explained in further detail in this chapter.

Many of the existing techniques employ a dynamic differential logic [23].The goal of this technique is to reduce the possibility of information security breaches, carried out through statistical analysis of the power consumption, and careful observation and measurements, recognizing the dependence of this data on the inputs, and leading to an unsecured system. The dynamic differential logic model employs the use of two rails, carrying complimentary signals, in attempt to hide the otherwise useful data

sought after during the Side Channel Attack.

3.2.1 Random Switching Logic

Random Switching logic (RSL) [32] is an example of a countermeasure which uses the Masking technique. Here, a random mask is applied to disguise transition probabilities of inputs and outputs. RSL is a single-rail logic which operates by employing a 1 bit random value to all input and output signals, this is the masking process. The process is guided by an enable signal for synchronicity, and to assure security, the transition of the random signal is not biased.

A disadvantage of the RSL is that the effectiveness of this technique is highly dependant on the quality of the random number generator; this being a costly component. This, along with the weight of the power consumption overhead render this design method expensive in terms of implementation.

3.2.2 Sense Amplifier Base Logic

The Sense Amplifier Base Logic (SABL), proposed by Tiri et al. in [9], employs a dual-rail with pre-charge technique. In dual-rail designs, a capacitance is constantly being charged, regardless of the input, and every input is associated with a specific switch position. In other words, for every input there is a complementary signal generated and transmitted on a secondary wire. It means to impede Differential Power Analysis attacks by maintaining a constant value at the load capacitance.

SABL employs the use of a clock to perform the pre-charging which has the disadvantage of adding a large clock load. Though this method is efficient in maintaining a

constant power consumption regardless of the data, designing a circuit in this method requires the implementation of a custom library. Standard CMOS libraries do not include SABL gates, rendering this method unsuitable for current logic design. For this reason, Tiri et al. proposed the Wave Dynamic Differential Logic as a solution.

3.2.3 Wave Dynamic Differential Logic

The Wave Dynamic Differential Logic (WDDL) [33] is a design scheme based on the SABL, however, uses the standard cell library. In this design, CMOS gates as well as a pre-charge phase are used in order to compensate for circuit activity. Here, instead of a clock cycle being used for the pre-charge, it is done through a pre-charge wave which travels through the circuit.

A disadvantage of WDDL is that it is known to be prone to early evaluation and pre-charge [31] [13]. The primary cause for such effects is due to the mismatch in delay of variables belonging to the same gate.

3.2.4 Masked Dual-Rail Pre-charged Logic

An improvement on WDDL, Masked Dual Rail with Pre-Charged Logic (MDPL) [20] [1] is a design architecture that aims to solve the issue of unbalanced signal propagation. This logic style was conceived by combining the WDDL and RSL methods, resulting in a fusion of the dual-rail pre-charge logic and the masking technique. In this logic style, the true and false routes are interchanged randomly. This results in improvements to the routing balance, and that of the dual-paired gates. The MDPL method is achieved in two phases, synchronized by a clock signal. When the clock

signal is high, initialization of the circuit's differential pair are executed by way of a traveling wave, setting both signals and all flip-flops to (0,0). When the clock switches low, the circuit enters its evaluation phase, changing the differential signals to (0,1) or (1,0) based on the masking applied and the data imputed.

The disadvantage to this scheme is that it, as with the WDDL, remains prone to early evaluation and pre-charge.

3.2.5 Dual-Rail Random Switching Logic

Due to the fact that a successful scheme that would solve the early evaluation and pre-charged problem proved very difficult to be designed, dual rail random switching logic (DRSL) [4] was created with the goal to render these two variables independent of the data. In this scheme, there is a validity check of all the inputs before allowing them to propagate. DRSL has a pre-charge phase and an evaluation phase. The generation of the pre-charge signal has the effect of synchronizing the input signals. Synchronization of the complimentary signals is imperative to maintain the independence of the intermediate data to the input signals. The pre-charge phase forces all signals to 0; the pre-charge signal becomes invalid after the inputs are evaluated in the evaluation phase. Random mask changes occur every clock cycle, which are applied to the values in the registers for the following clock. In this scheme, glitches are suppressed with the pre-charge logic in conjunction with a random mask.

Though this design technique overcomes the problems with early evaluation, the main drawback to this style of design is its high complexity.

3.2.6 Multiple-Valued Source-Coupled Logic

The Multiple-Valued Source-Coupled Logic (MV-SCL) [3] developed by Y. Baba and his colleagues, employs the hiding scheme along with Multiple-Valued Source Coupled Logic in designing an adder, to maintain a constant power consumption profile, which would be independent of input values. This circuit design method begins by converting the adder's two digital inputs to their analog signal equivalents, followed by a simple nodal summation. After this summation, these signals undergo a current to voltage conversion before entering comparators and subsequently carry generator unit. The main attribute of this method is in the generation of differential pairs in the comparator process, balancing the signals, thus employing the hiding scheme.

Although this design method allows the power consumption to remain constant, therefore minimizing the availability of side channel information, it does require a significantly higher power consumption than its conventional digital counterpart. The basic idea presented by Baba comprises of an analogous concept to the novel design elaborated in this research thesis, however, the novel design approach offers significant improvements to the issue of the high power consumption while maintaining the security of the system.

3.3 Summary

Presented above are several existing techniques that were developed to protect against Side Channel Attacks. As discussed in this chapter, these existing countermeasures possess various weaknesses such as the requirement of a custom library, delayed signal propagation, high implementation costs, early evaluation issues, high design complex-

ity as well as very high power consumption.

In the research presented herein, a novel design for crypto processor implementation was designed to overcome these weaknesses while providing an increased security against most known forms of Side Channel Attacks.

Chapter 4

Proposed Circuit Architecture

This chapter explains the theory applied in the hardware design of a secure adder. Montgomery Multiplication is presented and explained with the use of examples. The design approach of the Current-Mode mixed signal adder is described, followed by a discussion of the proposed overall circuit architecture.

In the proposed design, a mixed-signal approach was utilized in defending against Side Channel Attacks. This novel design incorporates the advantages of using Current-Mode Logic along with Domino Logic for a comparative power consumption and less glitch ridden circuit than would be achieved through a fully digital circuit implementation.

4.1 Montgomery Multiplication

Montgomery Multiplication is a widely used modular multiplication algorithm created by, and named after, the mathematician Peter Montgomery. It was first published in 1985 [22], and since then has been used to greatly reduce the required resources and time to evaluate the operation of modular exponentiation.

Modular exponentiation is defined as repeated modular multiplications. In such applications as cryptography, where the variables consist of often over a thousand bits, this mathematical process becomes a major bottleneck in the crypto processor system.

The Montgomery algorithm allows efficient computation of modular arithmetic when the word size of the operands are large. More specifically, this algorithm computes the product of two integers modulo a third, without requiring a division by n (the modulus). It yields the reduced product using a series of additions. Since it is characteristic of Montgomery Multiplication to perform modular multiplication by substituting addition and multiplication for the computationally expensive division, there is a necessity for fast and efficient adders.

To compare the classic method of evaluating a modular multiplication versus the Montgomery method, a simple example in radix-10 is demonstrated [21]. First, in Table 4.1, the classical evaluation method is employed to find the result of M=43×56(mod 97), then illustrated in the flow diagram in Figure 4.1, the Montgomery process is divided into simple steps.

Following the Montgomery method of evaluating the modular multiplication in the given example, as seen in Figure 4.1, it is plain to recognize that this method

Table 4.1: Modular Multiplication Using Classical Evaluation Method

Step	**Find: 43×56 (mod 97)**
1	43×56 = 2408 (mod 97)
2	2408-97 = 2311 (mod 97)
3	2311-97 = 2214 (mod 97)
4	2214-97 = 2117 (mod 97)
⋮	⋮
20	662-97 = 565 (mod 97)
⋮	⋮
24	274-97 = 177 (mod 97)

allows a drastically simplified means of computing the solution using fewer processing resources and results in quicker completion.

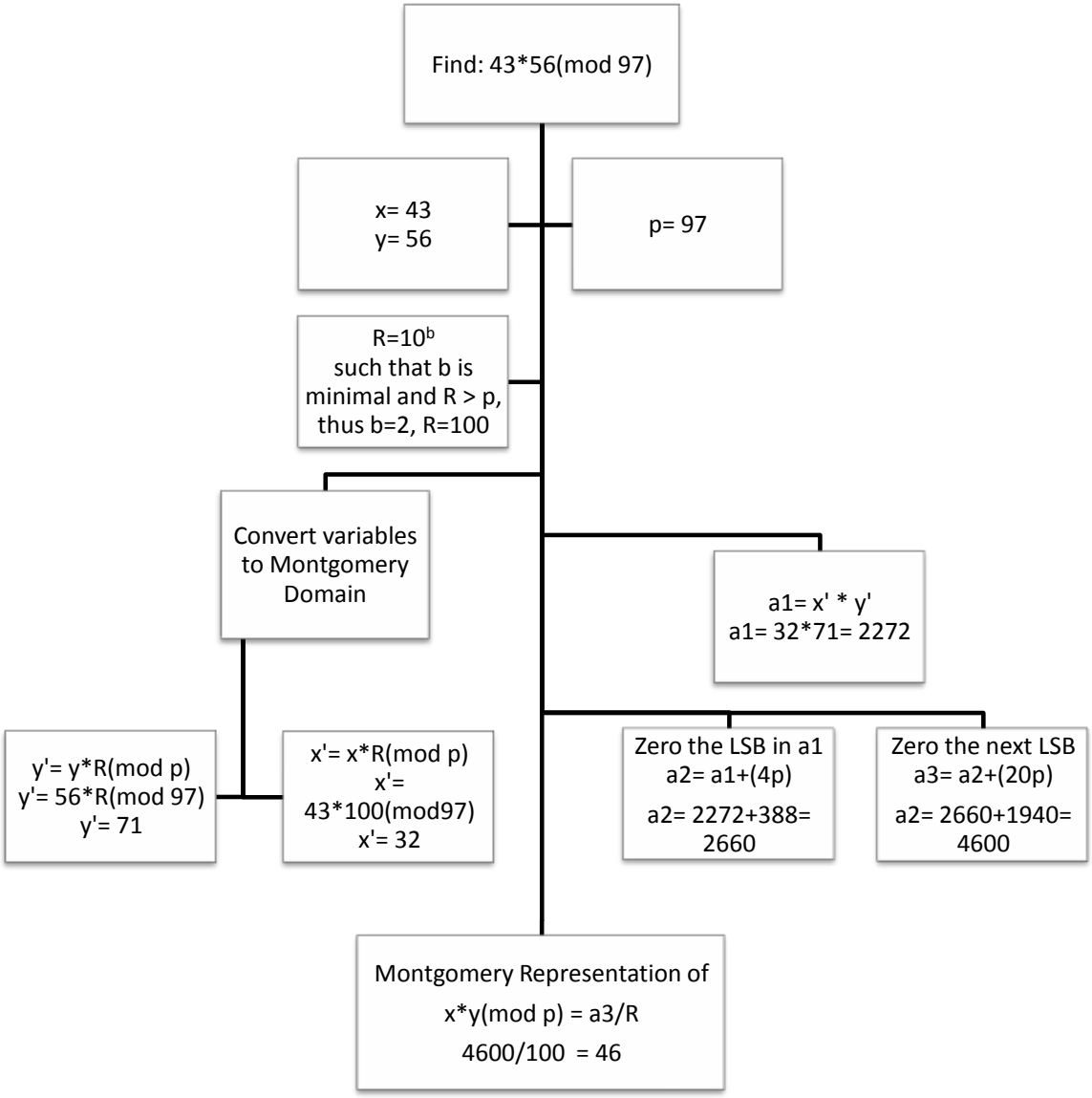

Figure 4.1: Modular Multiplication Using Montgomery Method

The Montgomery Multiplication process may be explained as follows: First the multiplicands are set to x and y respectively, and the modulus is set to p. The variable

R, is calculated such that it is the smallest power of the base in which the calculations are being performed, while being greater than p, the modulus. Then, x and y are both converted into the Montgomery domain, using R, according to the Equations 4.1 and 4.2.

$$x\prime = x \times R(modp) \tag{4.1}$$

$$y\prime = y \times R(modp) \tag{4.2}$$

After multiplying $x\prime$ by $y\prime$, multiples of the modulus, p, is added to their product in order to make the last two digits zero (this allows simple division by shifting the decimal twice to the left). The result is the Montgomery representation of the modular multiplication.

Note that the efficiency of this algorithm is achieved from the fact that in cryptography, an actual result of the modulus is not necessary. This means, the computations and results may all remain in the Montgomery Domain, where repetitive subtraction is replaced with multiple additions and simple division by a power of the radix (accomplished by using shift registers). Conversely, applying the Montgomery Multiplication to modular arithmetic where the actual final result is desired is costly; this, due to the necessary conversion to and from the Montgomery Domain.

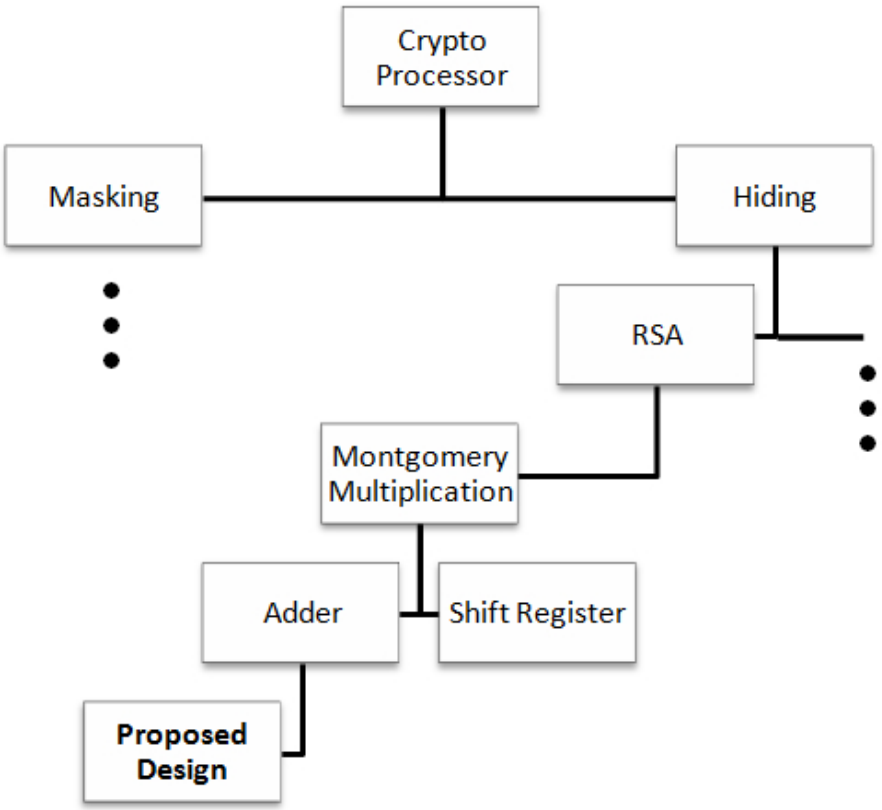

Figure 4.2: An Adder as Research Focus

Since the modular arithmetic in the RSA algorithm, the focus of this research, is conveniently replaced by simple addition, the main component of the crypto processor becomes an adder. This point is demonstrated by Figure 4.2. Therefore, the circuit design proposed in this research incorporates the layout of a mixed signal version of a Carry Look-Ahead Adder (CLA) logic.

4.2 Mixed Signal Carry Look-Ahead Adder

The design of this Multiple-Valued Carry Look-Ahead Adder (CLA) is the proposed solution to the recently popular security issues related to Side Channel Attacks. This novel design combines the benefits of Current Mode Logic and Domino Logic to result in a very low power and secured hardware architecture. Figure 4.3 illustrates the full block diagram of the proposed design.

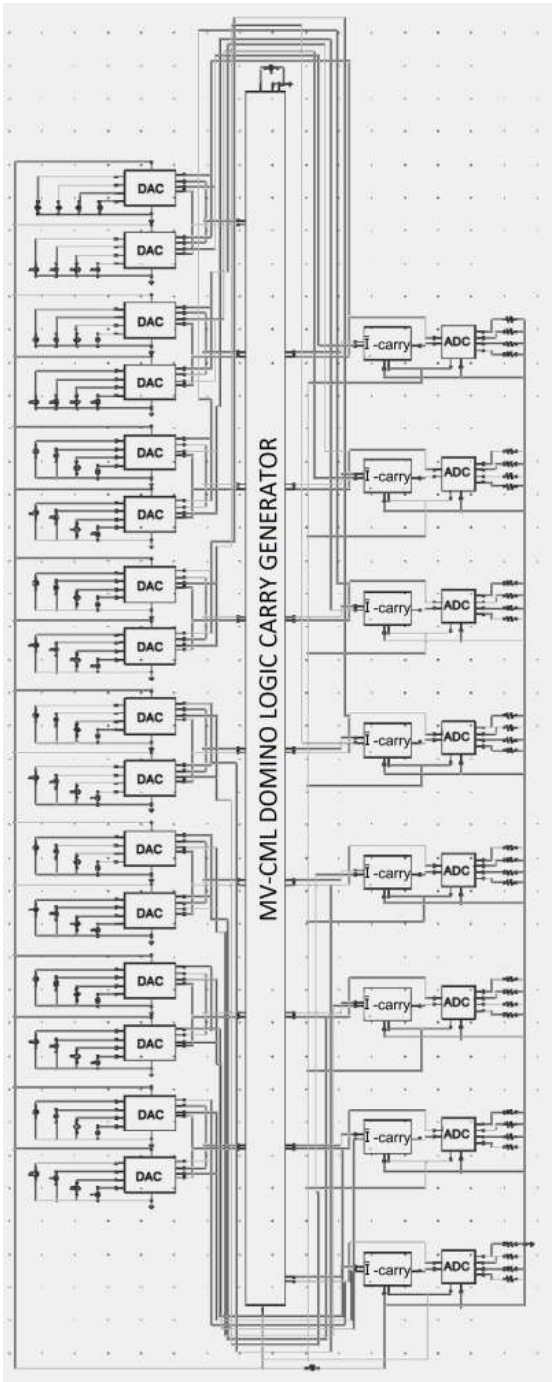

Figure 4.3: Block Diagram of the Proposed Full Circuit

Current-Mode Logic permits the circuit to accurately and effectively balance the values of the intermediate data, and allows for a constant power consumption independent of the input values. While the proposed method does require an increased area for the conversion of the input signal to analog and to generate its complement, these signals, in contrast to the previously describes architectures, are both used later in the circuit, removing unnecessary redundancy. Figure 4.4 illustrates the nature of the signal throughout its propagation in the circuit.

Figure 4.4: Signal Propagation of Proposed Design

4.3 Characteristics of the Proposed DAC

In order to hide the side channel information, a constant power consumption is desired. This is accomplished by employing a Dual-Rail Current-Mode Logic to more effectively and efficiently protect against Side Channel Attacks.

The digital inputs to the circuit are converted to their current-mode equivalent, while a complement of the signal is generated simultaneously, in the dual-rail system. A complement value in an analog system differs from that of a digital system. For example, in a mixed signal design, if we consider the radix to be B, then the complement value for the analog value, x, would be equal to $B - x$. Values of each

signal as well as their complement are generated using Analog to Digital Converters. As illustrated by Figure 4.5, the Digital to Analog Converter (DAC) is implemented by employing switches to simultaneously generate the analog values as well as their complement.

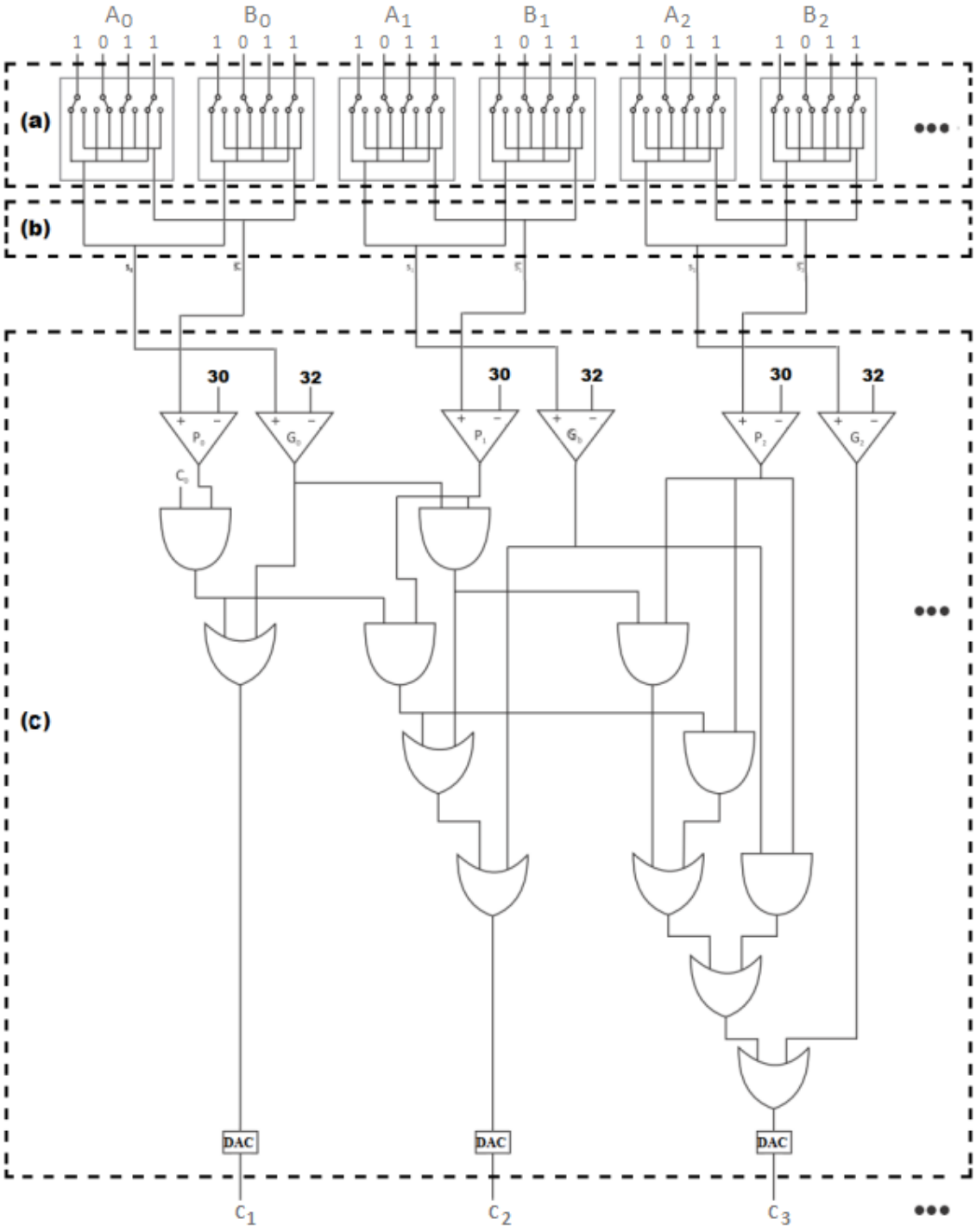

Figure 4.5: Gate Level Diagram of the DAC and Carry Generator blocks

There are, of course, two inputs (operands) to the adder circuit, namely A and B. As depicted in part (a) of Figure 4.5, the block representation of the DAC portion, the two 32-bit inputs, A and B, are divided into groups of 4-bits before being converted to their analog equivalent signal and its complement. Part (b) of Figure 4.5 demonstrates that the generated analog signal and its complement are summed by way of nodal addition with the corresponding analog equivalent and complement signals from the DAC of the second input, respectively. These new summed signals, forwarded to the Carry Generator block, are referred to as S_i for the summed analog equivalent signals of A and B, and $\overline{S_i}$ for the summed complement signals. Figure 4.6 shows that the average of these two signals, S_i and $\overline{S_i}$ is a constant current value. Following their conversion to the analog domain, these input signals are sent to the Carry Generation block. The Carry Generator is a mixed signal unit, and is composed of current comparators and domino logic. The gate level structure of the Carry Generator is described in the next section.

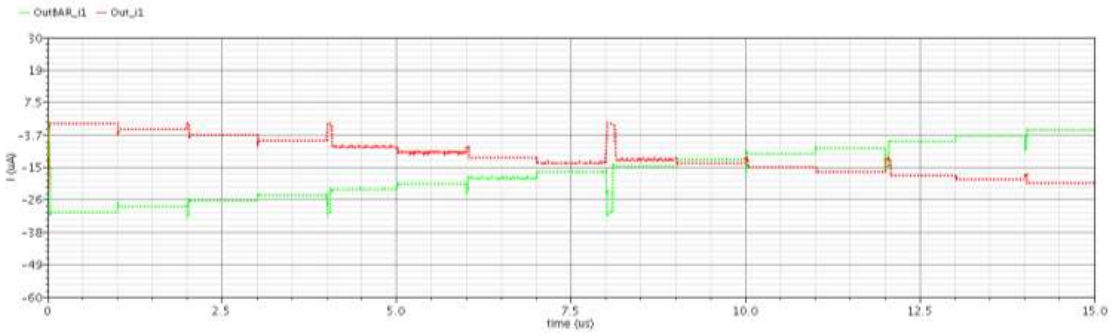

Figure 4.6: Analog Equivalent Signal and its Complement

4.4 Characteristics of the Proposed Carry Generator

An advantage of this design, is the facility with which it may be applied to the novel dual-rail Current-Mode Logic architecture for the cryptographic processor.

The typical model of the CLA [16] is redesigned to function as a mixed signal circuit. Figure 4.5 (c) shows that the inputs to the Carry Generator block of the CLA are analog signals, however, the internal circuitry of the Carry Generator is primarily implemented with the digital domino logic, while its outputs are again in current-mode logic. The method in which this mixed signal design functions is simplified and explained by the Equations 4.3 through 4.10.

The carries and sums are calculated based on Equations 4.3, 4.4, 4.5, and 4.6. Here, G_i and P_i are known as the carry generate and carry propagate signals, respectively. The constant A_{max} represents the maximum value representatble, and is determined based on the resolution of the DAC sub-blocks. For example, since the two 32-bit digital inputs are divided into groups of 4-bits, prior to their conversion to Current-Mode Logic, A_{max} has an analog value corresponding to the maximal 4-bit digital input of 1111. C_i denotes the carry and S_i is the sum of the current-mode input signals generated by the digital to analog converter, and S_{out} is the final result of the addition.

$$P_i = 1 \; when \; \overline{S_i} \leq A_{max} \tag{4.3}$$

$$G_i = 1 \; when \; S_i > A_{max} \tag{4.4}$$

$$C_{i+1} = G_i + P_i C_i \; (digital \; logic) \tag{4.5}$$

$$S_{out} = S_i + C_i \; (digital \; logic) \tag{4.6}$$

All of the carry signals, S_i and $\overline{S_i}$, enter the Carry Generator circuit from the DAC block simultaneously. These signals are then compared to the the value of A_{max} as shown in Equations 4.3 and 4.4, to determine the values of P_i and G_i, respectively. The last step of this block is to calculate the carries. The general method for the carry signal calculation is given in Equation 4.5. Equation 4.6 shows the final sum signal generation.

The logic used to generate the P_i and G_i signals is displayed in Table 4.2. The encircled rows represent the critical cases in which a change occurs in the output for the Carry Propagate and Carry Generate signals.

Table 4.2: Logic for Signal Generation

Inputs to CLA			**Generated Signals**	
Decimal Value	**MV-CML Representation**		**Carry Propagate**	**Carry Generate**
	S_i	$\overline{S_i}$		
0	0	60	0	0
1	2	58	0	0
2	4	56	0	0
3	6	54	0	0
⋮	⋮	⋮	⋮	⋮
12	24	36	0	0
13	26	34	0	0
14	28	32	0	0
15	**30**(A_{max})	**32**	**1**	**0**
16	**32**	**30**	**1**	**1**
17	34	26	1	1
18	36	24	1	1
19	38	22	1	1
⋮	⋮	⋮	⋮	⋮
28	56	4	1	1
29	58	2	1	1
30	60	0	1	1

As demonstrated in Equations 4.7 through 4.10 (all digital logic), the results of each consecutive carry relies only on the first carry input, C0 and the Carry generate and Carry Propagate signals from each stage, which are, as previously discussed, generated concurrently. The generation of these signals follow the traditional digital CLA algorithm. The first four carry signals are presented here as follows:

$$C_1 = G_0 + P_0C_0 \tag{4.7}$$

$$C_2 = G_1 + P_1G_0 + P_1P_0C_0 \tag{4.8}$$

$$C_3 = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_0 \tag{4.9}$$

$$C_4 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0C_0 \tag{4.10}$$

Domino logic was used in the implementation of the Carry Generator logic block for greater circuit optimization. This type of logic includes a pre-charge and an evaluation phase. Domino logic allows a smaller area as well as smaller parasitic capacitances permitting a faster circuit speed, and more importantly it reduces glitches and the circuit power consumption. Details on the circuit topologies are presented in Chapter 5.

4.5 Characteristics of the Proposed ADC

Once the carry signal is generated in the Carry Generator block, the sum signals, along with the carry information, must be then converted back to the digital domain by way of Analog-to-Digital Converter (ADC). Again in this block, it is essential to minimize data dependent spurs that would compromise the security of the hardware, and to maintain the feasibility of the proposed secure adder, it was an aim to minimize the power consumption. In order to create an efficient and optimized analog to digital converter, signals already generated and present in the circuit are scaled and used along with comparators and digital logic, according to Figure 4.7, to produce the final

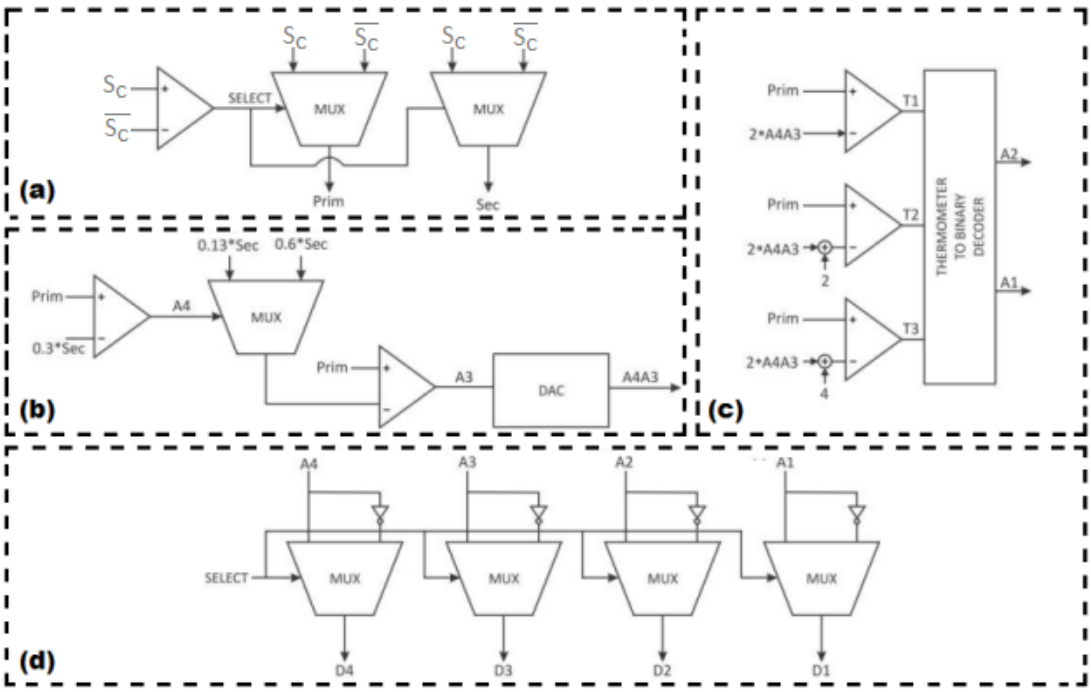

Figure 4.7: Block Diagram of Proposed ADC Design

digital result of the desired addition operation.

As can be observed in Figure 4.7 (a), the first step in the ADC block is to determine which of the $sum + carry$ (S_c) and its complementary signal $\overline{sum} - carry$ ($\overline{S}_c$) is the Primary ($Prim$), and which is the Secondary (Sec) signal, as well as setting the value of the $SELECT$ signal within this block. The value of $SELECT$ is set to 1 if S_c is greater than $\overline{S}_c$. Once this is determined, the $Prim$ and Sec signals proceed into comparator units, scaled as depicted in Figure 4.7 (b). The blocks depicted in Figure 4.7 (b) and (c) determine the digital values of the 4-bit signal, $A4A3A2A1$, $A4$ being the most significant digit, $A1$ the least significant.

After determining $A4$ and $A3$ in part (b) of Figure 4.7, they are again converted back to the analog domain ($A4A3$) and forwarded to part (c) of Figure 4.7 derive the two lowest significant binary digits, $A2$ and $A1$.

The last two bits are distinguished based on the logic exhibited in Table 4.3.

Table 4.3: Logic for the Generation of the Two Least Significant Bits

2$A4A3$	**2**$A4A3$**+2**	**2**$A4A3$**+4**	**Primary**	**T3**	**T2**	**T1**	**A2**	**A1**
0	2	4	0	0	0	0	0	0
0	2	4	2	0	0	1	0	1
0	2	4	4	0	1	1	1	0
0	2	4	6	1	1	1	1	1
8	6	8	8	0	0	0	0	0
8	6	8	10	0	0	1	0	1
8	6	8	12	0	1	1	1	0
8	6	8	14	1	1	1	1	1
16	18	20	16	0	0	0	0	0
16	18	20	18	0	0	1	0	1
16	18	20	20	0	1	1	1	0
16	18	20	22	1	1	1	1	1
24	26	28	24	0	0	0	0	0
24	26	28	26	0	0	1	0	1
24	26	28	28	0	1	1	1	0
24	26	28	30	1	1	1	1	1

Finally, as shown in part (d) of Figure 4.7, depending on the value of the $SELECT$ signal, which was previously determined, the final digital output of the adder is obtained. The two conditions represented in Equations 4.11 and 4.12, define the method in which the output is decided. The signal defined as $D4D3D2D1$ represents

the final output of the circuit. If the $SELECT$ signal has a value of 0, the binary values of $A4$ through $A1$ are set directly as the output. If the value of the $SELECT$ signal is 1, the inverse of the digital signal $A4A3A2A1$ is set as the final output of the circuit.

$$if S_c < \overline{S_c}, then\ SELECT = 0,\ \ D4 = A4, D3 = A3, D2 = A2, D1 = A1 \quad (4.11)$$

$$if S_c > \overline{S_c}, then\ SELECT = 1,\ \ D4 = \overline{A4}, D3 = \overline{A3}, D2 = \overline{A2}, D1 = \overline{A1} \quad (4.12)$$

Chapter 5

Results

This chapter presents the transistor level diagrams of the different components of the overall design. Also explored in this chapter are the results obtained from simulating the side channel resistent adder proposed in this research. The conclusions that may be derived from the results are described, and then compared to those of the State of the Art.

The 90nm technology, with a power supply of 1.2V, was used along with the Cadence software to design and simulate the proposed architecture. This proposed adder architecture is designed to be able to process blocks of data with a word length of 32 bits. This work opens a path for extended resolution adder, which would be able to function with a much higher input word length. Arbitrary digital values were fed as inputs to this novel design to test the functioning and obtain the results of the

circuit.

5.1 Digital-to-Analog Converter

The main goal of this research is to provide a solution for a robust building block of crypto processors that is able to withstand the covert threats known as Side Channel Attacks. Side channel information such as power consumption spikes, as is the focus here, may be observed and analyzed to then decipher an encoded message. As was previously mentioned, a technique referred to as hiding was employed to remove, or greatly minimize, any power consumption dependence on data, whether input, output, or intermediate, from the encryption operations it undergoes.

This first block is perhaps the most crucial to the overall design goal of the circuit. It is here where the digital inputs are fed and converted to their analog equivalent and its complementary signal concurrently.

As described in the previous chapter, there are two inputs to the adder circuit. Each input is composed of 32-bits. In the first phase, the two 32-bit inputs are divided into groups of 4-bits, prior to entering the DAC (Digital to Analog Converter) block and being converted into Multiple-Valued Current Mode Logic. Since each DAC block accepts a 4-bit digital input value, the possible binary inputs range from 0000 to 1111, in decimal this equates to a range of 0 to 15.

Figure 5.1 is the transistor level diagram of the digital logic switch connected to each binary input. This switch is the initial step in implementation of the dual-rail system. The value of the digital input to each switch directs the current to one of the specified dual-rail lines, in turn simultaneously generating the current-mode logic

equivalent to the input and its complement.

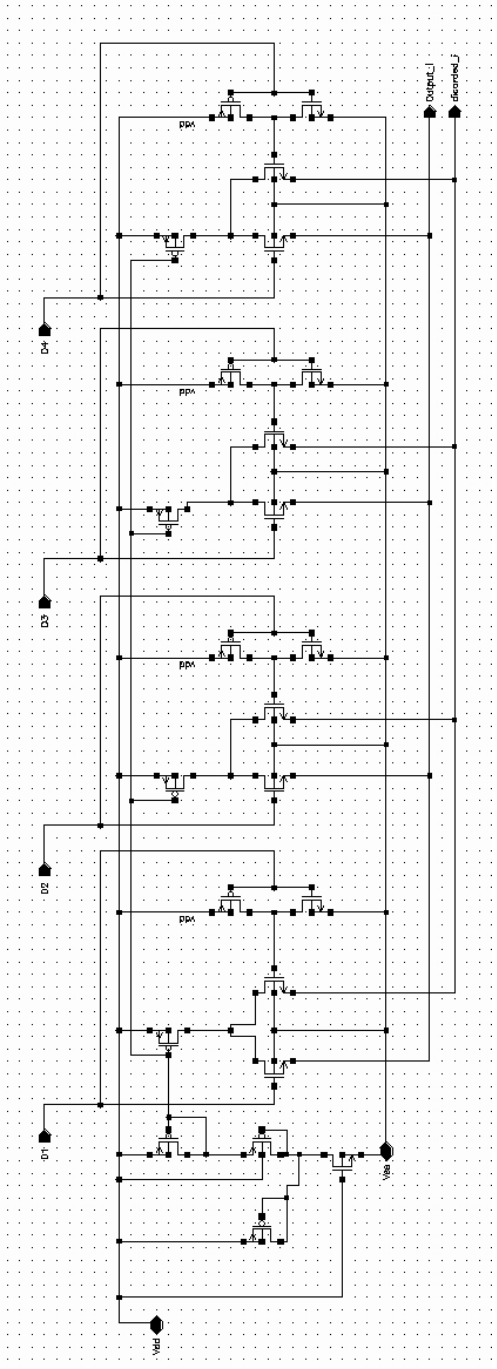

Figure 5.1: Transistor Level Diagram of One a One Bit Conversion

Each increment of 1 in decimal corresponds to the analog equivalent value of 2μA (ex: $1_{10} = 2\mu$A, $2_{10} = 4\mu$A, $3_{10} = 6\mu$A, etc.). This means, with a maximum binary value of 1111, or 15_{10}, the maximum current value resulting any line from the DAC stage is 30μA. Since the output of the DACs are summed, by way of nodal current addition, with the output of the DAC of the correspondingly weighted inputs of the second operand, the maximum summed current mode-logic equivalent (without considering the carry at this stage) is 60μA. It follows that the complementary signal, generated in each DAC, is equal to the maximum Current-Mode Logic (CML) output minus the input's equivalent CML value. This logic is described in Equation 5.1 to find the complementary signal, $\overline{output_i}$.

$$\overline{output_i} = 30\mu A - output_i \tag{5.1}$$

Furthermore, Equation 5.2 represents the value of the complementary signal, $\overline{S_i}$, after nodal summation with the corresponding DAC output. The signals signals S_i and $\overline{S_i}$ become the inputs to to Carry Generator which is explained in the proceeding section.

$$\overline{S_i} = 60\mu A - S_i \tag{5.2}$$

To find the worst case power consumption of the digital to analog conversion process, one of the 32-bit operands, A, was set to a constant value, and the other input, B, was set to an incrementing binary value. These two operands can be seen in Figure 5.2 and 5.3. The power consumption necessary to convert all 64-bits of the two digital inputs to the analog domain, is demonstrated in Figure 5.4. From

this figure, it is evident that the power consumption is successfully maintained at a constant level.

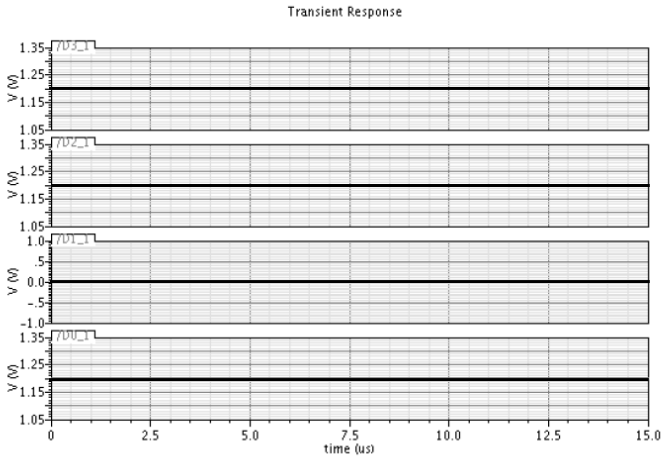

Figure 5.2: 4-bit Input A: Constant Value

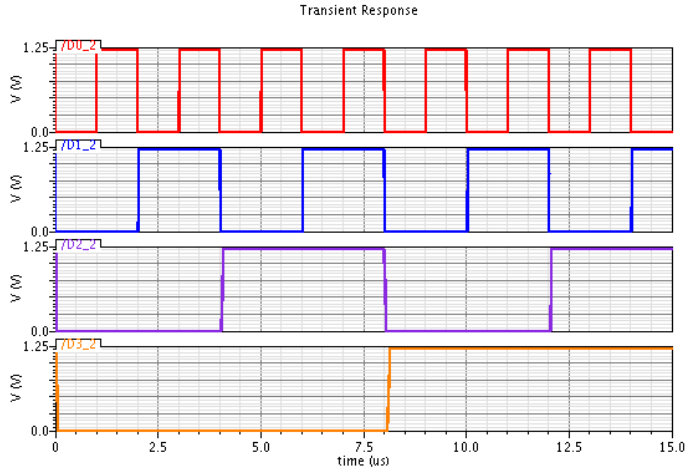

Figure 5.3: 4-bit Input B: Incrementing Value

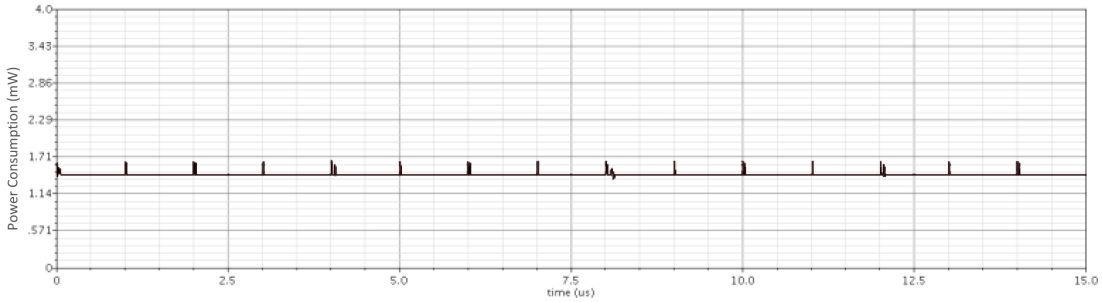

Figure 5.4: Power Consumption of two 32bit Digital to Analog Conversions (64 bits)

An inherent disadvantage of any dual-rail circuit design, is a higher overall power consumption. To oppose this negative property, the complementary signals, as well as the equivalent signals, generated in this block of the design, are used further in the circuit as a convenient and efficient comparator input for carry signal calculations. The complementary signals are used in place of current reference, which are always required for DAC and ADC. These signals are used for dynamic current comparators. By avoiding the use of static current references, power consumption of the proposed adder was reduced significantly. This approach lends to a more practical circuit architecture, with a much improved overall power consumption.

5.2 Mixed Signal Carry Generator

Classically, the Carry Look-Ahead adder has an entirely digital composition. To adapt this circuit model for the desired outcome of the design, the digital logic gates were replaced by both analog signal comparators, as well as domino logic gates.

Domino logic gates are a type of dynamic digital logic gate which functions with a pre-charge and an evaluation phase controlled by a clock signal. This logic requires fewer transistors to implement the same logic gates as in the typical static logic implementation. See Figure 5.5 for a comparison of a 2-bit AND gate in static CMOS Logic and Domino Logic. With the decrease in the number of transistors, there is a decrease in area, as well as parasitic capacitances. The main advantage of employing this logic is that it leads to a reduction in overall power consumption, as well as reduced glitches in the power profile.

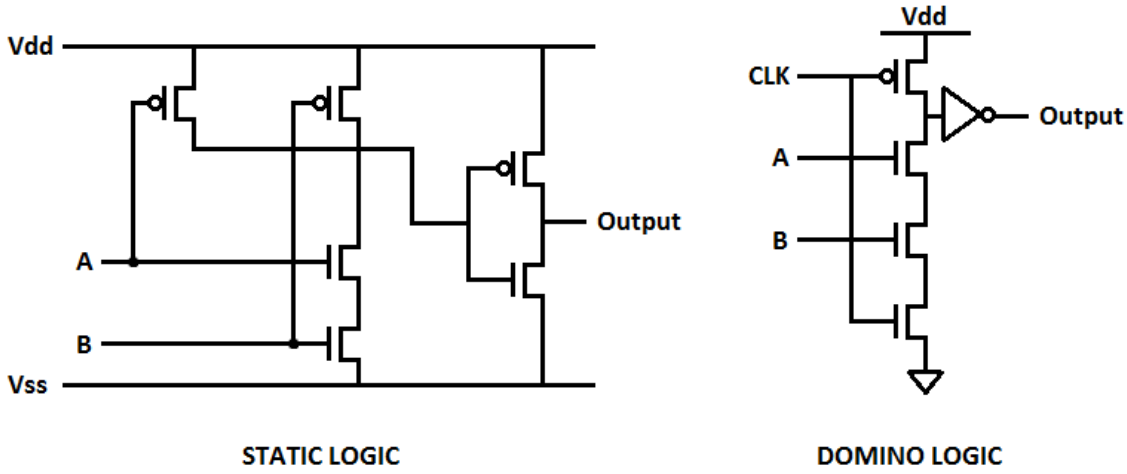

Figure 5.5: Static versus Dynamic Logic

The comparators, as seen in Figure 5.6, generate the required Carry Generate (G_i) and Carry Propagate (P_i) signals by way of comparing the current-mode sums, generated in the previous stage of the design, to the necessary constants of 30μA and 32μA.

In a 4-bit digital number, the maximum value that can be represented is 1111

(15_{10}), which in our multiple-value current mode environment is equivalent to 30μA. This observation leads to the conclusion that, if the sum is greater than 30μA, there will be a carry, hence the G_i signal is set to 1. Additionally, in the case where the sum is equal to 30μA, the P_i signal is set to 1, indicating the possibility of the generation of a carry. To synchronously determine the G_i and P_i signals, they are each compared to constant values based on the following logic (see Table 5.1): If the input sum (S_i) $\geq$ 32, then G_i=1, and if the sum complement ($\overline{S}_i$) $\leq$ 30, then P_i=1.

Table 5.1: G_i and P_i Generation Logic

$\mathbf{S}_i$	$\overline{S}_i$	G_i	P_i
28	32	0	0
30	30	0	1
32	28	1	1
34	26	1	1

Once all of the G_i and P_i signals are established, Domino Logic is then employed to, at once, determine all carry values based on the logic presented in Chapter 4. By maintaining the mixed signal approach, and after modifying the architecture for accurate arithmetic operation, a constant and low power consumption was able to be conserved. This outcome may be confirmed by inspection of Figure 5.7.

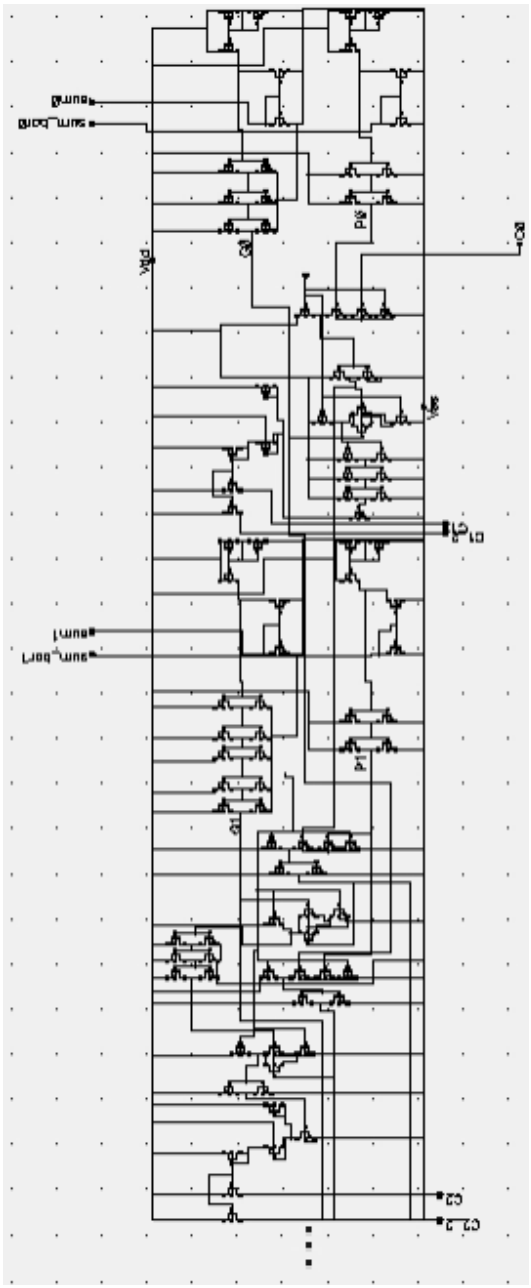

Figure 5.6: Transistor Level Diagram of the MV-CMDL Carry Generator

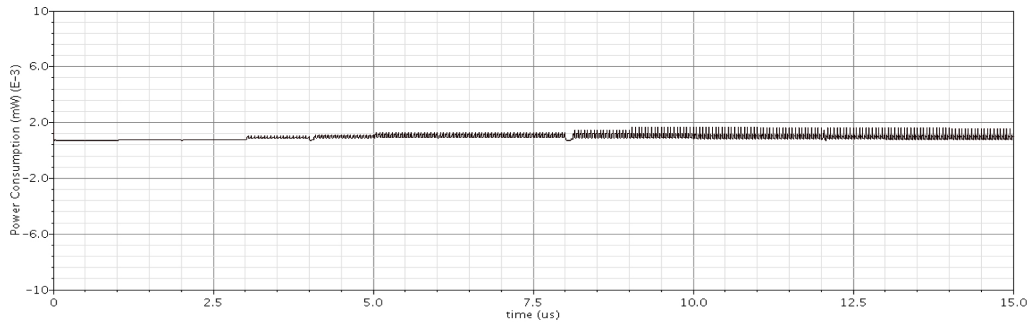

Figure 5.7: Power Consumption of the Carry Generator block

5.3 Analog-to-Digital Converter

Prior to entering this Analog-to-Digital Conversion (ADC) block, the carry information, determined from the previous block, is combined to both the sum (S_c) and the complement signal ($\overline{S}_c$). This now means that since the carry may assume the value of 0μA or 2μA, the new maximum values for S_c and $\overline{S}_c$ is 62 μA. These two new signals are then compared to each other in order to determine the value of the *Select* signal. The $SELECT$ signal is useful in determining which of the signals, S_c or $\overline{S}_c$, is the primary, and which is the secondary signal. Additionally, the $SELECT$ signal identifies whether the digital value at the last stage is inverted or not before being sent as the output. This logic can be seen in Table 5.2.

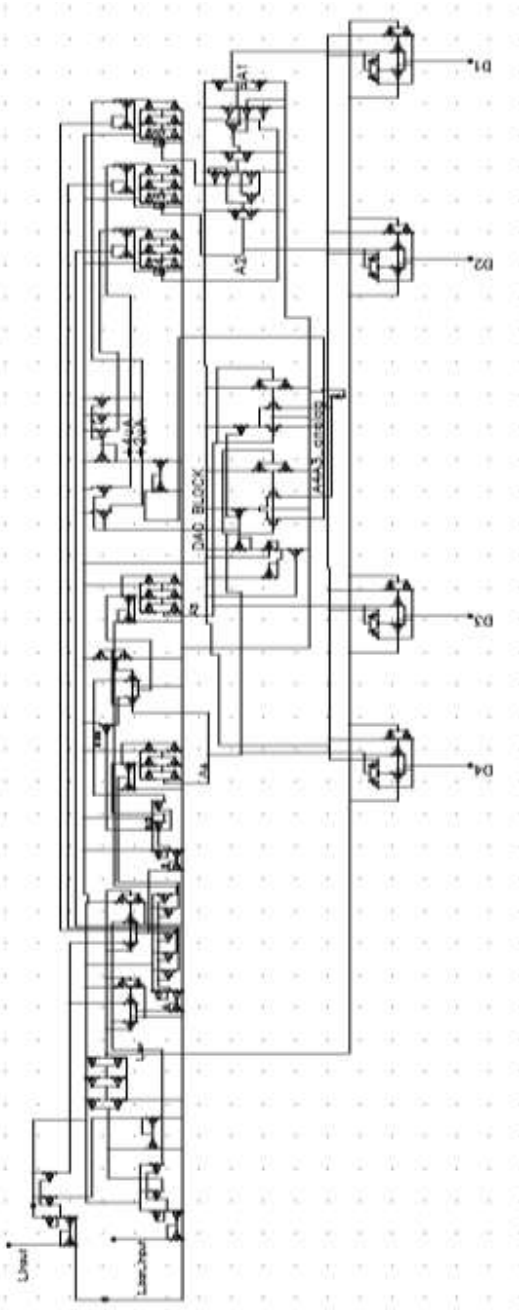

Figure 5.8: Transistor Level Diagram of a 4-bit ADC

Table 5.2: Function of the $SELECT$ signal

Condition	$SELECT$ **Signal**	**Primary**	**Secondary**	**State of Outputs**
$S_c < \overline{S}_c$	0	S_c	$\overline{S}_c$	Non-Inverted
$S_c > \overline{S}_c$	1	$\overline{S}_c$	S_c	Inverted

Figure 5.8 illustrates the transistor diagram for this specially designed ADC. As previously described in Chapter 4, the secondary signal is scaled and then compared to the primary signal. The secondary signal is scaled to find the two most significant digital values, as seen in part (a) of the figure. These are then converted back to their analog equivalent by passing through a smaller DAC as was designed during the first stage of the circuit, as seen in part (b) of the figure. The analog signal generated here can assume four values. This 2-bit conversion represents the MV-CML equivalent of the two most significant bits, $A4A3$. The conversion is as demonstrated in Table 5.3.

Table 5.3: 2-bit DAC Conversion Equivalents

Input	**Digital Weight**	**Analog Equivalent**
00	0	0
01	4	8μA
10	8	16μA
11	12	24μA

A fundamental principle in this novel ADC design is in the re-use of the signals already generated in the circuit. After analyzing the relationship of the signals prop-

agating in the circuit, this ADC takes advantage of such correlation by next biasing the signal $A4A3$ and then comparing the scaled primary signal again to determine the last two digital values. As seen in Figure 5.8, the transistors M1 and M2, in part (c), bias the $A4A3$ by the addition of 2μA and 4μ respectively. Depending on the value of the signal $SELECT$, in part (d) of Figure 5.8. Making this efficient ADC allows a low power consumption, and by applying similar logic as in the DAC proposed, the power consumption is also maintained at a constant level, see Figure 5.9.

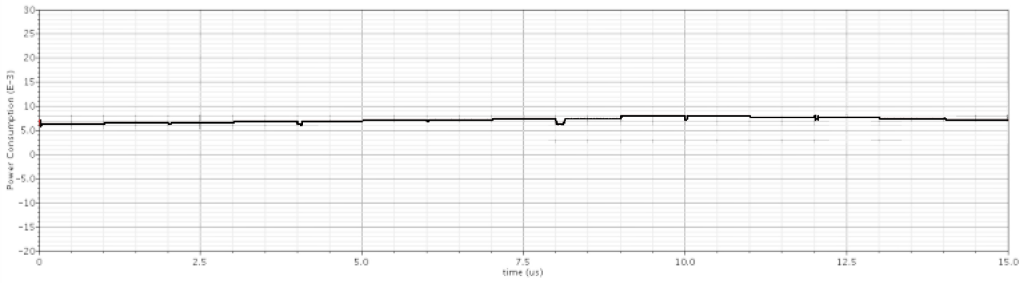

Figure 5.9: Power Consumption of two 32-bit Analog to Digital Conversions (64 bits)

5.4 Comprehensive Circuit Results

This section reports on the overall performance of the proposed adder circuit design. The average overall power of the complete circuit was found by introducing a series of random inputs to the circuit, and then through the Cadence software, measuring the average power. As can be concluded from Figure 5.10, the ultimate research goal of maintaining a constant power consumption profile, and completely dissolving any dependance of the glitches in the side channel information has been achieved.

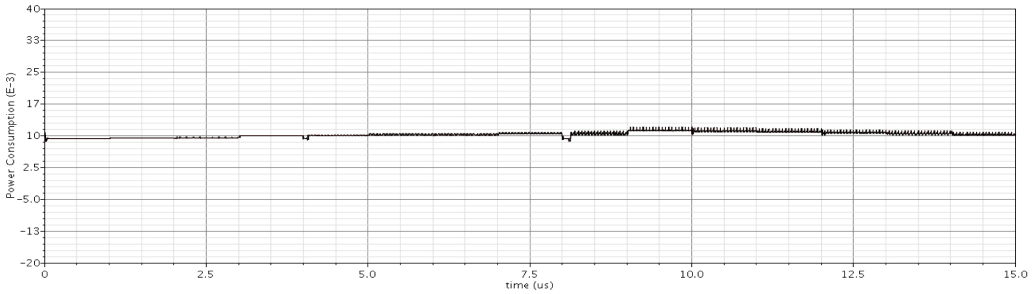

Figure 5.10: Overall Circuit Power Consumption

In the Table 5.4, a comparison was made between the novel design model and the leading state of the art results [3]. For a comprehensive comparison, data from the conventional adder, used in the RSA algorithm, was presented. It must be noted that, although the power consumption is much lower in the standard adder model, it is highly susceptible to Side Channel Attacks, even if stronger algorithmic securities are applied.

Table 5.4: Results Comparison

	Conventional [28]	**Baba et.Al. [3]**	**Proposed Design**
Technology	90nm	90nm	90nm
Resolution	32 bits	32 bits	32 bits
Power Consumption Range	0.0014-14.88mW	NA	NA
Avg. Power Consumption	0.58mW	20.76mW	9.3mW

Chapter 6

Conclusion and Recommendations

Technology is involved in nearly every aspect of modern life. More importantly, the transmission of private information is increasing as the market for embedded systems grows. Sensitive data is transmitted in financial transactions, Smartphone applications, even transportation instruments. As a consequence of the ubiquity of modern communication systems, it is imperative to consider hardware implementation techniques as the key to safeguarding sensitive information from the outside attackers.

Previously, algorithmic attacks were the predominant method breaches of data security measures. The research presented in this thesis focuses on the security of data beyond the algorithmic protection measures. The implementation of a secure system is achieved by purposefully designing the crypto processors circuitry to minimize the generation of side channel information, impeding the external attacks known

as Side Channel Attacks. The public-key encryption algorithm of RSA, which due to its continued popularity, is assumed and the hardware design of secure adder is implemented for its application.

The proposed solution to the Side Channel Attacks is the circuit design approach of this Multiple-Valued Current Mode and Domino Logic (MV-CMDL) Carry Look-Ahead Adder (CLA). This novel design proves to combine the benefits of Current Mode Logic and Domino Logic, resulting in a very low power and secured hardware architecture. The scheme used in disguising the side channel information created by the electronic devices is called the Hiding technique. In Hiding, the side channel information is obscured by keeping a constant power consumption, and therefore greatly reducing the side channel information to be observed.

The novelty of this research is in the combination of MV-CML and domino logic to this arithmetic system of an adder, and its application to a dual-rail system, producing a circuit which maintains a constant, and low, power consumption. In fact, major improvements have been accomplished in power consumption optimization compared to the similarly endeavoring circuits. Indeed, results show that this novel architecture consumes approximately 55% less power.

The main disadvantage of employing this design is the difficulty that is faced when attempting to increase the resolution of the system. For that reason, my recommendations for future work involve changing the logic of the carry signal generation from Carry Look-Ahead adder to a Carry Save Adder, where the true sum may not be calculated, but as the sum is solely needed for the purpose of multiplication, all that is needed is a partial sum. Once carry logic is changed, the resolution should be increased, for the advantage of crypto processor use. Another improvement that may

be made on the design is to minimize the switching spurs by adding sized buffers in the DAC block, as well as an enable/disable signal for further power saving measures.

References

[1] Elke De Mulder Benedikt Gierlichs Bart Preneel Ingrid Verbauwhede. Practical dpa attacks on mdpl. Cryptology ePrint Archive, Report 2009/231, 2009. http://eprint.iacr.org.

[2] Fred Cohen Associates. A short history of cryptography. *http://all.net/edu/curr/ip/Chap2-1.html*, 1990.

[3] Y. Baba, A. Miyamoto, N. Homma, and T. Aoki. Multiple-valued constant-power adder for cryptographic processors. In *Multiple-Valued Logic, 2009. ISMVL '09. 39th International Symposium on*, pages 239 –244, may 2009.

[4] Zhimin Chen and Yujie Zhou. Dual-rail random switching logic: a countermeasure to reduce side channel leakage. In *Proceedings of the 8th international conference on Cryptographic Hardware and Embedded Systems*, CHES'06, pages 242–254, Berlin, Heidelberg, 2006. Springer-Verlag.

[5] Wei Dai. Crypto++ 5.6.0 benchmarks, March 2009. http://www.cryptopp.com/benchmarks.html.

[6] IEEE Kris Tiri Member IEEE Alireza Hodjat Student Member IEEE Bo-Cheng Lai Student Member IEEE Shenglin Yang Student Member IEEE Patrick Schaumont Member IEEE David D. Hwang, Member and IEEE Ingrid Verbauwhede, Senior Member. Aes-based security coprocessor ic in 0.18-um cmos with resistance to differential power analysis side-channel attacks. *IEEE Journal of Solid-State Circuits*, 41, April 2006.

[7] Data Encryption Standard (DES). Federal information processing standards publication, fips pub 4, 1-26. 1999.

[8] Chari et al. Towards sound approaches to counteract power-analysis attacks. *CRYPTO*, page 398, 1999. 99, LNCS 1666.

[9] Tiri et al. Independent power consumption to withstand differential power analysis on smartcards. In *Ïproc. Of 28th European Solid-State Circuits Conference*, pages 403–406, pp, 2002.

[10] Gael Rouvroy Jean-Jacques Quisquater Franois-Xavier Standaert, Eric Peeters. An overview of power analysis attacks against field programmable gate arrays. *Proceedings of the IEEE*, 94, February 2006.

[11] K. Gandolfi, C. Mourtel, , and F. Olivier. Electromagnetic analysis: Concrete results. In *Proceedings of the 3rd International Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, vol, May 2001. 2162. Paris, France: Springer-Verlag, pp. 255 265.

[12] Travis N. Blalock Girish B. Ratanpal, Ronald D. Williams. An on-chip signal suppression countermeasure to power analysis attacks. *IEEE Transactions on Dependable and Secure Computing*, 1, July-September 2004.

[13] S. Guilley, L. Sauvage, J.-L. Danger, T. Graba, and Y. Mathieu. Evaluation of power-constant dual-rail logic as a protection of cryptographic applications in fpgas. In *Secure System Integration and Reliability Improvement, 2008. SSIRI '08. Second International Conference on*, pages 16 –23, july 2008.

[14] Dr. R.G.Ragel Isuru Herath. Side channel attacks: A reality. *Annual Technical Convference of the IET-YMS Sri Lanka*, 2007.

[15] Gary Kessler. An overview of cryptography, June 2012. http://www.garykessler.net/library/crypto.html.

[16] Dong Whee Kim and Jeong Beom Kim. Low-power carry look-ahead adder with multi-threshold voltage cmos technology. In *Solid-State and Integrated-Circuit Technology, 2008. ICSICT 2008. 9th International Conference on*, pages 2160 –2163, oct. 2008.

[17] P. Kocher. Timing attacks on implementations of diffie-hellman. *DSS, and Other Systems, CRYPTO96, LNCS 1109*, pages 104–113, 1996.

[18] P. Kocher, J. Jaffe, , and B. Jun. Differential power analysis. In *Advances in Cryptology: Proceedings of CRYPTO*, August 1999. 99, ser. Lecture Notes in Computer Science, M. Wiener, Ed., vol. 1666. Santa Barbara, CA, USA: Springer-Verlag, pp. 388397.

[19] O. Kommerling and M. G. Kuhn. Design principles for tamper-resistant smartcard processors. pages 9–20, 1999. Proceedings of the USENIX Workshop on Smartcard Technol- ogy (Smartcard 99).

[20] Stefan Mangard. Masked dual-rail pre-charge logic: Dpa-resistance without routing constraints. In *Systems CHES 2005, 7th International Workshop*, pages 172–186. Springer, 2005.

[21] Alfred J. Menezes, Paul C. Van Oorschot, Scott A. Vanstone, and R. L. Rivest. Handbook of applied cryptography, 1997.

[22] P. L. Montgomery and B. P. L. Montgomery. Without trial division modular multiplication, 44(170), 519-521. 1985.

[23] M. Nassar, S. Bhasin, J.-L. Danger, G. Duc, and S. Guilley. Bcdl: A high speed balanced dpl for fpga with global precharge and no early evaluation. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pages 849 –854, march 2010.

[24] National Institute of Standards and Technology: Advanced encryption standard (AES). Supersedes fips pub 197. November 2001.

[25] J. Jaffe P. Kocher and B. Jun. Differential power analysis. *Lecture Notes in Computer Science*, 1666, 1999. 388-397.

[26] R. L. Rivest, A. Shamir, , and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 1978. pp. 120126.

[27] E. Oswald S. Mangard and T. Popp. *Power Analysis Attacks: Revealing The Secrets of Smart Cards*. Springer- Verlag, 2007.

[28] S.J.A.V. Sebastian and J.A.P. Reyes. Logic style comparison using 32-bit cla in 90nm technology. In *Modelling Symposium (AMS), 2011 Fifth Asia*, pages 265 –269, may 2011.

[29] A. Shamir and E. Tromer. Acoustic cryptanalysis. *Preliminary proof-of-concept presentation*, 2004. http://www.wisdom.weizmann.ac. il/ tromer/acoustic/.

[30] W. Stallings. Cryptography and network security principles and practice. *Prentice Hall*, 1998.

[31] Daisuke Suzuki and Minoru Saeki. Security evaluation of dpa countermeasures using dual-rail pre-charge logic style. In *Proceedings of the 8th international conference on Cryptographic Hardware and Embedded Systems*, CHES'06, pages 255–269, Berlin, Heidelberg, 2006. Springer-Verlag.

[32] Daisuke Suzuki, Minoru Saeki, and Tetsuya Ichikawa. Random switching logic: A countermeasure against dpa based on transition probability. Technical report, on Transition Probability, IACR ePrint, 2004.

[33] K. Tiri and I. Verbauwhede. A logic level design methodology for a secure dpa resistant asic or fpga implementation. In *Design, Automation and Test in Europe Conference and Exhibition, 2004. Proceedings*, volume 1, pages 246 – 251 Vol.1, feb. 2004.

[34] Lu Xiao and Howard M. Heys. A simple power analysis attack against the key schedule of the camellia block cipher. *Information Processing Letters*, 95(3):409–412, August 2005.

[35] Y. Zhou and D. Feng. Side-channel attacks : Ten years after its publication and the impacts on cryptographic module security testing. pages 1–34.

Vita Auctoris

Ashley was born and raised in Windsor, Ontario. After completing her Bachelor of Applied Science Degree in Electrical Engineering at the University of Windsor, she pursued her Master of Applied Science. With an undergraduate background primarily focussed in the field of digital communications, she went on to expand her experience by working in the field of electronics in her post-graduate research.