University of Cincinnati

Date: 7/10/2012

I, Aditi Vijaykumar, hereby submit this original work as part of the requirements for the degree of Master of Science in Computer Engineering.

It is entitled:
DPA Resistance of Cryptographic Circuits Considering Temperature and Process Variations

Student's name:  Aditi Vijaykumar

This work and its defense approved by:

Committee chair: Ranganadha Vemuri, PhD
Committee member: Carla Purdy, PhD
Committee member: George Purdy, PhD

2867
DPA Resistance of Cryptographic Circuits Considering Temperature and Process Variations

A thesis submitted to the

Division of Research and Advanced Studies
of the University of Cincinnati

in partial fulfillment of the requirements
for the degree of

Master of Science

in the Department of Computer Engineering
of the College of Engineering
University of Cincinnati
July 2012

by

Aditi Vijaykumar

B.Tech Electronics and Communication Engineering
Indira Gandhi Institute of Technology, Delhi, India
June 2007

Thesis Advisor and Committee Chair: Dr Ranga Vemuri
Abstract

Integrated circuits are used for many applications including cryptographic systems. Differential Power Analysis (DPA) attacks are gaining prominence as a threat to information security of cryptographic devices. DPA attacks exploit the data dependency of power dissipated in a cryptographic system, using statistical methods that correlate and rank a set of keys to the power dissipation of the device based on the input data being encrypted. The ease of a DPA attack is determined by the number of input data encryptions and corresponding device power measurements that are required to retrieve the key with statistical confidence. A typical attack measures the correlation between model power traces generated during encryption of some input data using various guess keys, and device power traces obtained by encrypting the same input data with the unknown correct key. Potential keys are rated using this correlation metric.

The correlation metric and the number of traces depend on the variance of power values for a fixed cryptographic operation across many encryption power traces. The variance in turn depends on various factors such as process, voltage and temperature variations.

This thesis explores the effect of temperature and process variations on the ease of Differential Power Attacks on SPICE implementations of two cryptographic algorithms, namely KEELOQ and a single S-BOX of DES.

Exhaustive SPICE simulations are performed in Synopsys HSPICE® and Synopsys Nanosim® to determine a pattern in variance of power with respect to temperature and process variations during the encryption phase, with and without secure logic styles. Attacks are performed on these implementations using the SCARF [1] and DPA Attack Flow [2]
utilities developed at Digital Design Environments Lab, University of Cincinnati.

We show that the ease of a power analysis attack diminishes due to decrease in carrier mobility and therefore dynamic power variance with increasing temperature. We also show that using high threshold voltage devices can increase protection against power attacks at any temperature and reduce variance in power. Process variations are also shown to increase security by decreasing the power variance at a particular temperature.
Acknowledgments

I would like to thank God for being my silent witness and source of hope. I am indebted to my parents Srimati Prasanna Vijaykumar and Dr. Vijaykumar Krishnan. God could not be everywhere and that is why He gave us mothers. Thank you, Amma, for your unwavering belief in my abilities, unconditional love and constant effort to make life comfortable for me in every possible way. You are an exceptional mother and a wonderful human being.

Appa- you truly are my rock. Thank you for giving me the gift of music and love for engineering. Your life is an inspiration, not only for me but for many people like myself. Thank you, also, for the help with the essential concepts of statistics and results analysis, required during the course of this thesis.

I would especially like to thank my thesis advisor Dr. Ranga Vemuri, to whom I owe everything I am today. Sir- the Sankrit word Guru, meaning teacher, stands for Dispeller of Darkness. You are the embodiment of this word for me. Thank you for your belief in my abilities, patience and understanding, especially during my period of employment. It has been an honour and great learning opportunity, to be your pupil and I hope it continues lifelong.

Thanks also to all the professors at University of Cincinnati for the invaluable knowledge I gained during the Masters’ program. Thanks especially to Dr George Purdy and Dr Carla Purdy for taking time off their busy schedules to be on my defense committee. A special note of thanks to Ms Julie Muenchen, Graduate Program Director at CEAS, for her help during the course of this Masters’. Thanks also to Mr. James Tenney and everyone at the ISSO for their prompt assistance with important matters. Thanks to Ms. Katrina Bowling-Bergman, Mr John Tebo at UC Chemistry-Biology Library for the fun times and lifelong experiences, during my tenure as a student worker. Thanks also to Ms Rose Bruns, Mr Ted Baldwin and Mr Jim Clasper for the opportunity to work as Graduate Assistant.
at the CEAS Library - it was a great learning experience. Thanks to UC Libraries for the amazing technical resources.

I would like to thank my wonderful, extremely hardworking and brilliant sister Anaga Vijaykumar for loving me and being my side in every situation, especially in the last few years. Life is more beautiful with you around.

I would like to especially acknowledge Mike Borowczak for his efforts-thanks for the time on the telephonic conferences and for reviewing my thesis draft. Thank you Mike, for suggesting additional methods of experimentation as also help with results analysis. Many thanks to Greg Mefford for the help with the SCARF interface and expert opinion on many aspects of my work inspite of his work schedule. I would like to thank Antar for the help with the DES attack software. I would also like to thank Lakshmi and Arun, for their honest opinions. Thanks to Dr. Shubhankar Basu and Shikhanshu Agarwal for discussions on process variation modeling. Thanks also, to Mr. Robert Montjoy, for helping with the essential software installations required to run my experiments.

A special thank you to Romana Fernandes for being like an elder sister to me always- you are my inspiration and I love you with all my heart. Thanks also to Dr. Arun Janarthanan for all the invaluable help and guidance. Many thanks to Ms Hiromi her love and support when it was most needed. Thanks also to my great friend and partner in crime Mansi Krishan for the wonderful time spent as roommates. Thanks to Ghazal, Neha, Nikhil, Taher, Kovid, Aseem, Garima, Nanda, Kapil and Bala for the fun times in Cincinnati. Thanks also to Brendon for being such a wonderful source of support and my best friend here in Cincinnati. Thanks to Harshdeep Kamal for standing by me always and for the unconditional love your family has shown to me and my family. I would like to thank my good friend Nitin for being a source of constant encouragement and help alongside of being the most interesting person I have ever met!
I would also like to thank my paternal grandparents, Srimati Akhila Krishnan and Sri P.N. Krishnan for their help in facilitating the Masters’ program in the best way they could. Thank you, Patti and Thatha, for all the love and prayers.

In the end, I would like to dedicate this work to my wonderful, loving grandparents, Srimati Chandra Subramanyam and Sri Sanjeevi Subramanyam for their blessings and belief in my dreams. Thatha and Papa Patti- I am sure you are smiling upon me from somewhere...
# Contents

Title Page ................................................................. i  
Abstract ........................................................................... ii  
Acknowledgments ............................................................ v  
Table of Contents ............................................................. viii  
List of Figures .................................................................... x  
List of Tables ..................................................................... xii  

1 Introduction 1  
1.1 Introduction to Cryptography .............................................. 2  
1.1.1 Security requirements for a cryptographic device .............. 2  
1.1.2 Symmetric and Asymmetric Cryptography ......................... 2  
1.2 Security of Cryptographic Devices ....................................... 3  
1.3 Background ..................................................................... 5  
1.3.1 Power Analysis Attacks .................................................. 5  
1.3.2 Hypothetical Power Consumption Models ......................... 9  
1.3.3 Characteristics of Power Traces ....................................... 10  
1.4 Existing Countermeasures for Power Analysis Attacks .......... 14  
1.4.1 Hiding ....................................................................... 15  
1.4.2 Wave Dynamic Differential Logic .................................... 21  
1.5 Motivation for research ..................................................... 22  
1.5.1 Previous Work ............................................................ 22  
1.5.2 Proposed Countermeasure ............................................. 24  
1.5.3 Thesis Outline ............................................................ 24  

2 DES and KeeLoq Cryptographic Algorithms 26  
2.1 Data Encryption Standard (DES) ......................................... 26  
2.1.1 Structure ................................................................. 26  
2.1.2 DPA Attacks on DES .................................................. 30  
2.2 KeeLoq .......................................................................... 32  
2.2.1 Structure ................................................................. 33  
2.2.2 DPA Attacks on KeeLoq ............................................... 33
3 Effect of Temperature Variation on Differential Power Analysis 35
  3.1 Temperature Dependent MOSFET Parameters 35
      3.1.1 Threshold Voltage 36
      3.1.2 Carrier Mobility 36
      3.1.3 Saturation Velocity 36
      3.1.4 Parasitic Drain Source Resistance 37
  3.2 Impact of Technology Scaling on Threshold Voltage and Carrier Mobility 37
      3.2.1 Effect of Temperature Variation on Gate Overdrive and Drain Current 39
  3.3 Effect of Temperature Variation on Ease of Power Analysis Attacks 40
  3.4 Experimental Results 42
      3.4.1 KeeLoq 42
      3.4.2 DES 45
  3.5 Conclusion 48

4 Using High Threshold Voltage Cells in Cryptographic Circuits 50
  4.1 Experiments With High Threshold Voltage Cells 53
      4.1.1 KeeLoq 54
      4.1.2 DES 60
  4.2 Experiments with Wave Dynamic Differential Logic 62
      4.2.1 WDDL Experiments with KeeLoq 63
      4.2.2 WDDL Experiments with DES 69
  4.3 Conclusion 71

5 Consideration of Process Variations 72
  5.1 Experimental Setup 73
      5.1.1 Latin Hypercube Sampling 74
  5.2 Experimental Results 76
      5.2.1 Standard Threshold Voltage 76
      5.2.2 High Threshold Voltage 79
  5.3 Conclusion and Future Directions 84
      5.3.1 Future Directions 85

Bibliography 86
List of Figures

1.2 Schematic Description of DPA Flow [4] ........................................ 8
1.3 Device and simulation model power trace correlation [4] ................. 13

2.1 Overview of the DES Algorithm [5] .............................................. 27
2.2 The Feistel Function Block [5] .................................................. 29
2.3 Overview of the DES Key Schedule [5] ....................................... 30
2.4 Reference traces and difference traces for a correct and 3 incorrect keys [6] [3] 32
2.5 KEELOQ Encryption [7] ......................................................... 33

3.1 Temperature dependence of threshold voltage in n-Channel MOSFETs [8] [9] 38
3.2 KEELOQ: Average power variance trace with respect to temperature over five keys .............................. 43
3.3 KEELOQ: Average number of traces increases with respect to temperature over five keys ................................. 44
3.4 KEELOQ: Average correlation decreases with increase in temperature over five keys ................................. 45
3.5 DES: Average power variance trace with respect to temperature over five keys .............................. 46
3.6 DES: Average number of traces increases with temperature ............... 47
3.7 DES: Average correlation vs temperature over five keys .................. 48

4.1 WDDL NAND cell schematic diagram [4] ................................. 51
4.2 WDDL flip flop cell schematic diagram [4] ............................... 52
4.3 Organization of a WDDL circuit [4] ........................................ 53
4.4 KEELOQ: Average power variance vs temperature using high threshold voltage cells ................................. 54
4.5 KEELOQ: Average number of traces increases with respect to temperature over five keys using High Threshold Voltage Cells .............................. 55
4.6 KEELOQ: Average correlation with respect to temperature over five keys using High Threshold Voltage Cells .............................. 56
4.7 KEELOQ: comparison of effective power variance within average power variance trace - DFF(high $V_{th}$) vs NLF(high $V_{th}$) .............................. 59
4.8 DES: Average Power variance (Standard $V_{th}$) ............................. 60
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.9</td>
<td>DES: Average power variance (High $V_{th}$)</td>
<td>61</td>
</tr>
<tr>
<td>4.10</td>
<td>DES: Comparison of effective power variance within average power variance trace -std $V_{th}$ vs 50% high $V_{th}$</td>
<td>61</td>
</tr>
<tr>
<td>4.11</td>
<td>KEELOQ(WDDL): Number of traces comparison-std $V_{th}$ vs high $V_{th}$</td>
<td>64</td>
</tr>
<tr>
<td>4.12</td>
<td>KEELOQ(WDDL): Correlation comparison-std $V_{th}$ vs high $V_{th}$</td>
<td>65</td>
</tr>
<tr>
<td>4.13</td>
<td>KEELOQ(WDDL): comparison of effective power variance within average power variance trace-std $V_{th}$ vs high $V_{th}$</td>
<td>66</td>
</tr>
<tr>
<td>4.14</td>
<td>KEELOQ(High $V_{th}$): comparison of effective power variance within average power variance trace with and without WDDL cells</td>
<td>67</td>
</tr>
<tr>
<td>4.15</td>
<td>KEELOQ: Overall comparison of effective power variance within average power variance trace</td>
<td>68</td>
</tr>
<tr>
<td>4.16</td>
<td>KEELOQ: Overall average traces comparison</td>
<td>69</td>
</tr>
<tr>
<td>4.17</td>
<td>DES(WDDL): Plot of change in correlation with number of vectors</td>
<td>70</td>
</tr>
<tr>
<td>5.1</td>
<td>Density function of variable $X_i$ divided into $N$ non-overlapping intervals [10]</td>
<td>74</td>
</tr>
<tr>
<td>5.2</td>
<td>Pairing $X_1$ with $X_2$ when $N=5$ [10]</td>
<td>75</td>
</tr>
<tr>
<td>5.3</td>
<td>Effective power variance across average power variance trace, with process variation is 11% lesser than without process variation</td>
<td>76</td>
</tr>
<tr>
<td>5.4</td>
<td>Number of traces with process variation is 50% more than without process variation</td>
<td>77</td>
</tr>
<tr>
<td>5.5</td>
<td>Average, best and worst case effective power variance across average power variance trace, comparison</td>
<td>78</td>
</tr>
<tr>
<td>5.6</td>
<td>Average, best and worst case effective standard deviation across average standard deviation trace, comparison</td>
<td>78</td>
</tr>
<tr>
<td>5.7</td>
<td>Average, best and worst case number of traces comparison</td>
<td>79</td>
</tr>
<tr>
<td>5.8</td>
<td>Effective power variance across average power variance trace, with process variation is 22% lesser than without process variation</td>
<td>80</td>
</tr>
<tr>
<td>5.9</td>
<td>Number of traces with process variation is 8% more than without process variation</td>
<td>81</td>
</tr>
<tr>
<td>5.10</td>
<td>Average, best and worst case effective power variance across average power variance case comparison</td>
<td>82</td>
</tr>
<tr>
<td>5.11</td>
<td>Average, best and worst case effective standard deviation across average standard deviation trace comparison</td>
<td>82</td>
</tr>
<tr>
<td>5.12</td>
<td>Average, best and worst case number of traces comparison</td>
<td>83</td>
</tr>
</tbody>
</table>
# List of Tables

1.1 Truth Table for DDL AND gate - exactly one output transition per Precharge/Evaluate Cycle [6] ................................. 21
1.2 Truth Table for DDL XOR gate - exactly one output transition per Precharge/Evaluate Cycle [6] ................................. 21

3.1 KEELIQ: Number of traces required to recover the correct key ........ 43
3.2 KEELIQ: Correlation vs Temperature .................................. 44
3.3 DES: Average number of traces required to recover the correct key .... 47
3.4 DES: Average correlation vs temperature across five keys ............ 48

4.1 KEELIQ: Number of traces required to recover the correct key (high Vth) 55
4.2 KEELIQ: Correlation vs temperature (high Vth) .......................... 56
4.3 Standard and high threshold voltage values ............................... 56
4.4 KEELIQ: Number of traces required vs temperature, with std Vth and 26% high Vth ......................................................... 57
4.5 KEELIQ: Correlation vs Temperature (Std Vth vs 26% High Vth) .... 57
4.6 KEELIQ: Comparing number of traces required with different values of Vth 58
4.7 KEELIQ: Increase in number of device traces needed when using 50% High Vth cells ................................................................. 58
4.8 KEELIQ: Using High Vth in the KeeLoq NLF Combinational Block vs in the Flip Flops ........................................................... 59
4.9 DES: % Decrease in effective power variance within average power variance trace when using 50 % high Vth cells ......... 62
4.10 % Increase in number of traces when using 50 % high Vth cells ......... 62
4.11 KEELIQ(WDDL): Comparison of number of traces-high Vth vs std Vth 64
4.12 KEELIQ(WDDL): Correlation comparison -std Vth vs high Vth ...... 65
4.13 KEELIQ(WDDL): comparison of effective power variance within average power variance trace-std Vth vs high Vth ................. 66
4.14 KEELIQ(High Vth): comparison of effective power variance within average power variance trace with and without WDDL cells 67
4.15 DES(WDDL): effective power variance within average power variance trace with WDDL circuitry .............................. 70
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
</table>
Chapter 1

Introduction

We live in a day and age where embedded systems are responsible for data storage, access and manipulation in every field from finance to transport. This overwhelming dependence on electronics has reduced the need for physical presence during completion of tasks such as banking transactions and has reduced man-days at work. However, this convenience comes at the price of exposing us to dangerous security risks such as information theft that can impact important aspects of our lives. A system is only as secure as its weakest link. Hence, there is a need to make sure that an electronic device is secure with respect to every level of implementation, from the software security algorithm to the physical realization of the algorithm on a cryptographic chip at the transistor level. We must continually explore weak links that can compromise the security of a system and find ways to tackle them, while enhancing existing security mechanisms.

As per the Computer Security Institute 2011 report [12], understanding a security threat is the first step towards defending it. In keeping with this let us explore the many ways in which the security of a cryptographic system can be compromised. In order to do this we must also explore the various cryptographic systems in place
1.1 Introduction to Cryptography

1.1.1 Security requirements for a cryptographic device

Earlier, information security was an issue only with respect to communication systems (communication channels). Hence, Kocher et al [3], explain security requirements of embedded systems in terms of communication channels. They state that an ideal communication channel (embedded systems), although, accessible to attackers must provide security functions such as data confidentiality, data integrity and peer authentication. These three functions are defined by Kocher et al. [13] as follows:

(a) Data confidentiality: refers to protection of sensitive information from undesired eavesdroppers.

(b) Data integrity: refers to ensuring the legitimacy of data.

(c) Peer authentication: refers to verification (authentication) of sender and receiver in case of sensitive data.

1.1.2 Symmetric and Asymmetric Cryptography

Modern security systems use cryptographic algorithms to provide confidentiality, integrity and authenticity in keeping with the requirements stated in above. A cryptographic algorithm typically takes a message or a plaintext along with a secret key as input and produces a ciphertext as the output. This process is known as encryption. The reverse process i.e generating the original message/plaintext from the ciphertext is known as
decryption. In modern cryptography this algorithm itself is assumed to be known, so the security of the device hinges on the secret key used to encrypt the input.

Symmetric cryptographic algorithms use the same key for both encryption and decryption phases. They are further classified as either block or stream ciphers - block ciphers encrypt a block of text of fixed size and produce as output another block of fixed size. Examples include AES which has a standard block size of 128 bits whereas the key can be 128, 192 or 256 bits. Stream ciphers use a running key [2] to generate the output ciphertext one bit at a time. Encryption is usually performed by XORing the key bits with the plaintext bits. [4]

Asymmetric cryptographic algorithms use a combination of two keys, one of which is public and the other private. The private key is unique to every user and is secret. e.g RSA [4].

1.2 Security of Cryptographic Devices

The security of a cryptographic device depends on the encryption algorithm and on the implementation of this algorithm in the actual device. At first glance, algorithmic level security measures seem sufficient to deter attacks on cryptographic devices, owing to the amount of time and resources required for the intensive mathematical computation during such attacks. In [14], the authors describe an attack on the KeeLoq algorithm, that requires two days for successful completion in the best case and eight days in the worst case. However, the same algorithm can be attacked at the implementation level, within a few hours, using a powerful class of attacks called side channel attacks [15]. It is important, therefore, to examine all the vulnerabilities of any cryptographic system. Two broad classifications exist.
(1) **Passive attacks**: They constitute a class of attacks wherein the secret key of the device is recovered during normal operation mode, by observing the physical properties of the device e.g. power consumption, execution time etc.

(2) **Active attacks**: They constitute a class of attacks wherein the inputs to the device or the environment of operation e.g. temperature are tampered with. The abnormalities in operation occurring as a result are used to yield the secret key of the device.

Depending on the physical/logical interface of the device being exploited, attacks can be further classified as follows [4]:

**Invasive attacks** refer to a class of attacks wherein the device is taken apart (de-packaged) to probe its various parts. The probing is passive if data signals e.g. signals on a processor bus are only observed whereas it is active if the signals are altered in any way by laser cutters, ion beams etc. This class of attacks is the most powerful, but also requires very expensive equipment.

**Semi-invasive** attacks also involve depackaging of the device. However no direct electrical contact is made with the chip in this case. The contents of memory cells are observed without probing the normal read-out circuitry. Semi-invasive attacks are focused on inducing faults in the device using X-rays etc. However finding the correct place to probe on a modern day complex chip requires time and expertise. The equipment required is less expensive than that for an invasive attack.

**Non-invasive** attacks involve probing of only those interfaces in the device, which are easily accessible. There is no physical damage done to the device, so no trails are left behind. Consequently the attack itself is inexpensive and efficient. Hence, such attacks pose a serious threat owing to their subtlety.
Passive non-invasive attacks, also referred to as side channel attacks use power, timing or electromagnetic signals of the device to yield the secret key. This class of attacks has received widespread attention from the research community.

Active non-invasive attacks involve inducing a fault by changing the environment of the device without tampering with it, such as altering the temperature.

This thesis focuses on a particular class of Side Channel Attacks that involve power analysis of the cryptographic device under consideration.

1.3 Background

1.3.1 Power Analysis Attacks

Power analysis attacks exploit the data and operation dependency of power consumption during the operation of a cryptographic device [4].

Simple power analysis (SPA) attacks exploit the operation dependency of power. An attacker who understands the cryptographic algorithm being implemented can figure out different operations in the course of the encryption via visual inspection [4] of the device power traces and thereby recover the secret key of the device. Such attacks do not require many device power traces to accomplish the task. For example, the DES algorithm execution consists of 16 rounds of encryption. This translates into 16 distinct power peaks as shown in Figure 1.1

Figure 1.1: Power traces for 16 rounds of DES encryption [3]
Differential power analysis (DPA) attacks are one of the most popular and powerful methods of compromising the security of a device. They do not require a knowledge of the implementation of the underlying cryptographic algorithm. However they involve statistical computations based on garnering many device traces. This class of attacks was first proposed in a seminal paper by Kocher et al [3] in 1993. Whereas SPA focuses on recovering the key based on the variation in the power due to execution of certain operations, DPA focuses on variation in the power due to the data values being handled in the device. The general five step procedure below for DPA attacks has been described in [4]:

1. **Choosing an intermediate result of the executed algorithm** The intermediate result is produced by a function $f(d,k)$ where $d$ is a data value being decrypted (ciphertext) and $k$ is a small part of the key. Since the result is a function of $k$ it can be used to reveal $k$.

2. **Measuring the power consumption** In this step we record power measurements corresponding to encryptions of $D$ different plaintexts. Each power trace $d$ consists of a finite number of power samples $T$. It is important for the power traces to be aligned in time i.e. each column of the the matrix $D \times T$ should contain power values caused by the same stage of decryption.

3. **Calculating hypothetical intermediate values** We generate $K$ part key guesses and feed them to the intermediate function $f(d,k)$ where $d$ corresponds to each of the input $D$ plaintexts. By this we end up with a $D \times K$ sized matrix. Each column in the matrix corresponds to a partial key guess and each row corresponds to a data input. The objective is to find out which column was processed in the device under attack for all the rows.
(4) **Mapping intermediate values to hypothetical power consumption values**

Using the intermediate data values captured in (c) above we generate hypothetical power traces using a Hamming Model. This exploits the data dependency aspect of power. In any cryptographic device implementing CMOS transistor logic, there are primarily four kinds of data transitions that can occur - $0 \rightarrow 0$, $0 \rightarrow 1$, $1 \rightarrow 0$, $1 \rightarrow 1$. Current flowing through the circuit to pull the output high or low will only be significant in the $0 \rightarrow 1$ or $1 \rightarrow 0$ transitions. Hence the significant power consumption happens between any two encryption rounds since there are numerous transitions in the data register that occur as a result of the encryption operation. To capture the number of transitions between two states of a data register we only need to XOR the contents. Hence when determining hypothetical power values, XORing the results of two subsequent intermediate values of the function $f(d,k)$ produces a rough measure of the power consumption of the real world device. In other words we generate Hamming model guess power traces corresponding to every part key guess.

(5) **Comparing the hypothetical power consumption values with the real power traces**

This step involves comparing each of the Hamming Model guess power traces for every input $d$ belonging to $D$ to the device trace corresponding to the input $d$. We generate a correlation coefficient for each of the guess traces. The guess key corresponding to the highest correlation is determined to be the correct key or the best guess.

However this only reveals part of the key. In order to reveal the full key we save the keys corresponding to the top few "best guess" columns in the above matrix $D \times K$ and repeat this steps (c) to (e) to generate the next few bits of the key. In this manner we gradually build up the complete key.
The final three steps of DPA as described above are depicted in Figure 1.2

![Diagram of DPA Flow](image)

Figure 1.2: Schematic Description of DPA Flow [4]
1.3.2 Hypothetical Power Consumption Models

In order to attack a real world cryptographic device by exploiting its power dissipation, we need a suitable model of the power consumption of the device for the sake of comparison in the last step of the DPA procedure described above. We also need to know the minimum level of implementation of this model i.e it could be a software implementation of the underlying cryptographic algorithm, a behavioural/structural implementation, a transistor level implementation, or a layout level implementation. Each requires more design effort than the previous, but also affords more accurate modeling of power consumption. The basic idea, however is to be able to adequately model the exploitable power consumption of the real device with maximum possible accuracy and minimum design effort.

The most common models of power consumption that can be implemented at the software level with least effort are the Hamming Distance and Hamming Weight Models.

Hamming Weight Model

Hamming Weight of any binary number is the number of 1’s in that number. For example, Hamming Weight of a number 11011 is 4. An attacker using this model assumes that the power consumption of the device is directly proportional to the number of bits that are set in the data value being processed. The data values processed before and after this value are ignored. In a traditional CMOS circuit, the dynamic power consumption due to logic transitions between data values represent the data dependent power consumption to a great extent. Therefore a model that represents logic transitions is better suited for the purpose of an attack. The Hamming Weight model is therefore not as suitable.

Still the Hamming Weight of a data value is not completely unrelated to the power consumption caused by the processing of this value. If we consider two values $v_0$ and $v_1$ processed consecutively, assuming that $v_0$ is an all zero value, and $v_1$ has some of its bits
set, then the number of set bits in \( v1 \) is directly proportional to the power consumption.

If \( v0 \) and \( v1 \) are non zero constant values i.e \( v1 \) is always the value that is processed after \( v0 \), then also the power consumption is related to the Hamming Weight.

The case where bits of \( v0 \) and \( v1 \) are uniformly distributed is the worst case for Hamming Weight Models. [4]

**Hamming Distance Model**

The Hamming Distance between two binary values \( v0 \) and \( v1 \) is given as

\[
\text{HD} = v0 \odot v1
\]  

(1.1)

The Hamming Distance between any two binary values processed in a register consecutively is representative of the number of transitions (0→1 or 1→0) required to change one value to the next. Therefore this also roughly represents the dynamic power consumption of a circuit.

**Assumptions**

(a) All 0→1 and 1→0 transitions in a digital circuit lead to the same power consumption.

(b) All 0→0 and 1→1 transitions in a digital circuit lead to the same power consumption.

(c) Differences in parasitic capacitances of wires and cells are not considered.

(d) The Hamming Distance Model ignores static power consumption of cells. [4]

### 1.3.3 Characteristics of Power Traces

A successful power attack depends on the amount of *exploitable power consumption* in a power trace. Every point in a power trace is composed of four components as below:
Differential power analysis, as described before focuses on the data dependency of the power for a given operation in time. The greater is the variance in power, the easier it is to differentiate between power traces caused by different guess keys, in the processing of different data values due to a fixed cryptographic operation. If all input-key combinations cause the same amount of power consumption, then it becomes harder to recognize the correct key. This fact plays an important part in the design of secure logic styles which will be discussed later.

The exploitable power consumption for a given point in a trace as per the above therefore depends only on the variance in the data dependent component of the power \( \text{Var}(P_{\text{data}}) \) and the variance in the noise of the cryptographic device \( \text{Var}(\text{Noise}) \). Power measurements for a DPA attack are taken for a fixed operation hence \( \text{Var}(P_{\text{operation}}) = 0 = \text{Var}(P_{\text{const}}) \). In the case of simple power analysis attacks both \( \text{Var}(P_{\text{operation}}) \) and \( \text{Var}(P_{\text{data}}) \) are considered and consequently non-zero, SPA attacks focus on operation dependency of power.

**Signal to Noise Ratio**

The signal to noise ratio or SNR is a metric used to determine if the available exploitable power information is enough to perform a successful attack with minimum effort in terms of number of traces. Noise generated during the operation of a cryptographic device can obfuscate the exploitable power consumption. If the ratio of variance in exploitable power to that of variance in noise is high, then chances of a successful attack with minimum effort are higher. On the other hand if this ratio is low, then we need more device traces to establish the power variance needed to perform a successful attack.
The exploitable power consumption in case of DPA consists of $P_{\text{data}}$. Hence SNR is defined as [4]

$$\text{SNR} = \frac{\text{Var}(P_{\text{data}})}{\text{Var}(P_{\text{noise}})}$$  \hspace{1cm} (1.3)

### Relevant Points of Power Traces

Points in time of the power traces where the SNR is highest correspond to the points in time where the switching activities [4] is correlated to the power consumption. Hence for the purpose of an attack, it is important to separate the *relevant points of a power trace* from the other points thereby performing *compression* of traces. This makes the process of correlating the truncated power traces to the Hamming Distance model power traces easier. Since our setup considers the SPICE implementations as the actual device under attack, the power traces so recorded during encryption of various inputs, are processed using Perl scripts to extract the power peaks that are correlated to the Hamming Distance.

### Correlation

The authors in [4] have an example showing how to determine the relevant points of power traces for the purpose of an attack.

At first they record power traces of 1000 plaintexts sent to an actual AES ASIC implementation. Each plaintext is sent for encryption 100 times to the device and the average trace is calculated in order to eliminate noise. Therefore an average power trace is obtained for 1000 random plaintexts.

The same 1000 plaintexts are fed to a back annotated netlist of the AES chip and 1000 simulated power traces are recorded.
The simulated power traces and measured power traces are then analyzed together to determine any relationship between the two, the primary goal being to find out which points in time of the measured power traces are correlated to the transition counts recorded in the simulated power trace. Figure 1.3 shows the power peak in the second clock cycle as being correlated to the switching activity. The upper plot shows the power trace of the AES ASIC during four clock cycles. The lower plot shows the correlation between the measured power trace and the simulated number of transitions that occur during the second clock cycle.

![Power Trace Correlation](image)

Figure 1.3: Device and simulation model power trace correlation [4]

**Number Of Traces**

If the variance of the power values in all traces for a fixed operation in time is high then there is a greater probability that the traces themselves will be correlated to the Hamming Distance model traces that represent the switching activity. The objective in performing a successful attack is to maximize this variance for a fixed operation in time
Chapter 1: Introduction

E.g. a particular encryption round. However, variance is a statistical parameter of a normal distribution that needs to be determined across a certain minimum number of samples and in our case, traces. The minimum number of traces required to establish the value of this variance is 30. One can refer [4] for a more detailed discussion on the number of traces as a metric to determine the ease of a DPA attack.

The higher the variance in power for a fixed cryptographic operation in time, the lesser is the effort required in terms of number of traces to establish the correct key with reasonable correlation. As a designer therefore, one of the main ways of securing a cryptographic chip, is to decrease the variance in power i.e. prevent the attacker from being able to distinguish the switching activity based on the variance in power for a fixed operation in time. By decreasing the variance, more number of traces would be required to recover the secret key of the device and therefore higher effort.

1.4 Existing Countermeasures for Power Analysis Attacks

We start by examining existing DPA countermeasures. The main aim, as established early on, is to decrease or eliminate the power variance in a cryptographic circuit, as this decreases the signal to noise ratio and thereby makes it more difficult to perform a power analysis attack.

Security measures mainly exist at the algorithmic and the circuit level. Algorithmic countermeasures involve hiding the sensitive power consumption information by insertion of random operations or shuffling the existing cryptographic operations. The most comprehensive way of protecting a cryptographic circuit is however to make the power consumption independent of the processed data. Circuit level countermeasures such as Dynamic and Differential Logic enable this.
1.4.1 Hiding

"Power analysis attacks work because the power consumption of cryptographic devices depends on intermediate values of the executed cryptographic algorithms. Therefore, the goal of countermeasures is to avoid or at least to reduce these dependencies. In case of hiding, this is done by breaking the link between the power consumption of the devices and the processed data values. Hence, cryptographic devices that are protected by hiding execute cryptographic algorithms in the same way as unprotected devices. In particular, they calculate the same intermediate values. Yet, the hiding countermeasures make it difficult for an attacker to find exploitable information in power traces.

The goal of hiding countermeasures is to make the power consumption of cryptographic devices independent of the intermediate values and independent of the operations that are performed. There are essentially two approaches to achieve this independence. The first approach is to build devices in such a way that the power consumption is random. This means that in each clock cycle a random amount of power is consumed. The second approach is build devices that consume an equal amount of power for all operations and for all data values. Hence, equal amounts of power are consumed in each clock cycle.

It is not possible to achieve perfectly equal or randomized power consumption in practice. However there are two widely known ways of achieving these goals to a great extent. One method involves randomizing the cryptographic operations across the time scale. The other method involves directly changing the power consumption characteristics of the performed operations, thereby affecting the amplitude dimension of the power consumption.

**Time Dimension Hiding**

As per the DPA attack methodology, recorded power traces should be perfectly aligned. This means that the power consumption of each operation should be located
at the same position in each power trace. If this condition is not fulfilled, DPA attacks require significantly more power traces. This observation is the motivation for designers of cryptographic devices to randomize the execution of the cryptographic algorithms, i.e. the devices perform the operations of the algorithms at different moments of time during each execution. This makes the power consumption appear to be more or less random for an attacker. The more random the execution of an algorithm is, the more difficult it becomes to attack the device. The most commonly used techniques to randomize the execution of cryptographic algorithms are the random insertion of dummy operations and the shuffling of operations.

**Random Insertion of Dummy Operations:**

The basic idea of this technique is to randomly insert dummy operations before, during, and after the execution of the cryptographic algorithm. Each time the algorithm is executed, randomly generated numbers are used to decide how many dummy operations are inserted at these different positions. It is important that the total number of inserted operations is equal for all executions of the algorithm. In this way, attackers cannot get any information about the number of inserted operations by measuring the execution time of the algorithm. In an implementation that is protected by this approach, the position of each operation depends on the number of dummy operations that have been inserted before this operation. This number randomly varies from execution to execution. The more this position varies, the more random the power consumption appears. However, it is also clear that the more dummy operations are inserted, the lower is the throughput of the implementation. This is why in practice, a suitable compromise needs be found for every implementation.

**Shuffling of Operations:**

An alternative to the insertion of dummy operations is the shuffling of operations...
of the cryptographic algorithm. The basic idea of this approach is to randomly change the sequence of those operations of a cryptographic algorithm that can be performed in arbitrary order. In case of AES, 16 S-box look-ups need to be performed in every round. These look-ups are independent of each other. Hence, they can be performed in arbitrary order. Shuffling these operations means that during each execution of AES, randomly generated numbers are used to determine the sequence of the 16 S-box look-ups. Shuffling randomizes the power consumption in a similar way as the random insertion of dummy operations. However, shuffling does not affect the throughput as much as the random insertion of dummy operations. The disadvantage of shuffling is that it can only be applied to a certain extent. The number of operations that can be shuffled in a cryptographic algorithm are limited. This number depends on the algorithm and on the architecture of the implementation. In practice, shuffling and the random insertion of dummy operations are often combined.

**Amplitude Dimension Hiding**

These techniques lower the leakage of exploitable information of a cryptographic device by lowering the Signal-to-Noise (SNR) Ratio, already described in Chapter 1. Ideally, the goal is to set the SNR to 0. This can be achieved by setting $\text{Var}(P_{\text{exp}}) = 0$ or by increasing $\text{Var}(P_{\text{noise}})$ to infinity. In order to reduce $\text{Var}(P_{\text{exp}})$ to 0, the power consumption needs to be exactly equal for all operations and data values. Increasing $\text{Var}(P_{\text{sw.noise}} + P_{\text{el.noise}})$ to infinity means that the amplitude of the noise needs to be infinitely increased. In practice, $\text{Var}(P_{\text{exp}})$ can be reduced to small values and $\text{Var}(P_{\text{noise}})$ can be increased to large values. However, there is no countermeasure that reaches the ideal goal of $\text{SNR} = 0$.

The signal-to-noise ratio of an operation, i.e. the leakage of an operation, can be lowered by either increasing the noise or by lowering the signal.
(a) Increasing the Noise: The goal is to build a device where random switching activity dominates the power consumption.

Performing several independent operations in parallel achieves this objective. Hence, hardware architectures of cryptographic algorithms with a wide datapath are better than architectures with a narrow datapath. For example, it is more difficult to attack the power consumption of a single bit in a 128-bit architecture of AES than in a 32-bit architecture. Another way to increase the noise is to use dedicated noise engines. Noise engines perform random switching activities in parallel to the actual operations.

(b) Reducing the Signal: The goal is to build a device where all operations require an equal amount of power for all data inputs.

This can be done using dynamic and differential logic. CMOS logic consumes power only when output transitions from 0→1. Thus, instantaneous power consumed by a circuit depends on the current required to charge the output capacitance. This makes it possible to predict the input data by analyzing the power consumption, thus, facilitating DPA attacks. If all input data consume the same (constant) amount of instantaneous power for each and every input data, then, there would be no premise for attacks using power distinctions. To eliminate correlation of the power consumption to input data by achieving constant power consumption, the logic style must charge/discharge a constant value of total capacitance for every input data (cycle). Dynamic and Differential logic (DDL) enables this functionality. [6] [4]. style. ” [6] [4].

**Dynamic and Differential Logic:**

There are two key requirements for dynamic and differential logic to achieve constant instantaneous power consumption for every input.
1. **Exactly one transition \(0\rightarrow1\) must occur in every cycle**

This requirement can be split into two sub-conditions.

(a) The circuit must contain 100% output switching in each cycle. Differential logic (in DDL) introduces complementary logic in the circuit. So, for any input data, either the uncomplemented output or the complemented output switches, thereby ensuring 100% switching in each cycle.

(b) Exactly one charging/discharging must occurs for every cycle. Irrespective of the logic style used (static-CMOS or dynamic-CMOS), this requirement can be satisfied by mimicking dynamic logic operation. The dynamic circuit operation is divided into two states: Precharge state and Evaluation state. During the precharge state, all the nodes in the circuit are precharge. During the Evaluation state, depending on the logic function and input data, some of the nodes (with equal capacitances in every evaluation cycle) in the circuit are discharged.

Upon analyzing each of the 4 possible output transitions, we find the following:

(i) \(0\rightarrow0\) transition cycle - initially output \(Y\) (uncomplementary output) is '0', but output \(Y\) (complementary output) is '1'. Then, during precharge both the outputs become '0'. So, there is exactly one discharging as output \(Y\) discharges to '0'. In the evaluation that follows output \(Y\) remains '0', but output \(Y\) charges to '1'. So, there is exactly one charging. Thus, exactly one charging/discharging, and 100 % switching in the cycle is achieved.

(ii) \(0\rightarrow1\) transition cycle - initially output \(Y\) (uncomplementary output) is '0', but output \(Y\) (complementary output) is '1'. Then, during precharge both the
outputs become '0'. So, there is exactly one discharging as Ybar discharges to '0'. In the evaluation that follows output Y charges to '1', but Ybar remains to '0'. So, there is exactly one charging. Thus, exactly one charging/discharging, and 100 % switching in the cycle is achieved.

(iii) 1→0 transition cycle - initially output Y (uncomplementary output) is '1', but output Ybar (complementary output) is '0'. Then, during precharge both the outputs become '0'. So, there is exactly one discharging as output Y discharges to '0'. In the evaluation that follows output Y remains '0', but output Ybar charges to '1'. So, there is exactly one charging. Thus, exactly one charging/discharging, and 100 % switching in the cycle is achieved.

(iv) 1→1 transition cycle - initially output Y (uncomplementary output) is '1', but output Ybar (complementary output) is '0'. Then, during precharge, both the outputs become '0'. So, there is exactly one discharging as output Y discharges to '0'. In the evaluation that follows output Y remains '0', but output Ybar charges to '1'. So, one charging. Thus, exactly one charging/discharging, and 100 % switching in the cycle is achieved.

(2) The capacitances of the differential logic charged/discharge in every cycle must remain constant. The condition that each cell has exactly one transition every cycle, is necessary, but not sufficient condition. In order to achieve constant power consumption, fixed amount of charge must be drawn and discharged in every cycle. Hence, the capacitances of the differential (uncomplementary or complementary) logic charged/discharged in every cycle must remain constant.” [6]

Table 1.1 and Table 1.2 represent the truth tables for DDL AND XOR gates respectively.
Chapter 1: Introduction

1.4.2 Wave Dynamic Differential Logic

"Wave Dynamic Differential Logic" was introduced by Tiri et al in [16]. WDDL also uses De-Morgan’s laws as per equations 4.1 and 4.2 to generate the complementary output. Every cell (AND or OR) has a complementary output generated by complementary gate (OR or AND respectively) from complementary inputs in parallel. Using positive monotonic gates like AND and OR gates ensure that for an all zero input, the output is zero. The output is 1 only if there is a 0 to 1 transition.

\[
\overline{A} \cdot \overline{B} = \overline{A} + \overline{B}
\]  

(1.4)
\[ A + B = \bar{A} \bar{B} \] (1.5)

To implement the dynamic behavior WDDL uses an alternate precharging technique, which is to propagate a precharge wave through the inputs of all static-CMOS gates [16]. It uses the property of positive gates like AND/OR that the output is '0' when all the inputs are '0'. The precharge wave is implemented by forcing the differential inputs (uncomplemented and complemented) of all the gates in the circuit to 0s, simultaneously, thereby forcing the gate to a precharge state via inputs. When the differential inputs are 0s, differential outputs (uncomplemented and complemented) have to become 0, and these outputs are inputs to another set of gates. This causes all subsequent gates in the circuit to be precharged. Thus, the precharge 0s pass through the circuit like a wave making all the gate outputs in the circuit 0. This set of precharge inputs are then followed by evaluation inputs. Thus, the inputs are interlaced with '0s' to bring forth the precharge cycle after every set of inputs. Thus, no separate precharge control signal is required to force outputs to 0.

WDDL uses only positive gates to ensure that precharge wave passes through the circuit. This along with the fact that WDDL complements every gate to get the complementary output, results in at least 2X times increase in area and 3.5X times increase in power consumption compared to static CMOS logic” [6].

1.5 Motivation for research

1.5.1 Previous Work

As described above, variance in power values is a key factor in determining the success of a power analysis attack. Therefore factors affecting this variance also must be considered as they indirectly contribute to the degree of success of an attack. Primary
Chapter 1: Introduction

Factors affecting power in general are process, temperature and voltage variations. Existing research work regarding the relationship between ease of attack in terms of number of traces, and process, temperature and voltage variations will be discussed next, alongside methodologies to counter the effects of PVT variations on ease of attack. Most of the research work below uses a single S-BOX of DES as the cryptographic algorithm of choice.

In [17], the authors analyze the statistical properties of data dependent power with 45nm predictive CMOS device models and ITRS process variation models. They model a "Power Attack Tolerance" metric to model dynamic and leakage power data dependence and show that this metric can significantly deteriorate due to inter and intra die process variations. A transistor sizing optimization is also proposed to reduce the deterioration with minimal power and area overhead.

The authors of [18] show that the effectiveness of Leakage Power based attacks in the presence of process variations are a serious threat to information security of cryptographic circuits in sub-100nm technologies. Process variations are mainly classified into two types: inter die process variations which affect all transistors on a chip in the same way, and intra die process variations which affect different transistors on the same chip in different ways. In [18] it is also stated and shown that the degrading effects on leakage power based attack tolerance caused by inter die process variation can be mitigated by intra die process variation.

The work in [19] is a comprehensive analysis of the effect of process and temperature variations on Leakage Power Attacks. Temperature variations in particular are considered for the first time in this work alone. However they only analyze effect of temperature variations on LPA attacks. There is no consideration of dynamic power based attacks. The authors in [19] go on to show that the leakage power at any temperature is directly proportional to the Hamming Weight. Leakage power increases with temperature expo-
nentially in sub-90nm circuits independent of process variations, hence the aforementioned
trend between leakage power and Hamming Weight is retained. While it is mentioned in
[19] that the effectiveness of leakage power attacks can be said to remain unaffected given a
constant die temperature, the authors also go on to say that the exact effect of temperature
variation on power attacks is yet to be examined. This is the motivating factor for the
thesis.

1.5.2 Proposed Countermeasure

As will be discovered in the course of this thesis, power variance of cryptographic
devices has been found to increase with decrease in temperature, due to higher carrier
mobility at lower temperatures. This directly impacts the security as an increase in power
variance leads to a decrease in the number of traces required to recover the secret key.
To counter the higher power variance at lower temperatures, we propose the use of high
threshold voltage cells. High threshold voltage cells provide less of a current drive, and
therefore a decreased power consumption, that automatically leads to lower power variance
as will be shown in the experiments performed during the course of this thesis.

We also find that process variations decrease the power variance further and in-
crease the security of the device.

1.5.3 Thesis Outline

Chapter 2 will expound on the details of the DES and KEELOQ algorithms, as
well as power analysis attack mechanisms implemented on them. Chapter 3 will deal with
the effects of temperature variation on ease of DPA and associated experiments. Chapter
4 expounds on the use of high threshold voltage devices to counter temperature variation
effects on DPA with and without secure logic styles like Wave Dynamic Differential Logic.
Chapter 5 explores the effect of using process variations to increase security of cryptographic devices, concluding with results analysis and future directions.
2.1 Data Encryption Standard (DES)

This algorithm is a secure standard adopted by the Federal Information Processing Standards since 1977. DES is a block cipher, so it takes a fixed length of input along with a secret key and encrypts the input using a set of operations. The output is of the same fixed length as the input - in order to decrypt the output, the receiver should be aware of the secret key. Typically, the key size is 64 bits, 8 of which are used to check parity. The effective key size is 56 bits [20].

2.1.1 Structure

The DES algorithm is illustrated in Figure 2.1
DES comprises of 16 rounds of encryption and has a 64 bit block input. Every round uses a different key, provided by a key schedule. The algorithm splits the input plaintext, after initial permutation into two halves. In accordance with the Feistel scheme,
one half is combined with the key in the \( f \)-function, whereas the other half is combined with the output of the \( f \)-function. The Feistel scheme, eliminates the need for separate encryption and decryption algorithms, thereby making IC implementation simpler and cost-effective [20]. At the end of the 16 rounds, there is also a final permutation. In every round but the last, the halves of partly encrypted data, are swapped, before passing it on to the next round.

### The Feistel Function

The \( F \)-function operates on half a block (32 bits) at a time and consists of four stages:

(a) **Expansion**

The block denoted \( E \) in the diagram, expands the 32 bit input to 48 bits as follows. The existing 32 bits are replicated in sets of 4 bits. Each set of 4 bit is transformed to 6 bits by borrowing the immediately adjacent bits to the left and right of the set.

(b) **Key mixing**

The 48 bit subkey obtained from the key schedule per round is XORed together with the expanded input in the previous step.

(c) **Substitution**

The S-boxes transform the output from the previous step using specific look up tables. The six bit input to each S-box or substitution box is converted to 4 bits.

(d) **Permutation**

This scrambles the output of the S-boxes in preparation for the next round. The S-boxes in the next round do not have the same set of bits as input, as in the previous
The Feistel function is depicted in Figure 2.2

![The Feistel Function Block](image)

Figure 2.2: The Feistel Function Block [5]

**Key Schedule**

Initially, 56 bits of the key are selected from the initial 64 by Permutation Choice 1 (PC-1) the remaining eight bits are either discarded or used as parity check bits. The 56 bits are then divided into two 28-bit halves; each half is thereafter treated separately. In successive rounds, both halves are rotated left by one or two bits (specified for each round), and then 48 subkey bits are selected by Permutation Choice 2 (PC-2) 24 bits from the left half, and 24 from the right. The rotations mean that a different set of bits is used in each subkey; each bit is used in approximately 14 out of the 16 subkeys. [20]

The key schedule for decryption is similar the subkeys are in reverse order compared to encryption. Apart from that change, the process is the same as for encryption.
The same 28 bits are passed to all rotation boxes.

The key schedule is illustrated in Figure 2.3

![Figure 2.3: Overview of the DES Key Schedule][1]

2.1.2 DPA Attacks on DES

The left and right halves of the data input produced after initial permutation and in subsequent rounds are obtained as follows.

\[ L_i = R_{i-1} \]  \hspace{1cm} (2.1)

and

\[ R_i = L_{i-1} \otimes f(R_{i-1}, K_i) \]  \hspace{1cm} (2.2)
The attacker records power traces and cipher text for encryption of N plaintexts. He tries to determine the 48 bit key in parts of 6 bits.

Hence there can be 64 possible key guesses for the same. For each key guess the attacker does as follows:

1. calculates independently the value of any bit in L_{15}, say $B$.

2. based on the value of this bit (0 or 1), the set of N power traces is divided into two groups. The average power trace of each group is found.

3. calculates the difference of the two average traces.

The average power trace corresponding to the correct key will show a marked spike in value, since it represents the trace connected to the correct value of the selected bit for observation.

The differential traces are as illustrated in Figure 2.4 [3]
Technically speaking the five step DPA methodology described in Chapter 1 is a Correlation Power Analysis Attack, since the device power traces are correlated with the Hamming Distance/Hamming Weight model traces. This is the attack methodology [2] adopted in the course of this thesis. Additionally, the DES part under attack is a single S-BOX.

### 2.2 KeeLoq

KeeLoq is also a block cipher, of input size 32 bits and a fixed size, rotating key of 64 bits. KeeLoq uses a non-linear feedback shift register in its implementation and consists of 528 rounds of encryption. This algorithm is often used in today’s automobile security systems [7].
2.2.1 Structure

The hexadecimal function of the NLF block is as follows,

\[ F(a, b, c, d, e) = d \otimes e \otimes ac \otimes ae \otimes bc \otimes be \otimes cd \otimes de \otimes ade \otimes ace \otimes abd \otimes abc. \]  

\[ (2.3) \]

The KeeLoq encryption is illustrated in Figure 2.5

![Diagram of KEELOQ Encryption]

2.2.2 DPA Attacks on KeeLoq

Differential Power Analysis attacks on KEELOQ were first demonstrated in [21]. The traditional DPA attack method involving difference of means is described in this paper as well as a Correlation Power Attack using the Hamming Distance Model. The authors also highlight some real world issues faced during the attack.
As per the KeeLoq encryption algorithm the Hamming Distance and power consumption of the key register for a given key in each clock cycle is constant since the key is simply rotated [21]. Therefore the non constant exploitable data dependent part of the power is in the data register. This register $y$ becomes the focus of the attack.

The following Hamming Distance (HD) power model is used in this attack.

$$P_{Hyp}^i = HD(y^i, y^{i-1}) = HW(y^i \oplus y^{i-1})$$  \hspace{1cm} (2.4)

where the left hand side denoted the hypothetical power consumption in the $i_{th}$ round, HD and HW are the Hamming Distance and Hamming Weight respectively, $y_i$ indicates the content of the data register in the $i_{th}$ round and the XOR is a 32 bit function.

Assuming that we already know the ciphertext, 32 bits in the data register of the 528th round are known. This means that 31 bits of the 527th round are also known since the the bits in the 528th round are obtained by shifting the bits from the 527th round plus the new bit from the NLF block. Therefore the only unknown bit is is $y_0$ from the 527th round. This bit is obtained from XORing the output of NLF function, bits 16 and 31 of the data register, and bit 15 of the key register in the decryption process. Similarly $y_0$ corresponding to every round can be calculated.

However this method of recovering the key is very tedious, hence the authors of [21] propose a scalable DPA attack on KeeLoq, wherein they guess a few bits of the key at a time.

The principle behind this method is employed in the SCARF [1] mechanism used to attack the SPICE implementation of KeeLoq in this thesis.
Chapter 3

Effect of Temperature Variation on
Differential Power Analysis

MOSFET power consumption is affected by a number of factors. Some of these factors are temperature dependent and therefore cause variance of power with respect to temperature fluctuation.

We next examine each of the temperature dependent MOSFET parameters in detail: threshold voltage, carrier mobility, saturation velocity, parasitic drain source resistance [8] [22] [23] [24].

3.1 Temperature Dependent MOSFET Parameters

Fluctuation in the die temperature affects the device characteristics thereby altering the performance and power consumption of integrated circuits [8] [22] [25] [26] [23] [24]. Furthermore, the increase in the doping concentration and the enhanced electric fields with technology scaling tend to affect the rate of change of the device parameter variations when the temperature fluctuates [8] [9] [27] [28].
The concentration of the charge carriers in a semiconductor device increases with the temperature [29] [30]. This increase in the carrier concentration with temperature alters the behavior of semiconductor devices.

3.1.1 Threshold Voltage

The increase in the carrier concentration with temperature shifts the Fermi-level up towards the conduction band in a donor-doped semiconductor material [31]. Alternatively, in an acceptor-doped material, the increase in the hole concentration shifts the Fermi-level down towards the valence band when the temperature increases. The shift in the Fermi-level reduces the band-gap energy in a semiconductor material thereby reducing the threshold voltage of a MOSFET [24].

3.1.2 Carrier Mobility

The increase in the carrier concentration with temperature also increases the scattering of the majority charge carriers [8] [29]. The effective mobility of the charge carriers is limited by three scattering mechanisms: scattering with surface acoustic phonons, surface roughness scattering, and columbic scattering [8] [29]. Surface roughness and columbic scattering dominate the carrier mobility at very low temperatures. Alternatively, the fluctuation in the carrier mobility is primarily due to phonon scattering at elevated temperatures [8] [29]. The effective carrier mobility decreases when the temperature increases, as described in [29] and [28].

3.1.3 Saturation Velocity

Reduction in the effective mobility with temperature affects the saturation velocity of MOSFETs [8] [25] [26]. The saturation velocity is as described below [8]
\[ V_{\text{SAT}} = \mu_{\text{eff}} E_c \quad (3.1) \]

where \( \mu_{\text{eff}} \) and \( E_c \) are the effective carrier mobility and the electric field at which the carrier drift velocity saturates respectively. Although both saturation velocity and mobility have a negative temperature dependence, saturation velocity displays a relatively weaker dependence since \( E_c \) increases with the temperature \([8]\) \([24]\).

### 3.1.4 Parasitic Drain Source Resistance

Furthermore, as the MOSFET currents become higher while the supply voltages shrink, the drain/source series resistance becomes increasingly effective on the I-V characteristics of devices in scaled CMOS technologies \([8]\) \([24]\). The drain/source resistance increases approximately linearly with the temperature \([8]\) \([24]\).

### 3.2 Impact of Technology Scaling on Threshold Voltage and Carrier Mobility

Device and interconnect dimensions are scaled with each new technology generation to enhance the speed of CMOS integrated circuits \([8]\) \([32]\). Scaling the device dimensions strengthens the electric fields between device terminals. Furthermore, to counterbalance the short-channel effects, the channel doping concentration is typically increased (through halo-doping) in each new technology generation \([33]\). These changes in the device structure tend to alter the rate of change of the device parameters when the temperature fluctuates.

*Threshold voltage roll-off* \( \Delta V_t \) is defined as the difference between the long-channel and the short-channel MOSFET threshold voltages at a given temperature \([8]\) \([9]\). According to the simple onedimensional charge sharing model, the threshold voltage roll-off is
\[ \Delta V_t = \frac{-qN_A W_C W_{DS}}{C_{OX} L_{eff}} \]

where \( q \), \( N_A \), \( W_C \), \( W_{DS} \), \( L_{eff} \), and \( C_{OX} \) are the charge of an electron, channel doping concentration, channel depletion layer thickness, source/drain channel junction depletion layer thickness, effective channel length, and the gate-oxide capacitance per unit area, respectively.

The temperature dependence of threshold voltage in n-channel MOSFETs is shown in Fig 3.1 [8] [9]. As given by in the preceding equation, reducing the effective channel length and increasing the channel doping concentration effectively increases the threshold voltage roll-off with each new technology generation [8].

Furthermore, the threshold voltage roll-off is enhanced at lower temperatures due to the increase in the depletion layer thickness, as described in [9]. The gap between the long-channel and short-channel MOSFET threshold voltages therefore decreases as the
temperature increases as shown in Figure 3.1.

*Carrier mobility* is mainly affected by scattering effects. Changes in the electric field alter the scattering mechanism that dominates the mobility of the majority charge carriers [8] [29]. Coulomb scattering and phonon scattering dominate the carrier mobility at low electric fields [8] [28]. Alternatively, at high electric fields, the fluctuation in the carrier mobility is primarily due to the surface roughness scattering and the phonon scattering [8] [27]. The electric field produced in a MOSFET increases as the effective channel length is reduced [8] [34]. The increasing electric fields with technology scaling cause the carrier mobility to become essentially independent of the doping level, resulting in a universal behavior, as described in [8] [27] and [28]. This universal inversion layer mobility is affected uniformly by phonon scattering at elevated temperatures. The rate of change of carrier mobility with temperature fluctuations is therefore expected to be relatively insensitive to technology scaling.

### 3.2.1 Effect of Temperature Variation on Gate Overdrive and Drain Current

As discussed previously, temperature has been shown to have an effect on the dominant factors affecting drain current, namely carrier mobility and threshold voltage especially in scaled devices. Therefore temperature can also be shown to have an effect on the drain current which in turn *contributes to power and current variance*.

The supply and threshold voltage scaling with each new technology generation also alters the device characteristics when the temperature fluctuates. The supply voltage is scaled primarily based on the device reliability and target clock frequency requirements in a new technology generation. The speed of a circuit can be enhanced by scaling the threshold voltages. Due to the subthreshold leakage current constraints, however, the threshold
voltages are scaled at a much slower rate as compared to the supply voltage. The supply voltage to threshold voltage ratio is reduced with each new technology generation. The temperature fluctuation induced threshold voltage variation is therefore expected to have an increasingly important role in determining the MOSFET drain current variations when the temperature fluctuates, as described in [8] [22]. However, we must also consider the effects of carrier mobility on the drain current variation with respect to temperature.

The absolute values of threshold voltage, carrier mobility, and the saturation velocity degrade as the temperature is increased [8] [22] [25] [26]. The saturation velocity is typically a weak function of temperature [8] [24]. The degradation in carrier mobility tends to lower the drain current produced by a MOSFET [8] [25] [26]. Similarly, the increase in the drain/source resistance with temperature reduces the drain current [8] [25] [26]. Threshold voltage degradation with temperature, however, tends to enhance the drain current because of the increase in gate overdrive $\text{abs}(V_{GS} - V_t)$

The effective variation of MOSFET current is determined by the variation of the dominant device parameter when the temperature fluctuates [8] [22].

### 3.3 Effect of Temperature Variation on Ease of Power Analysis Attacks

In the preceding sections we examined all the MOSFET parameters that are affected by temperature variation. It was established that the effective current drive at any temperature is dependent on the dominant effect between threshold voltage scaling and carrier mobility decrease - they have opposite effects on the current drive. This leads us to hypothesize the following:

(a) Variation in circuit temperature can affect the variance in the power consumption
during cryptographic operations in the circuit.

(b) Threshold voltage of transistors in a CMOS circuit, decreases with increase in temperature, leading to an increase in the power variance. Carrier mobility of the transistors in CMOS circuit, also decreases with increase in temperature, leading to a decrease in the power variance. Therefore, the increase in variance of power due to threshold voltage scaling, can be countered by the decrease in variance of power due to carrier mobility decrease.

We established in Chapter 1, that the higher the power variance for a fixed cryptographic operation in time, the easier the attack in terms of number of traces. This, together with the above two hypotheses form the basis of our initial experiments in establishing the dependence of the number of traces metric on temperature variation, as also using high threshold voltage devices to decrease power variance.

We look at the dynamic component of the CMOS power consumption since that represents the switching transitions in the device. The experiments show that this component decreases with increase in temperature in keeping with our first hypothesis. Since carrier mobility is the parameter that causes a decrease in current with increase in temperature, the predominant factor affecting power variance due to temperature variation at the 90nm node is carrier mobility.

We show that due to the decrease in power variance with increase in temperature it becomes harder to attack a device based on the dynamic power consumption at higher temperatures. The algorithms considered in the course of these experiments are the aforementioned DES and KeeLoq algorithms.
3.4 Experimental Results

3.4.1 KeeLoq

Experimental Setup

The KeeLoq encryption algorithm (528 rounds) was implemented in SPICE using 90nm extracted RC standard cell netlists and model file provided in the Synopsys 90nm Generic Library [35].

35 32-bit input plaintexts were encrypted, each using five different random 64-bit keys. Power traces corresponding to each of these individual encryptions were recorded at four different temperatures viz 10, 25, 75 and 125 degrees Celsius. Relevant points of the power traces were extracted using Perl scripts and fed as input along with the recorded output ciphertext to SCARF [1], for performing an attack. The average variance across five keys for each temperature and the average number of traces across five keys for each temperature were recorded along with the correlation values corresponding to the correct key.

The total simulation, processing and attack time for the above was 10 days and the memory required was 30GB at a transient simulation resolution (step time) of 0.025ns in Synopsys HSPICE.

```
.tran 0.025n 6364.8n
```

The commands for transient simulation can be found in [36].

The temperature was varied using the .TEMP [36] statement as follows:

```
.TEMP = 10 25 75 125
```

Power measurement was done using the POWER statement in Synopsys HSPICE

```
.print tran POWER
```
Average Variance in Power Values With Temperature

The average power variance over the period of encryption across all five random keys is shown to be decreasing with temperature as per Figure 3.2

![Figure 3.2: KEELOQ: Average power variance trace with respect to temperature over five keys](image)

Number of Traces Required to Recover the Correct Key per Temperature

The number of traces, across all five random keys is shown to be increasing with temperature as per Table 3.1. On an average one can elicit the key with around 2 traces at a temperature of 10 degrees whereas with about 7 traces at a temperature of 125 degrees. This is due to the decrease in variance of power at a temperature of 125 degrees. The average trace curve is shown in Figure 3.3

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Key 1</th>
<th>Key 2</th>
<th>Key 3</th>
<th>Key 4</th>
<th>Key 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>25°C</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>75°C</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>125°C</td>
<td>5</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 3.1: KEELOQ: Number of traces required to recover the correct key
Chapter 3: Effect of Temperature Variation on Differential Power Analysis

Figure 3.3: KEELOQ: Average number of traces increases with respect to temperature over five keys

Correlation

The correlation of the power traces to the Hamming Distance model (for the correct key) used in SCARF [1] decreases with increase in temperature as shown in Table 3.2 and as per the average correlation graph over five keys with respect to temperature, shown in Figure 3.4

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Key 1</th>
<th>Key 2</th>
<th>Key 3</th>
<th>Key 4</th>
<th>Key 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>0.99</td>
<td>0.9</td>
<td>0.96</td>
<td>0.97</td>
<td>0.93</td>
</tr>
<tr>
<td>25°C</td>
<td>0.99</td>
<td>0.93</td>
<td>0.89</td>
<td>1</td>
<td>0.93</td>
</tr>
<tr>
<td>75°C</td>
<td>0.88</td>
<td>0.77</td>
<td>0.91</td>
<td>0.82</td>
<td>0.84</td>
</tr>
<tr>
<td>125°C</td>
<td>0.84</td>
<td>0.91</td>
<td>0.72</td>
<td>0.78</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Table 3.2: KEELOQ: Correlation vs Temperature
3.4.2 DES

A single S-BOX of DES has been implemented with 10 bit input plaintext and 6-bit key. The implementation and attack interface is as per [2]. This flow can generate any number of input vectors, record power traces and perform an attack all in a single makefile based flow. The simulator used is Synopsys Nanosim ®. The full description of this interface and guidelines for using it, are provided in Appendix A.

100 10-bit input vectors are used for encryption using five different random 6-bit keys and the corresponding variance in power is measured for every temperature. The average number of traces traces across all keys, required to recover the key at every temperature is measured as before.

Total simulation time was 2 days and the memory required was about 30 MB. The transient simulation resolution is 0.01 ns. The netlist and model files used are the standard cell netlists and standard threshold voltage model file provided under the 90nm Synopsys Generic Library [35] ®.
Average Variance in Power Values With Temperature

The average power variance over the period of encryption across all five random keys is shown to be decreasing with temperature as per Figure 3.5.

![Figure 3.5: DES: Average power variance trace with respect to temperature over five keys](image)

**Number of Traces Required to Recover the Correct Key per Temperature**

The number of traces, across all five random keys is shown to be increasing with temperature as per Table 3.3. On an average one can elicit the key with around 28 traces at a temperature of 10 degrees whereas with about 70 traces at a temperature of 125 degrees. This is due to the decrease in variance of power at a temperature of 125 degrees. The plot of average traces versus temperature is shown in Figure 3.6.
Chapter 3: Effect of Temperature Variation on Differential Power Analysis

Figure 3.6: DES: Average number of traces increases with temperature

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Average Number of Traces Across Five Keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>28</td>
</tr>
<tr>
<td>25°C</td>
<td>52</td>
</tr>
<tr>
<td>75°C</td>
<td>58</td>
</tr>
<tr>
<td>125°C</td>
<td>70</td>
</tr>
</tbody>
</table>

Table 3.3: DES: Average number of traces required to recover the correct key

Correlation

The average correlation graph with respect to temperature is shown in Figure 3.7 and the values of correlation are shown in Table 3.4.
Chapter 3: Effect of Temperature Variation on Differential Power Analysis

3.5 Conclusion

As we have seen from the experiments above, irrespective of the encryption algorithm, increase in temperature causes a decrease in the ease of power attack, since the variance in power decreases with increase in temperature. Due to decrease in the power variance we also see a decrease in correlation between the power traces and the Hamming Distance trace, with temperature. The MOSFET parameter that causes decrease in current/power variance with increase in temperature is the carrier mobility. Carrier mobility is higher at lower temperatures, which implies higher drain current variance. In order to reduce the drain current variance, therefore we propose the use of high threshold voltage.
Cells to increase the security of the devices against temperature variance.
Chapter 4

Using High Threshold Voltage

Cells in Cryptographic Circuits

In Chapter 3 we examined and established the inverse temperature dependence of dynamic power analysis attacks. It becomes necessary, therefore to explore if we can increase resistance of cryptographic circuits against power analysis due to temperature variations.

As previously discussed in Chapter 1, Wave Dynamic Differential Logic (WDDL) [16] is a circuit level counter measure against power analysis attacks in general, that has been proposed in recent times. This circuit technique achieves the objective of minimizing the exploitable power variance, thereby increasing security against power analysis attacks.

An example WDDL NAND gate is shown in Figure 4.1. It consists of four inputs, $a, b, \bar{a}$ and $\bar{b}$ and two outputs $q, \bar{q}$. As per De-Morgan’s Laws discussed in Chapter 1, the positive AND and OR gates shown in the figure together implement differential outputs of the NAND gate. When the inputs $a, b, \bar{a}$ and $\bar{b}$ are set to precharge values, the outputs $q, \bar{q}$ become 0. When the inputs are set to complementary values, the outputs become differential or complementary.
Sequential WDDL cells precharge and evaluate with every alternate clock cycle. An example WDDL flip flop consists of four D flip flops organized into two stages as shown in Figure 4.2. When stage 1 is in the precharge phase, stage 2 is in the evaluate phase i.e. output is available.
Figure 4.2: WDDL flip flop cell schematic diagram [4]

Figure 4.3 shows how WDDL cells can be used to build an entire circuit containing both combinational and sequential elements. The flip flops in the register provide the precharge and evaluate inputs every alternate clock cycle to the combinational cells, which in turn propagate the same values back to the registers in a cyclic fashion. This achieves the objective of the circuit going into precharge mode every alternate clock cycle thereby causing only one transition per flip flop in the data register.
4.1 Experiments With High Threshold Voltage Cells

It has been known that using high $V_{th}$ cells can reduce the current drive and therefore the power consumption of CMOS devices, since the gate overdrive responsible for current $V_{GS}-V_{th}$ decreases with increase in threshold voltage. This comes at a tradeoff of increase in delay. Since current drive is higher at low temperatures due to high carrier mobility we propose the use of high $V_{th}$ cells to reduce current drive. As before the netlists used for all the experiments are 90nm standard cell netlists and model files provided in [35]. DES and KeeLoq are the experimentation platforms for the same.

We first examine the effect on the number of traces metric of using High Threshold Voltage cells and then examine the combined effect on the number of traces of using WDDL cells with high threshold voltage cells. It is found that replacing the NLF combinational block of the Keeloq algorithm with WDDL cells along with usage of high $V_{th}$ causes the greatest increase in the number of traces and the greatest decrease in power variance. With respect to DES, while the number of traces metric increases significantly due to use of high $V_{th}$ cells, it becomes impossible to attack the device even with 300000 traces when WDDL logic is used in both the flip flops (which are the main source of exploitable power
consumption) and the combinational blocks. The correlation obtained in this case is less than 0.2. Hence WDDL logic if used both in the sequential and combinational blocks is sufficient to secure a cryptographic circuit against ease of DPA due to temperature variation.

The WDDL AND and WDDL XOR gates used for experimentation are as per layout extracted netlists in [6].

4.1.1 KeeLoq

35 input patterns are encrypted using five different random keys and the power variance across all recorded power traces is recorded in each encryption round for four different temperatures 10, 25, 75 and 125 degrees Celsius.

Average Power Variance Using High $V_{th}$ Cells

Figure 4.4 shows that the average variance in power over five keys, decreases with respect to temperature when using high threshold voltage cells.

![Figure 4.4: KEELOQ: Average power variance vs temperature using high threshold voltage cells](image)

Table 4.1 shows that number of traces required to recover the key is highest on average at 125 degrees celsius. Figure 4.5 reflects the same trend of increase in number of
traces required with temperature, as seen in the case of using standard threshold voltage cells. However, for every temperature, the number of traces required when using high threshold voltage cells is higher than when using standard threshold voltage cells.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Key 1</th>
<th>Key 2</th>
<th>Key 3</th>
<th>Key 4</th>
<th>Key 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>25°C</td>
<td>13</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>75°C</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>125°C</td>
<td>35</td>
<td>15</td>
<td>15</td>
<td>25</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 4.1: KEELOQ: Number of traces required to recover the correct key (high Vth)

Figure 4.5: KEELOQ: Average number of traces increases with respect to temperature over five keys using High Threshold Voltage Cells

Correlation

Table 4.2 and Figure 4.6 show that the average correlation between the power traces and the Hamming Distance model decreases across all keys, with increase in temperature.
Table 4.2: KEELOQ: Correlation vs temperature (high Vth)

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Key 1</th>
<th>Key 2</th>
<th>Key 3</th>
<th>Key 4</th>
<th>Key 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>0.95</td>
<td>0.93</td>
<td>0.9</td>
<td>0.85</td>
<td>0.89</td>
</tr>
<tr>
<td>25°C</td>
<td>0.89</td>
<td>0.74</td>
<td>0.76</td>
<td>0.8</td>
<td>0.82</td>
</tr>
<tr>
<td>75°C</td>
<td>0.94</td>
<td>0.69</td>
<td>0.61</td>
<td>0.84</td>
<td>0.73</td>
</tr>
<tr>
<td>125°C</td>
<td>0.5</td>
<td>0.51</td>
<td>0.6</td>
<td>0.59</td>
<td>0.48</td>
</tr>
</tbody>
</table>

Figure 4.6: KEELOQ: Average correlation with respect to temperature over five keys using High Threshold Voltage Cells

Comparison of Standard Threshold Voltage and High Threshold Voltage Cases

Table 4.3 shows the different values of threshold voltage used in the experiments [35].

<table>
<thead>
<tr>
<th>Transistor Type</th>
<th>Std $V_{th}$</th>
<th>26% High $V_{th}$</th>
<th>50% High $V_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMOS</td>
<td>0.397V</td>
<td>0.5022V</td>
<td>0.5955V</td>
</tr>
<tr>
<td>NMOS</td>
<td>-0.276V</td>
<td>-0.34776V</td>
<td>-0.414V</td>
</tr>
</tbody>
</table>

Table 4.3: Standard and high threshold voltage values

Average Number of Traces

Table 4.4 shows the comparison between the average number of traces across five
keys, required at each temperature when using cells with different values of $V_{th}$. One can clearly see that using higher threshold voltage cells causes an increase in the number of traces required at every temperature.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std $V_{th}$</th>
<th>26% High $V_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>25°C</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>75°C</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>125°C</td>
<td>6</td>
<td>22</td>
</tr>
</tbody>
</table>

Table 4.4: KEELOQ: Number of traces required vs temperature, with std $V_{th}$ and 26% high $V_{th}$

Correlation

Table 4.5 shows the comparison between the average correlation across five keys, required at each temperature when using cells with different values of $V_{th}$. On an average the correlation obtained when using high threshold voltage cells is lower than when using standard threshold voltage cells.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std $V_{th}$</th>
<th>26% High $V_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>0.95</td>
<td>0.904</td>
</tr>
<tr>
<td>25°C</td>
<td>0.948</td>
<td>0.802</td>
</tr>
<tr>
<td>75°C</td>
<td>0.844</td>
<td>0.762</td>
</tr>
<tr>
<td>125°C</td>
<td>0.84</td>
<td>0.536</td>
</tr>
</tbody>
</table>

Table 4.5: KEELOQ: Correlation vs Temperature (Std $V_{th}$ vs 26% High $V_{th}$)

Different Values of Threshold Voltage

The above experiments show that using High Threshold Voltage cells does make a difference to the number of traces metric. However we need to examine by how much, hence we now perform a comparative analysis to determine the number of traces metric when using standard, 26% high, and 50% high threshold voltages respectively. These experiments are performed on the hardest key to recover namely key 1.
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

Number of Traces

Table 4.6 shows the comparison between the average number of traces across five keys, required at each temperature when using cells with different values of $V_{th}$, while Table 4.7 shows the percentage increase in the number of traces metric when using 50% higher $V_{th}$ devices.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std $V_{th}$</th>
<th>26% High $V_{th}$</th>
<th>50% $V_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>3</td>
<td>4</td>
<td>18</td>
</tr>
<tr>
<td>25°C</td>
<td>13</td>
<td>13</td>
<td>18</td>
</tr>
<tr>
<td>75°C</td>
<td>9</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>125°C</td>
<td>20</td>
<td>30</td>
<td>35</td>
</tr>
</tbody>
</table>

Table 4.6: KEELOQ: Comparing number of traces required with different values of $V_{th}$

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std $V_{th}$</th>
<th>50% $V_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>3</td>
<td>18</td>
</tr>
<tr>
<td>25°C</td>
<td>13</td>
<td>18</td>
</tr>
<tr>
<td>75°C</td>
<td>9</td>
<td>20</td>
</tr>
<tr>
<td>125°C</td>
<td>20</td>
<td>35</td>
</tr>
</tbody>
</table>

Table 4.7: KEELOQ: Increase in number of device traces needed when using 50% High $V_{th}$ cells

Using High $V_{th}$ in the KeeLoq NLF Combinational Block vs in the Flip Flops

Number of Traces and Correlation Comparison

Table 4.8 shows the comparison between the average number of traces and correlation for key1, at each temperature when using cells high threshold voltage cells in the NLF combinational block versus in the flip flops of the data register. This experiment is performed to determine if it enough to just replace the combinational block with high threshold voltage cells. It is found that replacing the part of the circuit where most of the data dependent power consumption takes place provides better protection and lower power variance.
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

<table>
<thead>
<tr>
<th>Temperature</th>
<th>High $V_{th}$ in DFFs (Number of Traces)</th>
<th>High $V_{th}$ in DFFs (Correlation)</th>
<th>High $V_{th}$ in NLF (Number of Traces)</th>
<th>High $V_{th}$ in NLF (Correlation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>3</td>
<td>0.97</td>
<td>2</td>
<td>0.99</td>
</tr>
<tr>
<td>25</td>
<td>3</td>
<td>0.97</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>75</td>
<td>4</td>
<td>0.97</td>
<td>4</td>
<td>0.97</td>
</tr>
<tr>
<td>125</td>
<td>20</td>
<td>0.54</td>
<td>7</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Table 4.8: KEELOQ: Using High $V_{th}$ in the KeeLoq NLF Combinational Block vs in the Flip Flops

**Comparison of Amount of Effective Power Variance within a Trace**

The average power variance trace spanning 528 encryption rounds is calculated across five keys for each temperature. The average of the variance values across the 528 encryption rounds, in the average traces, obtained for each temperature is plotted in Figure 4.7. Average power variance is much lesser when DFF cells are replaced with High $V_{th}$ DFF cells as compared to replacing the NLF block with High $V_{th}$ cells.

![Figure 4.7: KEELOQ: comparison of effective power variance within average power variance trace - DFF(high $V_{th}$) vs NLF(high $V_{th}$)](image-url)
4.1.2 DES

100 10-bit input patterns are encrypted in a single S-BOX of DES using five different 6-bit random keys and the power variance across all recorded power traces is recorded in each encryption round for four different temperatures 10, 25, 75 and 125 degrees Celsius. The average value across each average variance trace is also calculated per temperature.

Average Power Variance Trace Comparison

Figure 4.8 and Figure 4.9 show the average power variance traces for the standard and high threshold voltage cases.

Figure 4.8: DES: Average Power variance (Standard $V_{th}$)
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

Figure 4.9: DES: Average power variance (High $V_{th}$)

Figure 4.10 (representing average value across average variance trace for each temperature) shows that as in the case of KeeLoq, power variance is much lesser when High $V_{th}$ cells are used.

Figure 4.10: DES: Comparison of effective power variance within average power variance trace - std $V_{th}$ vs 50 % high $V_{th}$
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

Table 4.9 shows the percentage decrease in variance.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std V_{th} Power Variance</th>
<th>High V_{th} Power Variance</th>
<th>% Decrease in Power Variance when using 50% High V_{th} cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>1.58691E-08 W^2</td>
<td>8.30977E-09 W^2</td>
<td>48%</td>
</tr>
<tr>
<td>25°C</td>
<td>1.37864E-08 W^2</td>
<td>7.19225E-09 W^2</td>
<td>48%</td>
</tr>
<tr>
<td>75°C</td>
<td>8.42751E-09 W^2</td>
<td>4.66241E-09 W^2</td>
<td>45%</td>
</tr>
<tr>
<td>125°C</td>
<td>6.589E-09 W^2</td>
<td>3.08071E-09 W^2</td>
<td>53%</td>
</tr>
</tbody>
</table>

Table 4.9: DES: % Decrease in effective power variance within average power variance trace when using 50% high V_{th} cells

Average Number of Traces Comparison

Table 4.10 shows the percentage increase in number of traces when using High V_{th} cells.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std V_{th} (Number of Traces)</th>
<th>High V_{th} (Number of Traces)</th>
<th>% Decrease in Variance when using 50% High V_{th} cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>28</td>
<td>36</td>
<td>29%</td>
</tr>
<tr>
<td>25°C</td>
<td>52</td>
<td>62</td>
<td>19%</td>
</tr>
<tr>
<td>75°C</td>
<td>58</td>
<td>74</td>
<td>28%</td>
</tr>
<tr>
<td>125°C</td>
<td>70</td>
<td>92</td>
<td>31%</td>
</tr>
</tbody>
</table>

Table 4.10: % Increase in number of traces when using 50% high V_{th} cells

4.2 Experiments with Wave Dynamic Differential Logic

We have examined how High Threshold Voltage cells can increase security of cryptographic devices. Now, we proceed to examine the combined effect of using Wave Dynamic Differential Logic and High Threshold Voltage cells. For Wave Dynamic Differential Logic to have a considerable effect of practically eliminating the exploitable power consumption...
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

information, the circuit must carry differential outputs in the sequential and combinational parts. However we show that using WDDL logic in the combination block along with high threshold voltage cells can increase the security to a greater degree compared to using only High $V_{th}$ cells. We also show that using WDDL logic in both the combinational and sequential parts of the circuit eliminates the possibility of an attack even with 300000 traces.

4.2.1 WDDL Experiments with KeeLoq

As before, 35 input traces are encrypted; power variance and number of traces are recorded considering five random keys, and plotted against temperature. The standard cell netlists and model files are as provided in [35]. We also look at comparative analysis with respect to power variance and number of traces, of four cases namely

(a) Using only Std $V_{th}$ cells

(b) Using only High $V_{th}$ cells.

(c) Using WDDL cells in the NLF combinational block of the Std $V_{th}$ KeeLoq implementation.

(d) Using WDDL cells in the NLF combinational block of the 50% High $V_{th}$ KeeLoq implementation.

Using WDDL Cells in the NLF block - Std $V_{th}$ vs High $V_{th}$

Number of Traces Comparison

Table 4.11 shows the increase in number of traces when using WDDL cells (only in the combinational block of KeeLoq) in conjunction with high $V_{th}$ cells (in full design including the NLF block). Figure 4.11 shows the graphical representation of the increase in the number of traces metric.
**Temperature** | **Std $V_{th}$** | **50% high $V_{th}$** | **Increase in Number of Traces**  
---|---|---|---
10°C | 2 | 13 | 6.5X  
25°C | 3 | 23 | 7.6X  
75°C | 4 | 24 | 6X  
125°C | 5 | 30 | 6X  

Table 4.11: KEELQ(WDDL): Comparison of number of traces-high $V_{th}$ vs std $V_{th}$

![Figure 4.11: KEELQ(WDDL): Number of traces comparison-std $V_{th}$ vs high $V_{th}$](image)

**Correlation Comparison**

Table 4.12 shows the decrease in correlation when using WDDL cells (only in the combinational block of KeeLoq) in conjunction with high $V_{th}$ cells (in full design including the NLF block). Figure 4.12 shows the graphical representation of the decrease in correlation.
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Std $V_{th}$</th>
<th>50% high $V_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>0.938</td>
<td>0.782</td>
</tr>
<tr>
<td>25°C</td>
<td>0.9</td>
<td>0.588</td>
</tr>
<tr>
<td>75°C</td>
<td>0.872</td>
<td>0.516</td>
</tr>
<tr>
<td>125°C</td>
<td>0.87</td>
<td>0.462</td>
</tr>
</tbody>
</table>

Table 4.12: KEELOQ(WDDL): Correlation comparison -std $V_{th}$ vs high $V_{th}$

Comparison of Effective Power Variance within Average Power Variance Trace

Figure 4.13 shows the average power variance comparison when using WDDL cells (only in the combinational block of KeeLoq) in conjunction with high $V_{th}$ cells (in full design including the NLF block), as opposed to WDDL(NLF)/std $V_{th}$ implementation. Table 4.13 shows the decrease in power variance for the same case.
Figure 4.13: KEELOQ(WDDL): comparison of effective power variance within average power variance trace-std $V_{th}$ vs high $V_{th}$

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Average Power Variance - Std $V_{th}$</th>
<th>Average Power Variance - 50% high $V_{th}$</th>
<th>%Decrease in Power Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>5.92078E-07 W²</td>
<td>3.57518E-08 W²</td>
<td>94%</td>
</tr>
<tr>
<td>25°C</td>
<td>4.28387E-07 W²</td>
<td>6.98187E-08 W²</td>
<td>98%</td>
</tr>
<tr>
<td>75°C</td>
<td>3.41787E-07 W²</td>
<td>1.89396E-08 W²</td>
<td>99%</td>
</tr>
<tr>
<td>125°C</td>
<td>2.32908E-07 W²</td>
<td>3.06607E-08 W²</td>
<td>99%</td>
</tr>
</tbody>
</table>

Table 4.13: KEELOQ(WDDL): comparison of effective power variance within average power variance trace-std $V_{th}$ vs high $V_{th}$

Using High $V_{th}$ cells in the design-with and without WDDL cells in the NLF block

Comparison of Effective Power Variance within Average Power Variance trace

Figure 4.14 shows the graphical representation of the clear decrease in variance when WDDL cells are used in the NLF, in the presence of High $V_{th}$ cells. Table 4.14 shows the amount in percentage of the decrease in variance. This decrease is lesser than
the decrease obtained when comparing with WDDL cells in the NLF along with Std $V_{th}$ cells in the full design which shows that using High $V_{th}$ cells potentially adds an additional layer of security. This point will become more clear later when we compare the variance and number of traces metric of the four cases mentioned in the introduction of this chapter.

![Figure 4.14: KEELOQ(High $V_{th}$): comparison of effective power variance within average power variance trace with and without WDDL cells](image)

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Average Power Variance- non-WDDL(NLF Block)</th>
<th>Average Power Variance- WDDL(NLF Block)</th>
<th>%Decrease in Power Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>10°C</td>
<td>2.13E-07 $W^2$</td>
<td>3.57518E-08 $W^2$</td>
<td>83%</td>
</tr>
<tr>
<td>25°C</td>
<td>4.6E-08 $W^2$</td>
<td>6.98187E-08 $W^2$</td>
<td>85%</td>
</tr>
<tr>
<td>75°C</td>
<td>1.01E-08 $W^2$</td>
<td>1.89396E-08 $W^2$</td>
<td>81%</td>
</tr>
<tr>
<td>125°C</td>
<td>1.66E-08 $W^2$</td>
<td>3.06607E-08 $W^2$</td>
<td>82%</td>
</tr>
</tbody>
</table>

Table 4.14: KEELOQ(High $V_{th}$): comparison of effective power variance within average power variance trace with and without WDDL cells

From the above results it is clear that using WDDL cells along with High $V_{th}$ cells can cause an decrease in effective variance of upto 99% and increase in security by about 6 times from the nominal case where only Std $V_{th}$ cells are used without any secure logic.
Comparison of All Cases

Figure 4.15 shows the average power variance comparison for all the four combination i.e. WDDL(NLF)/non-WDDL(NLF) with Std $V_{th}$ cells/High $V_{th}$ cells respectively.

It can be seen from the graph that the power variance is least for the case where both WDDL cells and High $V_{th}$ cells are used. Figure 4.16 shows that the number of traces required is the greatest in the case where the power variance is the least namely, when use of high threshold voltage is combined with use of secure logic.

![Figure 4.15: KEELOQ: Overall comparison of effective power variance within average power variance trace](image)
Chapter 4: Using High Threshold Voltage Cells in Cryptographic Circuits

4.2.2 WDDL Experiments with DES

WDDL logic was used in both the sequential and combinational parts of the DES circuit. The setting PRECHARGE=1 as per the DPA Attack Methodology for DES in [2] converts the complete DES S-box implementation into a WDDL implementation by introducing necessary complementary logic, removing inverters and modifying the netlists. It must be kept in mind that when WDDL is implemented throughout a circuit then there is no need for inverters as inversion is achieved free of cost in WDDL [4].

It was found that irrespective of the value of threshold voltage used, it was impossible to recover the key even with 300 power measurements. Hence WDDL logic if used throughout the circuit can prove to be a sufficient measure of protection against temperature variation based attacks as it eliminates the exploitable power consumption i.e. $\text{Var}(P_{exp})=0$. 

Figure 4.16: KEELOQ: Overall average traces comparison
Effective Power Variance within Average Power Variance Trace

Table 4.15 shows the difference between the average power variance with WDDL logic and without WDDL logic.

<table>
<thead>
<tr>
<th>Effective Power Variance with WDDL(Std V_{th})</th>
<th>Effective Power Variance Without WDDL(Std V_{th})</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.61E-08 W^2</td>
<td>1.59E-08 W^2</td>
<td>2%</td>
</tr>
</tbody>
</table>

Table 4.15: DES(WDDL): effective power variance within average power variance trace with WDDL circuitry

Correlation

Figure 4.17 shows the average change in correlation across all five keys over 300 input vectors. The correlation decreases with increase in the number of vectors or device traces considered.

Figure 4.17: DES(WDDL): Plot of change in correlation with number of vectors
4.3 Conclusion

As we have seen in the preceding experimental results, using high threshold voltage cells in cryptographic circuits increase the security of the device by about 30% at least and 6-7 times at most when the high $V_{th}$ approach is combined with using secure logic, as per the number of traces metric. The high $V_{th}$ method also succeeds in decreasing the exploitable power variance by almost 100%.
Chapter 5

Consideration of Process Variations

The effect of technology scaling on process parameters due to lithography inaccuracies, is well established. Variability in process parameters such as transistor channel length $L$, gate width $W$, threshold voltage $V_{th}$, gate oxide thickness $t_{ox}$ and channel doping concentration $N_{ch}$ directly affect performance and power of integrated chips. Therefore, we proceed to examine the effects of process variability on the dynamic power variance and security of cryptographic devices. The parameters considered for variation are transistor channel length $L$, gate width $W$, threshold voltage $V_{th}$.

It has already been established that it is easiest to perform a power attack at $10^\circ C$. We examine how security of cryptographic devices changes when random parameter variations are added to the circuit at this temperature. The variability values for the process parameters have been taken from the ITRS [11] and are as per Table 5.1. The mean values are taken from the model files used for simulation [35]. As before, the high threshold voltage value is 50% higher than the nominal value.
Table 5.1: ITRS Variability Values [11]

<table>
<thead>
<tr>
<th>Process Parameter</th>
<th>Mean</th>
<th>3σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channel Length(L)</td>
<td>0.1µ (2λ)</td>
<td>12%</td>
</tr>
<tr>
<td>Gate Width(W)</td>
<td>0.4µ (7λ)</td>
<td>12%</td>
</tr>
<tr>
<td>PMOS Standard Threshold Voltage(V tp)</td>
<td>-0.276V</td>
<td>58%</td>
</tr>
<tr>
<td>NMOS Standard Threshold Voltage(V tn)</td>
<td>0.397V</td>
<td>58%</td>
</tr>
<tr>
<td>PMOS High Threshold Voltage</td>
<td>-0.414V</td>
<td>58%</td>
</tr>
<tr>
<td>NMOS High Threshold Voltage</td>
<td>0.5955V</td>
<td>58%</td>
</tr>
</tbody>
</table>

5.1 Experimental Setup

The algorithm under consideration is DES and the simulator used is Synopsys Nanosim ®. Usually, experiments to determine effects of process variability are done in Synopsys HSPICE ®, using the Monte Carlo simulation interface to generate random samples of process parameters, according to a normal distribution with a mean and standard deviation. However Synopsys Nanosim ® does not have a Monte Carlo interface, hence we utilize an efficient, stratified random sampling technique called Latin Hypercube Sampling [37] to generate the parameter values. As per [10] the minimum number of samples needed with this method is as low as 50.

In keeping with the findings in [10], 50 different instances of the base DES netlist used in the DPA Attack Flow [2] are generated using Perl scripts. The transistors in each of these netlists have random values generated using the NORMINV function in Microsoft Excel, for L, W, V tn, and V tp as per normal distributions specified by Table 5.1. The NORMINV function is the equivalent of the GAUSS [36] function in Synopsys HSPICE ® and is defined as

\[ \text{NORMINV}(\text{probability}, \text{mean}, \text{standard deviation}) \]
For the same set of 100 vectors used in the previous experiments, and five keys, the average power variance and ease of attack in terms of number of device traces, are calculated as before. A comparison of best, worst and average case attacks and power variance is also presented for both the standard and high threshold voltage mean values.

5.1.1 Latin Hypercube Sampling

In order to sample a normal distribution on a random variable $X$ evenly, the number of instances of $X$ need to be anywhere between 500 and 1000. To conduct SPICE simulations with so many samples can be prohibitive in terms of time, especially if we are dealing with multiple random variables. Latin Hypercube sampling enables us to achieve the objective of evenly sampling a normal distribution with the fewest samples.

The probability space of every random variable $X_i$ is divided into $N$ equal intervals where $N$ is the number of samples, as shown in Figure 5.1. For 2 random variables, the complete probability space is a square grid with $N$ rows and $N$ columns; for 3 random variables it is a cube; for 4 variables it is a hypercube, and so on.

![Figure 5.1: Density function of variable $X_i$ divided into $N$ non-overlapping intervals [10]](image)
Latin Hypercube

A square grid representing probability space of two random variables together, containing sample positions is a Latin square, if and only if, there is only one sample taken from each row and column, as shown in Figure 5.2 i.e. the row and column belonging to a particular sample, are not considered again for future sampling. In this way the entire probability space of all the variables is efficiently covered. A Latin hypercube is the generalisation of this concept to an arbitrary number of dimensions, whereby each sample is the only one in each axis-aligned hyperplane containing it.

![Figure 5.2: Pairing X1 with X2 when N=5](image)

In this case, we have four random variables $L$, $W$, $V_{tn}$ and $V_{tp}$. Latin hypercube sampling of these variables in each generated instance of the base DES netlist, is conducted as follows:

1. Values of $L$, $W$, $V_{tn}$ and $V_{tp}$ are simultaneously generated randomly from within a randomly picked unit of the Latin Hypercube formed by the intersection of probability spaces of these variables.

2. The hyperplanes containing that unit are not picked again for future sampling.
5.2 Experimental Results

5.2.1 Standard Threshold Voltage

Effective Power Variance across the Average Power Variance Trace

The effective variance, across the average power variance trace obtained, considering process variation, during encryption, across five random keys and 100 vectors, is 11% less than that obtained without process variation as shown in Figure 5.3.

![Figure 5.3: Effective power variance across average power variance trace, with process variation is 11% lesser than without process variation](image)

Number of Traces

The average number of traces required to recover the key, with process variations is 50% more than without process variation as shown in Figure 5.4. This shows that adding process variations at a lower temperature makes a device more secure by decreasing exploitable power variance.
Average, Best and Worst Case Comparison

Effective Power Variance and Effective Standard Deviation

Figure 5.5 shows the worst case (highest), best case (lowest) and average case effective power variance across the average power variance trace with process variations. The highest effective power variance case was the easiest to attack in terms of number of traces. Figure 5.6 shows the effective standard deviation of power that shows a similar trend as the power variance.
Figure 5.5: Average, best and worst case effective power variance across average power variance trace, comparison

Figure 5.6: Average, best and worst case effective standard deviation across average standard deviation trace, comparison

Number of Traces

The average best and worst case number of traces required, with process variations
are corresponding to the power variance in each of the cases described previously and are shown in Figure 5.7.

![Figure 5.7: Average, best and worst case number of traces comparison](image)

5.2.2 High Threshold Voltage

Effective Power Variance across the Average Power Variance Trace

The effective power variance across average power variance trace, with process variation during encryption, across five random keys and 100 vectors, is 22% less than that obtained without process variation as shown in Figure 5.8.
Figure 5.8: Effective power variance across average power variance trace, with process variation is 22% lesser than without process variation

Number of Traces

The average number of traces required to recover the key, with process variations is 8% more than without process variation as shown in Figure 5.9. This shows that adding process variations at a lower temperature makes a device more secure by decreasing exploitable power variance.
Figure 5.9: Number of traces with process variation is 8% more than without process variation

Average, Best and Worst Case Comparison

Effective Power Variance and Effective Standard Deviation

Figure 5.10 shows the worst case (highest), best case (lowest) and average case power variance with process variations. The highest power variance case was the easiest to attack in terms of number of traces. Figure 5.11 shows the standard deviation of power that shows a similar trend as the power variance.
Chapter 5: Consideration of Process Variations

Figure 5.10: Average, best and worst case effective power variance across average power variance case comparison

Figure 5.11: Average, best and worst case effective standard deviation across average standard deviation trace comparison

Number of Traces

The average, best and worst case number of traces required, with process variations
are corresponding to the power variance in each of the cases described previously and are shown in Figure 5.12.

Figure 5.12: Average, best and worst case number of traces comparison
5.3 Conclusion and Future Directions

We have explored and established in the course of this thesis, the inverse temperature dependence of security against dynamic power based DPA attacks, of cryptographic circuits and shown that using High Threshold Voltage in the design can decrease exploitable data dependent power variance by about 99% and increase the security by about 6 times. Provided the die temperature is kept constant throughout it is quite easy to attack a cryptographic implementation at room temperature which is the ideal attack scenario. The main factor affecting ease of attack is the exploitable power variance. As discussed in Chapter 1 and Chapter 4, any security countermeasure has at least the objective of minimizing or eliminating the power variance. Existing countermeasures involve using dynamic and differential logic. This thesis has examined the effect of using High Threshold Voltage cells, which has a direct bearing on the power variance and therefore the number of traces metric. Security can be further enhanced by combining this approach with using dynamic and differential logic. The addition of process variations is also found to increase security by 50%, especially at lower temperatures that facilitate ease of power attacks.

This thesis focuses on attacks based on the exploitable dynamic power consumption. In sub-90nm technology while the dynamic power consumption reduces, leakage power consumption increases with increase in temperature. The major contributor of leakage current is sub-threshold leakage current that varies with temperature as per below:

$$I_{\text{leak}} = I_0 \frac{W}{L} e^{-\nu_{th}/nKT/q}$$  \hspace{1cm} (5.1)

Therefore the ease of Leakage Differential Power Attacks actually increases with increase in temperature, since the variance of leakage power is elevated at a higher temperature.
5.3.1 Future Directions

So far we have considered a constant and uniform distribution of die temperature. However depending on the leakage in the circuit, practically speaking, this temperature distribution is not uniform across different components of the circuit. More specifically, selected parts of the circuit that emit exploitable power consumption may start leaking more or less power depending on their specific temperature and this can affect the variance in exploitable power. This can be a future direction of research. One can also consider the effect of spatially correlated process variations on the security of cryptographic devices.

Additionally, it would be worthwhile to examine if KEELQ and DES are appropriate algorithms for experiments of this nature. A more complicated cipher like AES must be explored further.
Bibliography


[38] Synopsys, Nanosim User Manual.


[40] [Online]. Available: http://redis.io