DESIGN AND ANALYSIS OF CHALLENGE RESPONSE PAIR GENERATOR USING PUF AND DPA RESISTANT S-BOX

A Thesis

Submitted in partial fulfillment of the requirements for the

award of the degree of

DOCTOR OF PHILOSOPHY

in

Electronics and Electrical Engineering

By

Abhishek Kumar

41400729

Supervised By

Dr. Suman Lata Tripathi

LOVELY PROFESSIONAL UNIVERSITY
PUNJAB
2020

DECLARATION

I declare that the thesis entitled, "Design and Analysis of Challenge Response Pair Generator using PUF and DPA Resistant S-BOX" under the guidance of Dr. Suman Lata Tripathi, Professor, School of Electronics and Electrical Engineering, Lovely Professional University, Punjab, India. No part of this thesis has formed the basis for the award of a degree or fellowship previously.

Abhishek Kumar
School of Electronics and Electrical Engineering
Lovely Professional University
Phagwara, Punjab
Date

CERTIFICATE

It is to certify that Abhishek Kumar (41400729) has completed the scientific formulation of the thesis entitled, "Design and Analysis of Challenge Response Pair Generator using PUF and DPA Resistant S-BOX" under my guidance and supervision. To the best of my knowledge, the present work results from their original investigation and study. No part of the thesis has ever been submitted for any other degree at any University.

Dr. Suman Lata Tripathi
Professor
School of Electronics and Electrical Engineering
Lovely Professional University
Phagwara, Punjab

ABSTRACT

Traditionally, a large scale of integration (VLSI) implementation of a digital circuit is characterized as area power and delay; nowadays, additional parameter security is added as the 4^{th} paradigm. Scaling offers an advantage as a reduction in dimension, but smaller devices exhibit high leakage. A complementary metal-oxide-semiconductor field-effect transistor (CMOS) -based low power VLSI design is a default standard for the electronic design automation (EDA) tool cell library. Implementation of a complex computation function like substitution-box (SBOX), a demanding block in terms of power, area, and security arises certain limitations, additionally leaks information along with unavoidable leakage current, known as side-channel information. The side-channel analysis says leakage maintains relation with processed data, and attackers can access hidden secrets by analyzing it. The dynamic power of CMOS devices is dependent on data to be processed. Power trace varies depending on the number of high bit present and the number of bit position changes in the data. In this research work, we have implemented the SBOX with static CMOS and hybrid CMOS logic (Transmission gate (TG) and pass transistor logic (PTL)). Leakage cannot be avoided, but information leaks with leakage can minimize measure with Normalized energy deviation (NED) and normalized standard deviation (NSD). SBOX implementation with (TG+PTL) logic has achieved a reduction in NED by 48.64% and NSD by 21.8% compared to static CMOS. The cryptographic block's power consumption correlates with the intermediate result, allowing an attacker to predict the input data to be processed. Differential power analysis and correlation power analysis is the most critical side-channel attack, based on statistical analysis. Pearson correlation coefficient (ρ) examines the dependencies between input data and power consumption patterns. A positive and higher value of 'ρ' justifies power pattern is highly correlated with processed data.

An attack resistant cell should not maintain relationships with processed data. Power attack countermeasure examines the strength of the CMOS cell to counter the power analysis attack. Hiding and masking are two widespread attacks resistant features to keep the device secure from a power attack. In this work, we have adopted a boolean

masking technique to randomized the intermediate information. The intermediate value of SBOX is masked with a random mask bit and additional cells; the output is unmasked with the same random bit or derived one. Masking at cell level modifies the cell design so that power consumption should be free from input data without disturbing functionality. Here mask XOR and mask AND cell (require for SBOX) presented whose actual power consumption does not depend on output node, it to be distributed on the internal node equally. The truth table approach is to verify the hamming weight of the internal node uniformly distributed; thus, power depends on all internal node attackers cannot predict the information by monitoring the output node only. SBOX implementation with mask cell improves the static power by 84.4%, dynamic power by 77.6%, delay by 164.8% compare to SBOX with unmask cell. The presented SBOX with Boolean masking at cell level requires 2173 gate counts, which is much lower than earlier reported work. Reduction in correlation coefficients between actual power and approximated power in the hamming weight power model is 32.17%, and 26.65% hamming distance power model implies independence between power consumption pattern and processed data. In Mask, cell input is mixed with the random number; the software mechanism to generate a random number is not truly random and repeats itself after definite sequence length. A physical unclonable function is one of the emerging hardware security modules whose response is a function of challenge input and device/specific feature. Due to manufacturing variation, a silicon-based device generates a unique response from the different circuits is preferred to design a physical unclonable function (PUF). Here we have presented Schmitt trigger PUF(STPUF), whose response is a function of circuit delay and hysteresis width. The presented STPUF validates the parameter with a uniqueness of 49.2% and reliability 99.7% under stringent operating conditions.

ACKNOWLEDGMENT

First of all, I would like to express my gratitude to my supervisor, Dr. Suman Lata Tripathi, for her supervision, advice, and guidance, as well as extraordinary suggestions throughout the work. My special thanks to my previous supervisor Dr. Ravi Shankar Mishra, and Dr. KR Kashwan. I am very fortunate to have the opportunity to work with him. I found that their guidance is precious.

It is our pleasure to be indebted to various people, who directly or indirectly contributed to the development of this work and who influenced our thinking, behavior, and acts during the study. I am great full to my friend and colleague for their constructive suggestion.

I want to show gratitude to the School of Electronics and Electrical Engineering, Lovely Professional University, for providing me with a suitable research environment to carry out the research work.

I want to thank the almighty and our parents for their moral support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

Abbreviations	Description
CMOS	Complementary Metal Oxide Field Effect Transistor
CPA	Correlation Power Analysis
DPA	Differential Power Analysis
EDP	Energy Delay Product
HDPM	Hamming Distance Power Model
HSM	Hardware Security Module
HWPM	Hamming Weight Power model
SCA	Side-Channel Attack
LUT	Lookup Table
MuxPUF	Multiplexer based Physically Unclonable Function
PDP	Power Delay Product
PTL	Pass Transistor Logic
PUF	Physical Unclonable Function
RNG	Random Number Generation
RO-PUF	Ring Oscillator Physically Unclonable Function
SPA	Simple Power Analysis
ST	Schmitt Trigger
ST_PUF	Schmitt trigger PUF
SBOX	Substitution Box
SVM	Support Vector Machine
TG	Transmission Gate

Chapter1
Introduction

1.1 Introduction

Hardware security is a protection methodology that comes with physical devices rather than a software algorithm. While accessing the equipment, we put a lot of trusts, are we interacting authorized devices? Whether our sensitive data is preserved? The cryptographic protocol restricts access from unauthorized users. Cryptographic algorithms like Advance encryption standard (AES), Data encryption standard (DES), etc. require the secret key. During encryption user, input data is mixed up with a secret key, generates ciphertext, which would decrypt with the same key. The security level of the algorithm aligns with the secret key. The contender tries to predict or clone the intermediate round key, elaborated in [1]. Hardware limitation of very-large-scale integration (VLSI) implementation of the cryptographic module highlights the weakening of the security algorithm. Traditionally VLSI circuit's attribute is characterized by area, power, and delay, now a day's 4^{th} parameter identified as security. Security into the system must arise from the root of trust. The hardware security module is a function whose output (response)is a nonlinear function of challenge input and unpredictable for unauthorized users.

Physical unclonable function (PUF) is a simplified structure of the hardware security module (HSM) [2] [3] that, for a part of the given information, provides a physically defined response. Semiconductor-based PUF utilizes unique physical equipment or circuit variation, which occurs naturally during manufacturing [4]. According to NXP semiconductor, electronics properties which are variable into nature and last with the stable state can use to have a PUF system. Selected unique features of electronic devices are included in computation to generate a response; output response is a function of input and device-specific properties. The expectation from PUF is (a) for the same challenge multiple PUF should have different response (b) response of PUF are uniformly distributed (c) PUF's responses should stable even in the unsecured operating

environment. If secret keys and structures are known to 3rd parties, they will not have access to unauthorized access. PUF found two significant applications as secured authentication and key generation in a cryptographic key generation [5] [6]. 2 accessible structure of PUF is delay based arbiter-PUF and frequency variation-based ring oscillator PUF(RO-PUF).

In this work, Schmitt trigger based PUF structure presented; whose output responses are a function of challenge two device-specific feature threshold voltage and hysteresis width. Schmitt trigger acts as a wave shaping circuit to variable signal consists of 2 trigger voltage. Here eight different architecture of trigger circuit has been utilized to generate multiple trigger points. The cascaded connection of Schmitt triggers the circuit to sample the input signal numerous times. Hysteresis width or delay associated with circuit put their effect in the computation of responses for a given input challenge. The first time in the literature that PUF response is a function of more than one device signature. A Schmitt trigger PUF (STPUF) enhances the response bit generation.

AES is adopted as the standard for the specification of electronics data by the national institute of a standard technology (NIST) in 2001. AES algorithm is based on complex mathematical computation. AES-128 bit required 128-bit size of plain text and secret key, repeat internal module AddRoundKey, ShiftRow, SubByte, and MixColumn by the ten rounds. Out of which, SubByte is the most complicated block; it includes the non-linearity into the computation. In this step, 256 single-byte arranges as 16*16 matrix, known as substitution box (SBOX). Two famous architecture of SBOX presented in [7] is a lookup table (LUT) based and computational architecture using the Galois field (GF). LUT based SBOX require the large size of memory and efficient fetching device to substitute by a single nonlinear byte.

The limitation of LUT-SBOX is an unavoidable delay and leakage of the memory element. VLSI implementation of GF-SBOX shown in [8] requires substantial resources. Consumes maximum power in the encryption takes 75% of total power alone. CMOS transistors-based logic gate is the default standard for design lower power circuits, but the power requirement of CMOS devices depends on the input data. In addition to primary

output, SBOX leaks secondary information in terms of power, simulation time, electromagnetic radiation, and sound, known as side-channel information. Dynamic power consists of significant chunk information about hidden data at the internal terminal, which can be revealed. Attackers target the point where maximum information can be extracted from, a power trace discussed in [9]. The power trace of SBOX depends on the number of high bits (hamming weight) of input data and the number of bits changing at the input (hamming distance). Information can reveal from a power trace; like how many highs in the data, time to application of data, change of bit position on MSB or LSB. The attacker performs statistical analysis from the stored power trace, available information, and tries to guess the correct secret key.

Power analysis attack is based on a hypothetical model of power traces obtained with hamming weight or hamming distance of input. Specific statistical parameters, like mean for differential power attack (DPA) and person correlation coefficient for correlation power attack (CPA), are utilized to develop a relationship between actual power trace and mathematically calculated power value. Since the power consumption of CMOS devices is data-dependent, to include power attack resistant features, power consumption must be independent of the input data. Hiding and masking are two popular algorithms of power countermeasures. Hiding countermeasures shown in [10] hides the power consumption of key-dependent cryptographic computations by increasing the signal-to-noise ratio (SNR), randomize the power, and equally balance the power at each moment. Masking countermeasures randomize the intermediate value of cryptographic computation to avoid dependencies. Masking applies at (a) algorithm level where some part of the algorithm needs to rewrite and (b) cell level where each cell of design is resistant to power attack explain in [11]. Boolean secret sharing of mask schema employs new cells into a model with a mask bit. Here input bits mixed-up with random mask bit using XOR gate and final output are unmasked with external mask bit or internally generate signals. A mask gate does not change the functionality but reduce the dependencies of power consumption with input data. Masking countermeasures decreases the correlation between actual power consumption and power value computed by the mathematical model. These

countermeasures enhance the security level at the cost of area and high power consumption.

1.2 Researcher's Contribution

In this work, we have presented the Schmitt trigger (ST) based physical unclonable function, whose response is a function of challenge input and two device-specific features. It is the first time in literature, where PUF responses are a function of multiple hardware-specific features. Another part of the thesis is power attack countermeasure with mask cell. Power attack analysis enables the adversary to extract the hidden information by statically analyzing the leakage with a hypothetical model. The proposed mask XOR and mask AND cell include the power attack resistant feature into the design. SBOX implement with mask cell shows achieve the lower value of the correlation coefficient, Implies the power consumption pattern does not maintain linear relation with processed data.

1.3 Organization of the thesis

The thesis is organized into seven chapters. A brief outline of the chapters is given below.
Chapter 1 introduces the requirement of hardware security. It highlights the key point to design physical unclonable function, and brief about limitation arises due to side-channel information. The author's contribution included in this chapter.
Chapter 2 presents the research work done by a different researcher, according to the author's objective. The literature review is segregated into three categories physical unclonable function creation, power and attack, and power attack countermeasure.
Chapter 3 proposes a novel design of a Schmitt trigger-based physical unclonable function novel. The simulation result of STPUF is thoroughly analyzed, and their achievement has been explained in the chapter.

Chapter 4 proposes an optimized design of SBOX. A detailed description of the Internal block of SBOX covered in this chapter. Implementation of SBOX with static and hybrid CMOS logic is described in the section. The results of the simulation are also analyzed in the chapter.

Chapter 5 includes the power analysis attack onto SBOX with hamming weight and hamming distance power model. The dependency of actual power and predicted power is analyzed in the chapter.

Chapter 6 highlights the attack resistant feature of the logic cell involved in the design. This chapter analyzed the masking technique to mask the sensitive information of the circuit.

Chapter 7 concludes the thesis and highlights the prime outcomes of the author's current research and the significant contribution of the thesis and notifies the scope for future research in this area.

Chapter 2
Review of literature

Historically three-parameter power, area, and delay used to optimize semiconductor design. The area is related to cost; larger areas reduce the number of dies onto a wafer led to an increase in processing and material cost. The performance of the VLSI circuit measured in terms of circuit speed. Lower propagation delay leads to higher performance of the digital circuit. Power dissipation has a profound impact on both cost and performance. High power dissipation requires additional heat removal and advanced packaging technology, increase the cost and system size. Power and delay are contradicting factors as a function of the supply voltage. Delay of circuit reduces with higher supply voltage, but higher supply voltage results in high power consumption. For the given circuit these factor trade-offs according to the application [15]

CMOS is dominant in the VLSI era since 1970. The progress of VLSI technology makes an instructive trade-off in the I.C. design process. In 1970, yield or die area was the primary concern. With advances in technology in 1980, circuit speed is the principal limit. With the progress of scaling era of 1990 was focused on optimizing speed and power. Integrating a higher number of components onto a wafer leads to noise, in 2000, is known for optimizing the power and speed with noise. 2010 is known for the inclusion of security at the design level. It is an additional block in the circuits introduced to make the leakage information independent of the data to be processed, leading to an increase and die area, power consumption. However, the delay can optimize by a directed critical path.

2.1 Hardware Security Module (HSM)

Hardware security is an emerging field of engineering to enhance the security level of the cryptographic algorithm by including the unpredictable features form semiconductor devices.

2.2 PUF as HSM

A physical unclonable function is an innovative circuit extract unique intrinsic feature of the integrated circuits. Gassend and Blaise et al. in 2002 introduced delay PUF [16] based on the silicon IC-based PUF circuit. Two silicon devices cannot have the same properties in all aspects, even manufacturing with similar material and corresponding instruments. G. Suh and S. Devdas in 2007 [17] introduced the application of PUF into authenticating and random numbers of generation. PUF response should be unique for each challenge and not able to predict by other users. The feature of the PUF circuit is validated by Horie et al. [18] in 2013 with uniformity, uniqueness, and reliability.

2.3 Power Attack on AES -SBOX

Traditionally, a circuit's performance depends on area, power, and delay; time demands to add 4^{th} parameter security. Security should emerge from the hardware, preferably only with a cryptographic algorithm. A circuit response must be secured; the adversary must not predict the circuit detail like internal block function, hidden secret, etc. During computation, the VLSI circuit must leak secondary information in terms of power, delay, electromagnetic radiation. According to [64], power analysis is the most efficient side-channel attack. The power attack starts with identifying high-power consumption points in the crypto circuit so that maximum information can be collected. Identifying sensitive location in a power attack is necessary, where the relation between power trace and data can be analyzed. Substitution box in AES is the hungriest power consumption block, 75% of encryption/decryption power consumed by SBOX alone. SBOX is the most demanding block in terms of area, power, delay, and security measure. Figure 2.1 shows that a typical target point is the output of SBOX. AddRoundKey mixes the 128-bit input byte with the same sized secret key followed by SBOX. An SBOX operated on a single byte; thus, 16 SBOX needed in parallel to performs substitution. The output of each SBOX provides an exciting point to collect power traces. Since the algorithm is available for public, input, or output, it may also be available, only unknown is the secret key byte. An

attacker stores the power information [127] for each possible input and tries to predict the processed data at SBOX. The data can reveal time to application of data, number of high bits in the data of SBOX, number of bits switching at the input. SBOX consumes minimum power for the lowest of hamming weights of input data, and power consumes maximum for the highest of hamming weights.

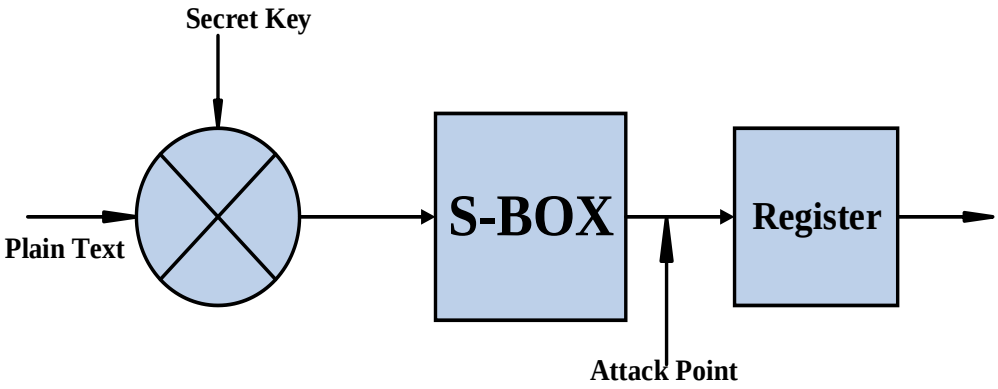

Figure 2.1 Attack over SBOX

2.4 Side-Channel Information

A side-channel attack is a methodology to obtain a circuit secret based on information gained rather than the security algorithm's weakness. Security algorithms like DES, AES is mathematically secured require primary input by a user, and secret key computes primary output. [61, 71] described VLSI implementation of such an algorithm suffers from hardware limitations. CMOS device suffers from leakage issues, and this issue arises more for scaled MOS transistors. CMOS implementation of the cryptographic algorithm does compute primary output, in addition to that leak's secondary information. This information contains information related to circuit internals terminal; the attacker uses this information to gain access over the device by reverse engineering known as a side-channel attack. Sources for side-channel attacks are power, heat, sound, timing, fault, etc., out of which power attack is considered the most productive offence. Side-channel attacks presented in [69, 82] classified as (a) passive side-channel attack utilizes leakage information external to a device with the interface. (b) An active side-channel

attacker manipulates the device, injecting probes, and injecting fault to learn secret information discussed in [126].

2.5 Categorization of Literature Review

2.5.1 Literature Review of PUF

In paper [19] by Maiti, A., Kim, I., & Schaumont, P. (2011), challenge-response pair (CRP) is the essential requirement of device authentication explained, for each challenge generated response must be unique. A reliable PUF must create a different response from multiple responses for the same input challenge; A reliable PUF must make a large number of random numbers to derive the secret key. In this paper, the author suggested an R.O. based scheme to enhanced CRP set. The entropy of a PUF is controlled by circuit parameters to build it. Large size key and a broad set of challenge-response pairs enhance the circuit hardware area and hardware cost. Here the author has implemented ROPUF to generate the same size of CRP pairs with reduced resources. CRP generator using ROPUF map to FPGA platform capable of creating new functions using statistical hypothesis test; reduce the area by two times. PUF quality is measure using uniqueness, reliability, and unbiased. The average value inter-chip H.D. of presented PUF is 50.07% achieve high uniqueness and reliability of PUF output with the slow area. A PUF circuit should able to generate individual responses even in a secured environment. Memory-based response bit generation at high temperature and threshold voltage variation. Study of the effect of environmental and diversification into supply voltage in terms of mean, standard deviation, and its limitation explored.

In paper [20] by Eiroa, Susana et al. (2010), different structures of PUF, like Arbiter PUF, Ring oscillator PUF, Butterfly PUF, and NOR-based PUF architecture designed and their performance compared in term of inter hamming distance, intra hamming distance. PUP map the set of challenges bit/byte to a set of responses bit/byte driven by physical properties that are difficult to predict and reproduces. 2 significant applications of PUF have secured authentication and hardware-based random number generation [21]. Exact random generation scheme based on silicon I.C. discussed, and their statical analysis of

randomness, uniqueness reliability explored. Implementation difficulty and their sensitivity to the environment investigated. The selection of PUF could be difficult since several parameters to control to generate the response. Here the author has presented a report to various PUF architecture; reconfigurable PUF dynamically modifies the inherited properties. Controlled PUF uses a general processing element to access through a specific application. Leakage current PUF based on leakage current of a cell, leakage current to be converted to a digital one, and codified to obtains the I.D.

In paper [22] by Kumar et al. (2008) presents one of the most challenging tasks for I.P. vendors, FPGA manufacturers are I.P. protection use bitstream encryption. PUF is a different physical system extract response from the intrinsic physical properties of the integrated circuit. Describes PUF provides un-clonability means of a highly secure method of generating a volatile secret key of the cryptographic operation. In this paper, the author has utilized D-latch's unpredictable behaviour to settle on a stable state after metastability due to setup and hold time violation. A cross-coupled D latch structure acts as butterfly PUF implemented using I.P. available in every FPGA, i.e., CLB, BRM. Inter and Intra hamming distance variation at temperature variation from -20 to 80 C is analyzed. Validation of PUF with a difference of hamming distance measured on the FPGA and between the difference in H.D. of multiple FPGA. Butterfly PUF uses the internal matric of available resources on FPGA to generate the response using the integrated circuit's intrinsic physical properties. Butterfly PUF is a promising module that generates stable response and FPGA operating environment.

In paper [23] by Katzenbeisser et al. (2012), the contribution is ASIC implementation of delay arbiter and ring oscillator (RO-PUF), SRAM memory-based PUF based on SRAM, latch and flip flop is designed using CMOS 65nm technology. The author has analyzed the properties of PUF at temperature, supply voltage, and noise. ROPUF achieve all desired properties [24], SRAM PUF instance is independent. Evaluation result based on 96 ASICs based on TSMC 65nm CMOS technology. PUF is analyzed based on the following properties (a) robustness- quantified by bit error rate, the number of bits in a response1 is different from response2. (b) Unpredictability contenders not efficiently

compute the response of PUF to a known input challenge. The author has experimented with ring oscillator and SRAM PUF and analyzed it with 96 ASICs. Arbiter PUF sows very low entropy; a change influences the entropy of flip-flop and latches in temperature. Suitability and selection of PUF are essential for critical application for authentication and key generation.

In paper [25] by Majzoobi et al. (2012), slender PUF protocol presented a secure method to authenticate and respond generated from strong PUF using arbiter. Linear arbiter and two independent parallel arbiter path response mixed (XORed) to put into effect PUF with better statically properties. Slender PUF used the right random number and found that arbiter PUF of length 64 modelled as 95% accuracy. Presented slender PUF does not follow the classical paradigm of exposing the complete responses, a random subset of responses sent for authentication. The response pattern is lightweight and possesses resilient against machine learning attack, additional error correction logic, and fuzzy extractor for robust authentication. The critical length of slender PUF of 128, 256, and 512 achieve hamming threshold 33,76 and 154, respectively.

In paper [26] by Handschuh et al. (2010), PUF is used to extract the key and be considered an electronic fingerprint or feature of a device. PUF involves a physical device's feature challenging to clone due to the unique micro and nanoscale properties driven by manufacturing variation. Cloning of PUF requires lots of time, complex mathematical models, and software programs to compute the precited response of a challenge within the expected amount of time. Here author used variation among threshold voltage in the transistor of SRAM as a biometric feature. Fast and slow transistor chooses logical preference to settle both logic 1 to 0. The secret key stored in non-volatile memory; in this paper, the author introduced a reliable key storage technique on SRAM, ROM. A threshold voltage of different transistors achieved by carrying a length of individual transistor and provide a unique start-up value from each cell. Quality of PUF evaluated with operation consultation variation into temperature, voltage variation, and radiation. Reported PUF of different PUF are; SRAM PUF 950/1000 bits Delay PUF 130/1000 bits and Butterfly PUF 600/1000 bits.

In paper [27] by Barak et al. (2014) true random number generation scheme is explained. The random number should not be computable. PRNG based random number is not entirely non-deterministic, and the upcoming number can guess statically based on the previous history. The random number found applications in multiple critical applications in a lottery, Monte-Carlo simulation, cryptographic key generation, etc. The software mechanism to generate random is not entirely random and repeats itself after a particular set and can compute if the initial seed or previous number pattern is known. A true random number generator mix selected propertied from the hardware. Generated random number is a non-linear function of this feature. In this paper, the author has discussed the noise-based entropy generation, which is further processed to create a random number. R.O. based glitch is the source of entropy for generation scheme. The author has discussed the open problem and outlook in random number generation. The noise mechanism generates small voltage strongly amplified and converts to digital form. Since the I.C. designer reduces noise to any level design of the amplifier is cumbersome. Chaos RNG conceptually mixing the randomness, set of free-running oscillator generates multiple frequency, randomly chosen rate applied to digital block to have a random bit.

In paper [28], Bhargava et al. (2010) Attack resistant sense amplifier based PUFs based on the sensed voltage. The amplifier acts as a reading path in static memory. This circuit able to detect small voltage variation in the cell. In this work, the author has designed a large number of SRAM cells with a different threshold voltage. Forlag input, each cell carries a different voltage. For each column sense amplifier, enables, Generated response depends on offset voltage and mapping scheme. Sense amplifier resolves 0 or 1 bit with equal probability, with minimum voltage microvolts. A post-processing block corrects the error onto the response to have a stable and reliable bit. The cost-effective technique of error-correcting code (ECC) block presented in the paper sensed amplifier PUF is resistant to environmental change. SA PUF architecture accesses the S.A. array challenge bit select the address the response passed through scrambler. CMOS 65nm technology-based simulation result present that average Hamming distance 16.000937 and reduce the area by 2.5-time.

In paper [29] by NXP semiconductor (2013), Processor-Based Strong PUF verified with an ageing analysis of response reliability. Process variation based inherent randomness in silicon I.C. affect threshold voltage, effective gate length, and various side effects. Two different ALU core response arbitrated to have final response bit. in between circuit xor gate additionally intended to have data obstruction. Hamming distance variation on applying various challenge and PUF behaviours are studied. A smaller circuit added into the processor known as on the fly extraction of response bits to have a unique timing signature. The presented PUF achieves average Hamming distance 16.1% for a 32-bit response; the ageing parameter measures the deterioration of logic level with time—this difference used as a stimulus to an amplifier and digital conversion block. CMOS 45nm technologic node implementation shows the average intra chip hamming distance by 98.1%.

In paper [30], Hossain (2015) promising security module based ROPUF random function of I.C.'s specific functions that are unique for every instance of the die. I.C. manufacturing variation used to generate a different kind of PUD. RO is the best candidate for the PUF circuit; it always gives properties as expected. In this work, the author proposed 13 sets of RO PUF generating independent frequency, each stage of inverter in R.O. designed in CMOS and FTL inverter. A comparative study of normal and FTL inverter presented; FTL more inverse comprises additional NMOS in parallel gate terminal controlled by the independent clock. FTL inverter provides a high operating frequency. Randomly one of the frequencies selected to respond a bit. Achieved result show that uniqueness is 45.24% (ideal 50%) reliability 91.14%(ideal 100%) and uniformity 41.45 %(ideal 50%). PUF quality measure uniqueness, reliability, and uniformity with PVT variation reported with low threshold voltage 46.85%, 95.54%, and 40.799% for high threshold 41.71%,90.12%, and 42.34% respectively.

In paper [31], Maiti et al. (2009) application of PUF presented as True Random Number Generator (TRNG). PUF and TRNG are two successful components in secrete system design. PUF extracts a random chip signature to generate a volatile secret key and TRNG use for making random padding bits and initialization vector. Here the author has

explored R.O. based TRNG; their original contribution is area reduction, scalability, portability to different platforms, and bridge between circuit and system. Quality of bitstream from TRNG is passing on the NIST test suit and DIEHARD test suite. RO PUF implemented with Xilinx's Spartan 3S500E series FPGA with 32 and 64-bit rising oscillators. RO PUF response is 42.8 % unique and 80.6 % reliable; it observed that with the high number of R.O. stages, uniqueness and reliability start to decreases. An XOR gate controls the circuitry giving 3.2 MBPS truly random bits and 32-bit unique device signature.

In paper [32], Kong et al. (2013), low power and smaller circuits for device authentication, which size PUF compact. The threshold voltage variation of nmos and pmos is, and the temperature co-efficient of threshold voltage significantly improves the stability of silicon PUF. The different architecture of inverter with different width size enhance the reliability of PUF. Technology dependent PUF with CMOS 90nm and 45nm response tested with NIST test suit. Leakage current based PUF used leakage sensor; the output of each sensor is identical, but process variation includes a slight change into the PUF response. CMOS 90nm SPICE with Monte Carlo simulation analyses with 2000 chip instance, leakage PUF score 60% reduction and power, and 84% reduction in area, 35% improvement in temperature stability compare to RO PUF.

In paper [33] by Maiti et al. (2013), transient effect based ROPUF has proposed. The design, evaluation, and optimization of the ROPUF transient effect. The temporary impact of the ring oscillator during start-up adds more randomness and generates response more reliable than ROPUF. Response generation scheme with TERO PUF tested on 30 chips with CMOS 350nm process technology in a nominal corner and voltage temperature condition. TERO PUF is focused on optimize performance and reduce area without noticeable loss in output. An additional balancing block extract high-level entropy. TERO PUF can generate several response bits from the same circuit; TERO PUF causes a unique response by extracting oscillation from TERO. Uniqueness involved in producing responses is 49%, steadiness 5%, and Randomness 99%.

In paper [34] by Shaza et al. (2015), PUF design is presented with the unique feature if remanence decay. Non-volatile memory used to store the binary value for a long time; remanence is the property that measures the channel in the stored information with time due to decay or leakage or environmental conditions. Here 8KB SRAM array PUF proposed as lightweight security primatial at CMOS 65nm technology. SRAM PUF has a countermeasure against side-channel attack used remanence decay to improve cloning resistance o SRAM PUF. To generate stable high response voltage remanence 44.82% and time remanence 43.455. Similarly, log logic response voltage remanence 45.14% and time remanence 41.86%.

In paper [35], Herder et al. (2014), PUF broadly classified as weak and robust PUF; strong PUF can generate secret key while weak PUF response process through error-correcting code. Author have Implemented arbiter, ring oscillator, and SRAM based PUF in CMOS 180nm technology. According to the application, it Classifies PUF as a string and weak PUF. A strong PUF suitable for secured authentication and weak PUF uses for cryptographic key generation. A weak PUF requires a small number of CRP and exploits the manufacturing variation while strong PUF requires a large number of CRP.

Additionally, error correction schemes on the PUF response used to produce stable and reliable answers. The author has tested the different topology of PUF like optical PUF, Arbiter PUF, and RO PUF. Obtained PUF response inter and intra variation is 6% and 5% respectively at temperature range -25 to 85C. PUF is resistant to training attacks since it is challenging to measure the internal hardware parameter, testing to clone, challenging to predict the past CRPS.

In paper [36] by Meguerdichian (2011), A new design of dynamic physically unclonable functions (DPUFs) proposed whose physical properties are subject to unpredictable changes between uses. Here device ageing adapted because it provides rapid and low-energy customization and exhibits fast reversibility since. Here the author has proposed numerical algebra technique for reverse engineering. Cad tool ages model based on negative bias temperature instability (NBTI) and hot carrier injection (HCI) where

selected hardware features for PUF non-linear function decay with time. Pair of a module under ageing effect show variation, based on a variation arbiter decode to generate a response of logic one or logic 0. To include attack countermeasures, a series of buffer to be added. Dynamic PUF has high unpredictability and can quickly stabilize.

In paper [37], Zouha et al. (2012) method that allows evaluating a silicon PUF, based on delay elements, at the design stage without the need to have the circuit. The essence of the way based on a Monte Carlo (MC) simulation of the netlist that considers the process variation and the environment. Evaluation of PUF is an open problem different parameter is analyzed. Randomness derived from maximum entropy. Uniqueness says two PUF must have a different response of the same challenge and reliability say response bit not be affected by the environmental and operating condition—arbiter PUF designed with HDL and simulation results based on Cyclone II FPGA. To be validated, this method compares the simulation results and measurements on silicon, showing that arbiter PUF has a uniqueness of 97% and seediness of 99%. Loop PUF utilizes the delay chain, the odd number of cascaded structures like ring oscillator follow by frequency measurement device achieves uniqueness 95% and steadiness 98.7%.

In paper [38], Pegu et al. (2015) multiplexer PUF based non-linear and unique response for each challenge input by taking advantage of an uncontrollable intrinsic random feature of the integrated circuit during manufacturing variation. This paper presents the well-experimented analysis of the different MUX-based PUFs based on layout-based simulation performed in CMOS 50nm.rul—using Microwind and DSC2.7 tools. A comparison of the various topology of MUX based arbiter PUF presented. The feedforward structure of mux PUF enhances the nonlinearity in mux. MUX PUF gives the reliability 88% uniqueness 79% and randomness 86%. Inter and intra chip variation with the different condition is 11% and 9%. A feedforward structure increases nonlinearity and decreases errors caused by environmental conditions. MuxDmux PUF selects the direction of the propagation signal and implements reconfigurable PUF.

In paper [39] by Ozturk et al. (2008), delay-based PUF based on intrinsic process variation and random switching by input challenge that implements the pseudo-random

function. In addition to logic 1 and 0 tristate have 3rd stage as high impedance (Z). IF the gate is not enabled output is tri-states, A tristate delay unit implement with BUF0 and BUF1 stage when letting the same input pass since the delay of BUF0 and BUF1 is the different response time of tristate buffer is different. The tristate barrier is similar to arbiter PUF, where the inverter substituted with tristate buffer Delay of BUF0 is higher than BUF1 due to addition invert gate. A mathematical model can compute the response time for output. Synthesized result-based ion TSMC 130nm technology library shows at 100MHz mux PUF require 193.67μW and tristate PUG require 152.93μW with 49 units of reduction in getting counts.

In paper [40] by et al. Hussain(2014), an online test evaluation for testing quality metrics, i.e., NIST test, Diehard test, Dapra Shield test. BIST-PUF enables the on-the-fly assessment of the desired PUF properties, all in hardware predictability and stability. BIST PUF has its response generator integrated with a test suite. An online assessment tool developed to report quantitatively and to evaluate PUF stability and unpredictability. BIST PUF assesses by three different methods sensor-based, parametric interrogation, and multiple interrogations. A high entropy TRNG includes BIST PUF to exclude the biases in challenge generation. Stability tool removes the requirement of error correction logic, Xilinx Vertex-5 based FPGA implement power requirement of BIST PUF are 18.92mW.

In paper [41] by Yao et al. (2013), new PUF techniques can extract secret from pairwise skews between sinks of a clock network. These techniques acquire the stability of the clock network to acquire unique chip signatures from process variations. The clock network implemented in CMOS 45nm technology node. On-chip clock network builds resilient to attack; tunable delay enhances the entropy of clock PUF. Algorithmic selection of pairwise comparison performed by multiplexors, the inclusion of return oath minimizing the routing congestion. SPICE simulation measure as Inter and Intra hamming distance verify on performing Monte Carlo analysis with 1000 key. PUF arrangement shows inter-Hamming distances ∼50% as desired. Reproducibility is

responses ensure the response is independent to ±10% in voltage and -20°C to 120°C temperature. Compare to the ROPUF clock, PUF has a 2-time higher reproducibility.

2.5.2 Literature Review of Power Attack Analysis

In paper [42] by Mestiri et al. (2013) number of power traces necessary to guess the key of AES SBOX with the SASEBO-GII board has experimented. This board comprises cryptographic FPGA and control FPGA supports HDL programming language. Power trace to compute output plotted on to CRO. Here the relation between actual power and predicted with hamming weight power model or hamming distance power model and switching distance exploits to implement CPA attack. Here the author has focused on how many power traces are required to ascertain each byte of the correct key. Based on 4000 traces correlation coefficient between actual and estimated model 10^{th} key can distinguish. The switching distance model requires less power to guess the correct key successfully compared to the Hamming distance model. H.D. and S.D. model shows a high degree of approximation at corresponding key; 1^{st} pivotal HD 5.7 SD 5.6, 4^{th} key HD 6.1 SD 6.5, 8^{th} key SD 8.5 HD 11, 12^{th} key SD 4.4 HD 5 and 16^{th} key SD 9.3 HD 9.3.
In the paper by [43] by Takeshi Fujino et al. (2017) power consumption pattern linearly depends on the number of switching bits at the input of the cryptomodule. Authors have reviewed various countermeasures against side-channel attacks. A countermeasure cell does not reveal hidden circuit information under a power attack; it provides the random or constant power for inputs. Attack resistant cell shows a notable increase in circuit area and power dissipation. SBOX implementation with Threshold implementation and HMDR-ROM shows excellent resistance to side-channel attack, but area increase by five times and power 15 times in case of T.I. and 6% and 20% for HMDR-ROM. MDR ROM based SBOX implementation shows the constant ability to fetch substituted byte from memory.
In the paper by [44] by Massimo Alioto et al. (2014) have explained the effectiveness of power attack countermeasures to leakage power described by Leakage power in the CMOS circuit is unavoidable. The effectiveness of leakage power analysis to implement

differential power attack (DPA) and their countermeasures. Leakage power is smaller but unique for each input, here the author tries to recover key of the crypto module by analyzing the leakage power consumption. Total leakage current is a combination of two distinct currents; the first part of the current depends on the hamming weight of both text input and round key applied on add round key. The first part of the current linearly increases the number of high bits, and another part of the current is a constant leakage current. The coefficient of correlation shows the probability of successfully guessing the correct kay. CMOS architecture of innate cells such as WDD, SABL, MDPL can achieve a smaller value of relationship but increase the area.

In the paper by [45] by Junrong Liu et al. (2010)have shown correlation power analysis is a useful technique to predict secret from cipher by analyzing H.W. and H.D. is the two most preferred models to apply correlation power analysis (CPA) with steam cipher MICKEY v2. H.W. model map the presence of the number of high bits at the output and H.D. model map the number of transitions occurring at the input with power consumption in the actual model. Assumption follows to model the estimate; the model is the power consumption for load capacitor charging when output changes 0-1 and discharging for production 1-0. No power consumption occurs for 0-0 and 1-1 transitions. The limitation of the Hamming weight model is it can recover the key more effectively. For k=0 and 1 correlation effect in -0.43 and 0.4 respectively, the power consumption follows inverse relation to the hamming distance. Here authors describe that only ten power trace requires the difference between k=0 and k=1.

In paper [46] by Eiric Brier (2004) have a mathematical model of DPA and CPA. CPA and DPA are two commonly used power attack analyses to evaluate the crypto module's side-channel analysis. DPA attack relies on the difference between the mean value of power trace at input LSB 0 and LSB 1. At the same time, the CPA utilizes the correlation coefficient to measure the relation between estimated power and actual power. Here the author exposed the defect in the former approach of differential power attack and presented a mathematical model of correlation power attack. The CPA model finds a correlation between a power consumption pattern and the number of high bits to an input

of SBOX. Limitation of DPA attack is a DPA attack that shows a ghost peak, whereas the CPA attack presented as the higher value of the correlation coefficient. CPA attack applied in the first round of 8 SBOX with only 40 power traces. The maximum amount of correlation achieves for the first SBOX is 92% and a minimum of 65%. Compare to CPA, DPA requires more sample power traces. Even partial correlation still provides an exploitable indication of attack analysis. CPA must require reverse engineering (leakage rate) on known data.

In paper [47], Satok et al. (2001) present a methodology to retrieve a key from cache memory between processor and memory system. Cache memory uses in the novel process sharing scheme in a cryptographic algorithm. Lookup table-based SBOX exhibits the cache resistant feature; since each time slice provided b victim is tiny gives a limited number of table access while accessing the table if available byte cache hit else cache miss. After cache miss flush and loading new byte required. This information used to exploit the side-channel attack. The author in [47] has demonstrated the side-channel attack (SCA) base on the recent data available in cache memory as hit or miss. Requirements of large-sized cache memory highlighted if the victim has up to 60 cache access open can develop an analytic model to develop leakage model. Cache attack can recover a full 128-bit AES key using 5-7 encryption or decryption.

In paper [48] by Cecile Canovas and Jessy Clediere(2005) have classified power attack as DPA, DEMA, and CPA. DPA and CPA use the leakage power information at the SBOX, while DEMA is based on electromagnetic radiation while a module performs computation. A capture probe requires the collection and process of the E.M. radiation for each input to the SBOX. DPA and DEMA exhibit the wrong key with ghost peak, but in CPA correct key and the false key is identified as the various weighting of the correlation coefficient. CPA attack based on the hamming distance model applies to multi-bit and single-bit output of SBOX. Hypothetical model of hamming weight and hamming distance normalized with the factor of 32. Correlation value replaces the ghost peak problem and gives better classification in terms of the correlation coefficient.

Imbalance increases the wellness of the approach. These types of tests improve the rectify and relevance of SBOX architecture before the actual test.

In paper [49] by Davide Bellizia et al. (2016) exploited the dependency of the static current of the submicron CMOS device with internally processed data. Power attack can recover the secret key by the univariant attack, by statistically quantifying the relation between actual leakage current and leakage model. Countermeasures of power attack are WDDL, MDPL, SABL shows hiding the information into power trace or randomized the power such that equal power paternal for different input. A new type of power resistant logic is time-closed logic (TEL) into dynamic CMOS logic implementation. The security level s measured as a function of the data path and CMOS technology, and it is independent of the clock signal. The evaluation person s set to 10ps, which is lower than predicted settling time of transient time. Thus, the attacker does not get sufficient time to collect the power trace.

In paper [50] by Y.J. and Noh, M.J. (2005) presents a survey report on differential power attack and their countermeasures as masking. Leakage power is the source of power attack analysis since power value is related to input data. The power trace of the masked cell does not follow the linear relation with input. Masking based on a regular Boolean logic gate such that power consumption value does not depend on the primary data rather than distributed in the internal stage of the circuit. Information is masked with random bit and output bit in again unmasked i^{th} another set of the random bit. A combination of logic gate arranged such that final output unaffected; Here, the author has tested 4-bit ripple carry adder and AES, SEED, and SHA function with mask gate. Masking increases the gate count and critical path delay. AES with masked SBOX increases gate count from 17.4K to 25.9K.

In paper [51], Owen Lo et al. (2017) implemented a DPA and CPA attack on SBOX output. DPA attack presents the difference between the mean value of power trace for LSB1 and LSB0. A peak in the differential curve is identifying the correct key while the wrong key presented ad ghost peak. CPA attack utilizes a Pearson correlation value easy to interpret, Power trace storage setup developed with Arduino uni, key-sight

oscilloscope, probe, and data analyzing machine. Limitations of DPA are the availability of open-source code of cryptographic libraries that capture the power trace in real-life devices during operation. By selecting the exact attack point for DPA and CPA, one can predict the output of the AddRroundKey for the given plain text. For correct guess, one can have the original secret key.

In paper [52] by Saravanan P and Kalpana P.(2015), the XOR gate's importance is presented in the encryption process. CMOS based XOR gate is vulnerable to power attack. The security level of XOR gate measures as non-leakage on information along with power trace shown by Here, the author has proposed a new architecture of the XOR gate with charge sharing option in adiabatic logic. NED and NSD are two energy-based parameters present power attack resistant features. NED measured as a percentage of the difference between the minimum energy and maximum energy, and NSD is deviation into energy for input plain text. NED-NSD value of different existing architecture of XOR like CPL, DCSVL, Adiabatic, SABL, SyAL, CSSAL shows that reduction into NED and NSD but in increases the gate count and power consumption. The proposed XOR gate saves energy 79.6% and 82.5% concerning SyAL and CSSAL XOR gates, respectively, and shows a reduction in NED by 20.8% and 1.6% and NSD by 26.1% and 91.1% compared to SyAL and CSSAL. Limitation exhibits in area requirement show 409.1% and 463.6% concerning the SyAL XOR gate and CSSAL XOR gate, respectively.

2.5.3 Literature Review of Power Attack Countermeasure

In paper [53] by Lu Zhang, Luis Vega, and Michael Taylor (2016) have implemented a Power model to verify DPA and CPA attacks over the crypto module and their countermeasures. CPA attack performed with a hamming distance using a power consumption pattern with the change of high bits in the input. At the input data of CMOS circuit change of bit 0-0 and 1-1 does not consume power (assumption), power trace of XOR and AND gate gave the visible appearance of input bit. DPA and CPA successfully recover the secret key with a high correlation factor. Masking is a pronounced technique

to have a secure crypto module. Usage of the masked gate to implement masking at algorithm level capable of reducing correlation factor compare to without mask gate. Mask gate considered better countermeasure compare to masking logic like WDDL, RSL, SABL.

In paper [54] by Stefan Mangard et al. in 2005, it was observed that the mask gate's performance decreases with glitch; the presence of glitch introduces susceptibility to DPA attack. DPA countermeasure tries to remove to reduce correlation ultimately between actual power and the hypothetical power model. The objective of countermeasure by keeping power consumption of the CMOS circuit independent of the input pattern. A mathematical model to study the energy requirement E0-0, E0-1, E1-0, and E1-1 for output 0-0,0-1,1-01 and 1-1, respectively. The energy required to produces one is more significant, and energy needed to switch output 0 provides an opportunity for DPA attack. Attack implementation of a mask gate requires a 45*16 power trace compare to 25*16 on a standard gate. Mask gate not only makes the computation difficult to modifies the shape of the differential curve of mean in DPA analysis.

In paper [55] by Maneesha Jayakumar (2018), the efficient architecture of mask AES engine are resistant to DPA attack. A Boolean asking approach is applied to implement masking of the AES algorithm. The boolean equation of each component of AES given as a present logic gate-based schematic. SBOX is considered the most sensitive point on the encryption process, masking applied at the input and output od SBOX. Masking a methodology to change the input of SBOX by XORing with the random bit, the output of SBOX masked must be unmasked with another set of the random bit. The random bit at input and output of SBOX used for masking must be related such that the final production of SBOX should be as expected to unmasked data. Power trace of masked SBOX and unmasked SBOX must be different from each other. FPGA implementation of unmasked and masked AES shows the delay is 36.886 nS, and the memory requirement is 342.532 Mb.

In paper [56] by Tiri, Kris et al. (2005) have analyzed power resistant logic styles to prevent DPA attacks are SABL; WDDL and TDL are focused on random charging-

discharging of load capacitor. Minimized energy technique SBOX based on adiabatic charging proposed in this work. The energy required to charge low to high transition is $2(R/_{tp})C_L VDD^2$, adiabatic pre-charging, and evaluation change the power trace. The technique to make output power independent of input bits are; charge sharing symmetric adiabatic logic SBOX provides balanced output power for each transition. eCSSAL SBOX achieve Emin 4.85pJ Emax 5.87pJ at 50MHz. DPA resistant adiabatic charging-discharging achieve NED NSD are 3.7 and 0.65% at 50MHz.

In paper [57] by M. Masoumi(2019) have implemented an efficient architecture of AES based on a modified Boolean masking scheme without adding area overhead. A classical randomized table included breaking the correlation between actual power and mathematical power model—the difference between the ability to compute outputs 0 and 1, identified as Welch's t-test. The XORed output of the add-around key stage is applied to SBOX to find a substitutable byte and mask table. The mask table is 16*16 matrix contains a unique mask value for each XORed output. At output SBOX output and mask, value is again XORed together to find unmasked output. The outcome of the paper found as; affine transform shows a 300% increase in area 60% decrease in speed.

In paper [58] by Tena-Sanchez et al. (2014), cryptanalysis is a technique that explained to find hidden secrets by a third-party power attack with the correlation between power and input such that resistance to DPA attack. Limitation of mask logic style presented, such as; SABL sensible to unbalanced output, SyCML shows reduced swing, TDPL generates an additional control signal, DDPL needs an additional inverter. Here, the author has modified ht pull-down criteria during the precharge and evaluation phases with single and double switch implementation. It reported that for AND gate improvement in NED value during precharge 0.17% with a single switch 35.15% with double switch, in NSD 10.9% with a single switch and 27% double witch switch. Improvement in NSD value during evaluation 32.95% with single switch 157.17% with double switch, in NSD 35.08% with a single switch and 53.05% double witch switch. During the evaluation phase, XOR / XNOR architecture requires the energy of 1.28 times and 1.02 delay 1.14time and 102times for double and single switch pull-down compared to classical implementation.

In paper [59] by Elisabeth and Stefan Mangard(2005), hardware limitations with side-channel attack reviewed. Three categories of SCA are timing, power analysis, and electromagnetic attack. Power attacks further classified as SPA, DPA, and CPA. DPA attack over SBOX implemented with power traces, a difference of mean curve for key LSB=0, and LSB=1 present ghost peak for wringing and peak for the correct key. Countermeasure of DPA attack re hiding- device power consumption changes such that attackers cannot find data dependency. Different hiding implementation is dual-rail precharge, current mode logic (CML), dynamic current mode logic(DML) sense amplifier-based logic. Hiding is unprotected implementation; these changes either time of evaluations or reduces power value. They were masking at cell level randomized the power value of the cell of the cryptographic device. Limits of DPA resistant logic style are balancing of cell and interconnect layout, are overhead, memory requirement, glitches are highlighted.

In paper [60] by Juanli Zeng and Cheng Xu (2012), explains power attack countermeasure is masking at the gate level is popular methods to protect SBOX from power attack analysis. WDDL and SABL logic implement attack resistance gate. Still their chance to get sensitive information from the masked gate. Here an improved mask AND gate and secured XOR gate is proposed in this paper to compute a non-linear function of SBOX, i.e., GF28. Masked gate applied modified input to the logic gate, as a non-linear function of ordinary and masked data. A masked SBOX AES-SBOX implementation at 180nm CMOS technologies with masked AND gate shows 635 gate count rather than 198 in unmasked SBOX, delaying from 7.02n to 9.44 ns; static power increases from 22.973nW to 51.151nw and dynamic control 4.187mW to 16.2mW. 32-bit masked SBOX implementation with Virtex-5 XCVLX30 requires 594 gate slices. The limitation of the masked S-box is 34% slower than the unmasked one. Power requirement and area measures twice in contrast to unmasked SBOX with implementation at CMOS 0.18μm CMOS standard cell library.

In paper [61] by Shuaiwei Zhang and Weidong Zhong(2018) SBOX protection scheme by converting multi SBOX and number input 4*4 instead of 8*8. Reusable SBOX

framework highlighted. The smaller size of SBOX shows power consumption for a minimized number of bits shows a reduction in power consumption. The security of the logic level is analyzed by power value to compute output high or low since the difference in power is DP=p1-p0-ε/2. The zero value of D.P. shows the power and input data are completely uncorrelated. The objective of DPA resistant logic style o maintains the smaller amount of D.P. Countermeasure to power attack introduced by introducing the noise and masking. Attacker target the leaking point on the SBOX to collect power traces. The success rate to correctly guess the key is minimized by parallelizing the SBOX, align power trace, and increase noise exists. Here the key is recovered from 8 SBOX with a success rate of (1/2)9 to (1/2)51. So, first, SBOX is more susceptible; to attack while last SBOX has a lower value of success rate.

In paper [62] by Siva Nishok Dhanuskodi et al. (2016), a more vigorous leakage power analysis attack depends on the internal state of the circuit presented. LPA ate susceptible is dual-rail precharge side-channel resistant logic. Since SBOX architecture is available in the public domain, a leakage current is predictable. A statistic method to compute leakage current based on the number of transistors on and off. For any guess, if the critical internal state of the circuit calculated and flow predicted. Find the Pearson correlation coefficient between predicted current and measured current depend on the hamming weight (H.W.) power model. The correct key has a high value of the Pearson correlation coefficient. Effect of process variation and temperature achieve a 59% success rate of key recovery at 300MHz with 1000 samples.

In paper [63] by Craig Teegarden et al. (2010) ROM based SBOX implementation is dominants on the subject of area and delay. Leakage power is the source of the side-channel attack, where the attacker tries to find out the correlation between the variation of power requirements and input data processed. Combinational logic like WDDL and TDPL try to harder the power consumption; thus, relationship reduces. SBOX implementation with WDDL increase area by 3.1 times and power 3.7 times compare to standard CMOS cell but reduce the throughput of the system by 3.8 times. Here switched capacitor current equalization draws an almost equal amount of current for all input data.

It has found that SBOX implementation with switched current logic show 1.07time area overhead, power consumption increases by 1.33 times, and reduces system throughput by two times.

In paper [64] by Halak, B., Murphy et al. (2015) power balanced circuit presented as a leakage power attack resilient design. Static power consumption led to static power analysis known as leakage power analysis. Input with high hamming weight shows the high-power use and low power consumption for lower hamming weight. Power balanced logic is a technique to exhibit equal power value-form out on n input data. Leakage current at CMOS 90nm technology classified as subthreshold leakage, gate leakage, and band to band tunnelling. Here the author has implemented a leakage power differential attack that relies on the hamming weight power model. To design secure logic design, leakage power dependence of input pattern of AND, OR, and XOR gate is an experiment for input pattern (0101, 0110, 1001, and 1010). Maximum leakage power reported for input "0101" and minimum leakage power for input "0010". The presence of a high bit on MSB position shows higher leakage power. Pearson correlation coefficient is more significant for the OR gate and minimum for the XOR gate. Change of temperature doesn't show much effect over the coefficient of variation for power balanced logic.

In paper [65], Adiabatic logic-based power attack resistant logic presented by Cancio Monteiro et al. (2011). Masking preferred for power resistant crypto module. However, masking is not secure for all input and require a random bit for masking and unmasking purpose. This paper explores single-rail and dual-rail logic family for secure logic implementation in adiabatic logic. Adiabatic differential logic such as SABL and 2N-2N2P have complementary parallel structure follow differential pull up and differential pull-down CMOS inverter requires single clock input for precharge and evaluation. SPICE simulation result shows that XOR implementation of adiabatic differential logic possess NED 91.22% NSD 46.8% compare to 2N-2N2P NED 65.178% NSD 29.8% and SABL logic NED 0.79% NSD 0.329%. SABL and 2N-2N2P logic show great power attack resistant features compare to adiabatic dynamic logic.

In paper [66] by Tiri et al., the power consumption of CMOS circuits is free from the input processed data. In this paper, a CMOS dynamic and differential logic (DDL) in this paper, dynamic differential logic for secure crypto, is deiced from a power attack. Proposed logic shows power consumption independent of input data sequence and logic value. The author has experimented with AND-NAND and flip flop. Problem identified by dynamic logic, i.e., it masked the input value, energy dissipation independent of input while output node discharges. No difference identified during the 0-0 and 1-1 switching events; however, 0-1 and 1-0 always consume energy. Dynamic logic breaks the input data sequence, and everyone consumes while capacitor charge 0-1 transition. SABL logic follows the properties of both.; upper half act as dynamic logic gate and lower half are differential pull down. The security measure of SABL SBOX is NED and NSD. SABL SBOX achieves NED 0.032 and NSD 0.006 at the cost of area 38541μm^2.

In paper [67] by Popp, T., & Mangard, S. (2005), side-channel attack resistant cells for SBOX implementation and energy consumption is independent of input data. Asynchronous logic implement device was resistant to side-channel attack. The probability model of switching of data and their energy dissipation on each switching presented, energy difference to pull up output node high is greater than the pull-down. A side-channel resistant cell tried to achieve this difference to a low value. The problem in the masked cell is glitches which encountered through MDPL logic. Transition at a different moment (glitches) reduces the attack resistant feature of a masked cell. The difference in energy required during the change of input without and with a masked cell compared to exhibit attack resistant analysis. MDPL logic every signal is masked with the same mask bit using the XOR gate. Compare to CMOS logic, MDPL cell-based AES implementation has area overhead 4.54 times and speeds 0.58 times.

In paper [68] by Fournier et al. (2003), Data-dependent switching activity is the origin of side-channel analysis to collect many power trance and study statically. Electromagnetic analysis, E.M. wave emitted through the active component. Like power attack, E.M. wave received and try to find the statistical correlation between exploited magnitude and hamming weight of input data to be processed. Fault injection analysis is an invasive

method to guess the secret by inserting an optical probe or introducing the external glitch. Change in temperature noted while performing the computation. Switching activity is a good approximation on DPA and DEMA attack, thus requiring if the countermeasure technique focused on this paper. A balanced asynchronous circuit is a superior replacement for the conventional synchronous circuit. Design time security analysis measures the fault possibilities with a wide range of input data.

In paper [69] by Mentens et al. (2003), security should include in the circuit during VLSI design at various levels. The requirement of the design flow of a secure side-channel attack. Design flow starts with VHDL modeling and lasts with attack resistant layout. The experimental result shows that the DPA attack on a CMOS standard cell security can break with 200 power traces. In contrast, DES implementations with stack resistant cells cannot recover hidden kay over 2000 power traces. Scope of other attack are; timing attack based on the arrival time of ciphertext may determine the operation that depends on the secret key. SCA resistant design start with standard design starts Verilog HDL based on digital design. The electromagnetic analysis is similar to the power analysis attack instead of electromagnetic power wave is analyzed. Differential fault attack attacker injected error into the circuit internally and exploited the weakness of the algorithm.

In paper [70] by Khan et al. (2003), a core building blocks AES algorithms identified as SBOX; it includes the nonlinearity in the computation. Nabihah Ahmad and S.M. Rezaul Hasan (2013) have implemented a novel XOR used to implement SBOX and inv SBOX. Internal block of SBOX with composite filed arithmetic in GF2. All brick is presented with a Boolean equation and presented with a logic gate. Logic gate count of SBOX is 158 and transistor requirement in 948 with delay 7.32nns and achieve throughput in 130MBPS at 00.8v supply voltage. The proposed XOR gate in Galois filed arithmetic requires symmetric coupled inverter pain and additional not gate design with CMOS 65nm technology. Proposed 6T XOR gate achieve PDP is 23.453 YJ at 1v supply. The simulation result shows that the SBOX delay is 7.322 ns, PDP 0.659 fJ, and throughput 1 GBPS at 0.8v supply.

In paper [71] by Franc¸ois-Xavier Standaert(2009), a different type of cryptanalysis is the review. Their effectiveness is measured where adversaries try to take advantage of the physical specification of actual information to gain access over hardware. Here adversaries gain access by analyzing hardware leakage information instead of a cryptographic algorithm. Have discussed primary requirements that are collected power attack from the active area of the algorithm during computation. Invasive or active attack repackage the chip puts external wire or connection and collect trace but physically damage the chip. Non-Invasive or passive attacks exploit externally available information. Power analysis attack (DPA, SPA) is a widely studied attack over DES and AES. Countermeasures to power attacks at the cell level are dynamic differential CMOS logic. At algorithmic level time randomization, encryption of bus a, hiding leakage current or adding noise during computations.

In paper [72] by Eman Mohammed Mahmoud et al. (2013), AES128 is designed with hardware description language, the secret kay generated with pseudo-noise sequence generator, and apply a dynamic permutation-combination with SBOX. Permutation bloc used to produce two sequences; these sequences arrange the row and column of SBOX. Key dependent SBOX based in permuted sequence, the first sequence arranged the column, and the second sequence arranges the row. The security measure of designed SBOX is measure with randomness, avalanche effect, correlation factor, and simulation time. Avalanche effect lies between 41% - 61%; the correlation factor is between -0.3 to +0.3; the difference in simulation time between AES and DES is 0.0031s.

In paper [73] by Pammu et al. (2016) have compared two different implementations of SBOX are presented with a lookup table and on the fly with Xilinx XCV100e-8bg560 and XC2VP20-7fg676 FPGA. LUT SBOX requires 7215 logic gates while the same can implement in on the fly SBOX with 4752 logic gates. SubByte is a non-linear substitution byte work independent on every byte and realizes as the matrix. SubByte follows the two operations on input data are multiplicative inverse and affine transform. Throughput calculated as a function of the number of generated bits, clock frequency, and a large number of clock cycles required to create bits. Power requirement to implement LUT

SBOX is 2.229mW, and on the fly, SBOX is 2.849mW, the chip is of LUT SBOX is 1.4-time larger than on the fly SBOX.

In paper [74] by Dey et al. (2018), the vulnerability of hardware and software explained to analyzed the cryptographic algorithm on an 8-bit Atmel microcontroller described by Han Yu et al. (2018). The simulation-based experimental environment created to collect the power trace at SBOX output to examine the 1-bit DPA and CPA attack. Value changes dump (VCD) file stores the power consumption corresponding to input transition. Data analysis methodology shows the CPA attack takes correlation -03 to 0.3 at correct data points. An improved DPA attack takes a couple of SBOX, difference actual power requirement correlated with hypothetical hamming power model required 5120 power trace to return the secret key. The computational complexity of CMOS is large compare to DPA is medium, and improved DPA is small.

In paper [75] by Kazuyuki Tanimura and Nikil D.Dutt(2012), differential power analysis(DPA) based on a correlation between power trace at the output of SBOX and hypothetical power model. Shows WDDL cell logic assures a 100% switching factor to estimate the information of power requirement. This paper proposed homogenous dual-rail logic to implement attack resistant cells. HDRL shows better attacks resistant; it wisely combines the Vss current wave and suppresses the differential power curve. 128-bit AES circuit with 16-SBOX is evaluated with HDRL since SBOX consumes 75% of AES power itself. It can minimize the differential power to zero with a hypothetical power computation model. HDRL shows 200% area overhead and energy to SBOX implementation. HDRL logic implements higher security with 100% energy overhead estimates with WDDL has 2371.7% energy overhead. HDRL doesn't need an overhead delay with a smaller area. Thus, HDRL promises SCA resistant cells.

2.6 Research Gap

Mathematically obtained random secret keys are not genuinely random. If the initial seed is known or a sequence of response patterns are known, predicting the upcoming

sequence is possible. Statistical measure weakens the randomness of the secret key. These are not passing in standard statically test for randomness, i.e., National Institute of Standard Technology (NIST) statically test [12] or Diehard test [13]. Certain unclonable features must include in computation to generate a true random response. The generate response should unpredictable; even response patterns are known. PUF utilizes the physical variable properties of an integrated circuit (I.C.) to generate an unclonable secret key. Storing the key in non-volatile memory is not a vital choice. The leakage current of memory cells enables attackers to predict the hidden information and side-channel attack (information leaks in the form of power, radiation, and timing) breaks the software and protocol-based security mechanism [27]. PUF does not store the secret key; it generates the critical run time. A list of challenge-response pairs stored in a secure environment, during the authentication key is created even in an insecure environment for a particular challenge current response approximately matches to stored response secured authentication succeed. All the PUF architecture is not unclonable; many have been prone to attack. A research team from the Berlin Institute of Technology had cloned a static random-access memory (SRAM) PUF within 20 hours using tools failure analysis labs [1]. PUF is vulnerable to side-channel attack offer entry point for hacking into cryptography. Existing MuxPUF and ROPUF suffered from low response generation rates and complex structures, respectively. New properties of CMOS devices need to explored to have a news structure of PUF that enhances performance [5].

CMOS based device leaves their input signature over power consumption, S-Box is the significant component of the encryption module. It consumes 75% of the total power [14]. The dependency of power consumption with processed data opens the door of the side-channel attack (SCA). Recent development has identified that; hidden keys can obtain by analyzing side-channel information by statistical method. Power-Data independency measured in terms of normalized energy density (NED) and normalized standard deviation (NSD). Hybrid CMOS logic shows the power consumption pattern should be independent of the input pattern. Power attack is a methodology to obtain the hidden secret key from cryptographic key while in computes, and by statically analyzing

the power trace. Countermeasure to power attack is hiding and masking, which modifies the cell design or alters the algorithm. Existing masking countermeasure masks the input of SBOX, compute with masked value and unmask the result at the output end. The finite state machine (FSM) controller requires to generate a mask pattern [57]. Masking specific input bit and presenting the same power consumption for all input vectors are preferred methods to design SBOX, complicate the prediction algorithm in a side-channel attack, and preserve the key safe [64]. Existing countermeasures enhances the area and power requirement, highlights the minimization of gate counts.

2.7 Hypothesis

1) Two silicon-based CMOS circuits do not have the same properties in all aspects: process variation and mismatch result in contrast.
2) A CMOS circuit consumes power only during 0-1 and 1-1 transition, and no power consumption occurs during 0-0 and 1-0 development.
3) The power trace is glitch-free.

2.8 Objectives

1) Design of PUF based challenge-response pair generator circuit using CMOS.
2) Testing & validation of generated response.
3) Design of S-Box and analysis of its power pattern independence from the input pattern.
4) Design of power analysis attack model using svm to measure unpredictability.

Chapter 3
Physical Unclonable Function (PUF)

3.1 Introduction

VLSI IC characterized by three-parameter area, power, and delay. The emerging technology adds 4^{th}-factor security during design. Weather IC used by an authentic one? Is IC working being reliable? To answer these, require sophisticated cryptographic algorithm, which increases area overhead and vulnerability due to a side-channel attack. PUF is a prominent hardware module discussed in [17], which includes security while designing the ICs. Any electronics properties, which are variable and lasts with a stable state, can use to turn the device into a PUF-circuit. Before creating a computational PUF circuit, the need to identify a unique feature, the output of the course depends on applied challenge input and individual behaviour of the circuit which makes PUF response unpredictable shown in [22]. Security and privacy importance in our life, while we are communicating with some person or interacting with the system, we put a lot of trust without knowing trustworthiness. People rely upon system and software protocol for secured communication. PUF application increases widespread in a digital transaction of smart card and secured authentication like RFID tags. The security of smart cards is required greater context in the context of the application. The smart card should perform reliable authentication and secure communication between devices. Cryptography is a branch of computer engineering where information kept secured via encryption and decryption. The intelligent application requires a secured cryptographic module.

Secured authentication and random number generation analyzed in [26] are two significant areas where security requires maximum and suspect to vulnerable attack. The cryptographic protocol requires complex mathematical functions for the production of random numbers. Hardware implementation of such a complex task consumes considerable power that can be vulnerable. Random number neither computed nor guessed. The software mechanism to generate random numbers is not entirely random. Pseudorandom noise generator (PRNG) shown in [76] produces a random number based

on initial seed value and XOR in the feedback path. The PRNG sequence is not entirely random. The upcoming series based on the previous sequence; if a set of the series is known, future random sequences can be predicted statistically. A true random number generator (TRNG) preferred over an ordinary random number generator which uses to derive secret keys during authentication or encryption. TRNG shown in [27, 31] mixes hardware feature during computation such that output sequence derived from the initial value and hardware feature, to guess the output attacked should know the selected hardware feature. Very first John von Neumann find that computers cannot produce truly random numbers and stated

"Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."

In the digital world, we rely on a password to access the system; it is a challenging task to generate passwords in a secure environment. Generally, non-volatile memory like EEPROM or SRAM integrated with the method used to store keys shown in [77]. When the user's current key matched with secret key user authenticated to use the system, however, this approach is expensive, requires more extensive and consume more power, are requires non-volatile memory vulnerable to side-channel attack. The presence of non-volatile memory intricates the system and comes with a cost; leakage current continuously leaks the information. Adversary utilizes leakage to obtain hidden information during side-channel information. Since the last decade, hardware-based security primitive widely studied, whose response is a function of inherent hardware properties. Integrated circuit-based electronics circuits attract all due to its physical variability nature [32, 78]. Two silicon IC's do not have similar features; process variation includes variability into parameters during manufacturing can analyze with corner analysis. Physical unclonable function (PUF) is the innovative circuit extract secret from the physical circuit feature of integrated circuits. PUF is a hardware security module that finds its application as low-cost authentication and resistance to security attack. In contrast to a classical method where secrets stored in memory PUF compute secret run time based on physical characteristics of which is unpredictable [79].

3.2 Application of PUF

The author in [80] has proposed PUF two niche applications are low-cost authentication and cryptographic key generation.

3.2.1 PUF based Authentication

Figure 3.1 shows the schema for secured authentication;1st phase of authentication is a large set of challenge-response pairs database creation, and 2nd phase is authentication. Each PUF provides a unique response for each challenge based on its non-linear behaviour. An adversary can only record a PUF output and compared it with regenerated one [81], but he does not have information, exactly which hardware feature selected. If he tries to clone the ICs, he must have to store all CRP pair in the memory.

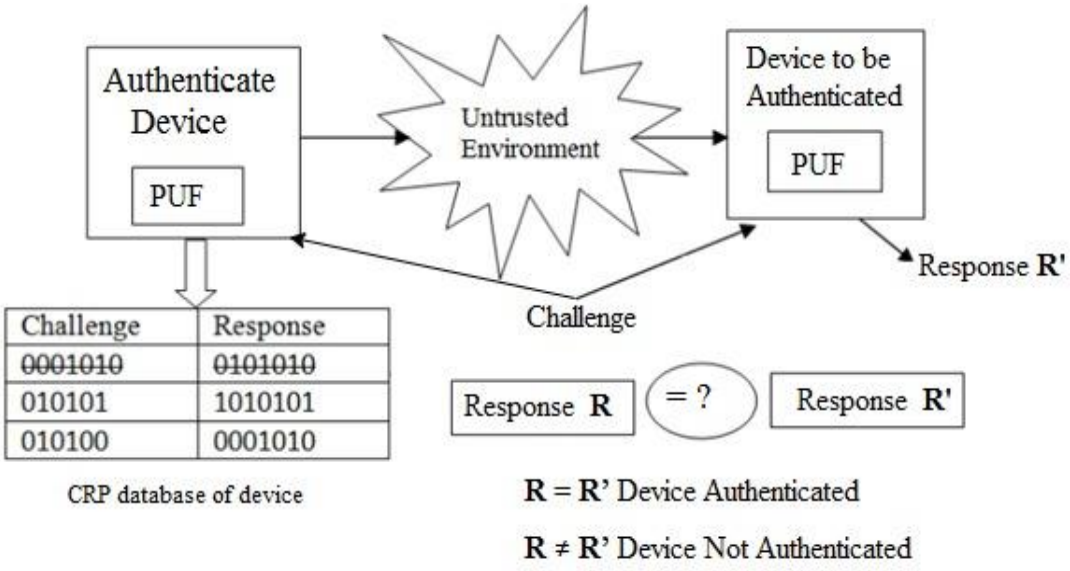

Figure 3.1 PUF based Authentication [17]

During manufacturing, a large number of CRP pairs created in a trusted environment and stored internal secret storage. PUF is given to the client. Anyone tries to authenticate even in the un-trusted environment; applies challenge to PUF. If the observed response is matched with response stored in the databased for approximately matched with the stored response; IC authenticated. Authentication is successful because of the authentic IC and

trusted database response match; otherwise, it fails. To avoid man in middle attack used pair must be erased after authentication.

3.2.2 Cryptographic Key generation

With the advancement of electronic devices, we are progressively reliant on IC to deal with sensitive data safely; for example, smart cards are utilized to perform a financial transaction. Significantly, IC ensures the protection of sensitive data. The conventional method is to store a key in non-volatile memory (EEPROM). The drawback of storing the key in memory is invasive and non-invasive attack allows the attacker to extract the stored key and compromise the cryptographically secured mechanism. PUF is security primitive doesn't store key; it produces the key. Due to noise and error-prone nature PUF, the response cannot be utilized for secret keys. ECC reconciles error in the response. PUF based key generation has two-phase mechanism Key generation and key extraction, as shown in figure 3.2 discussed in [82, 83].

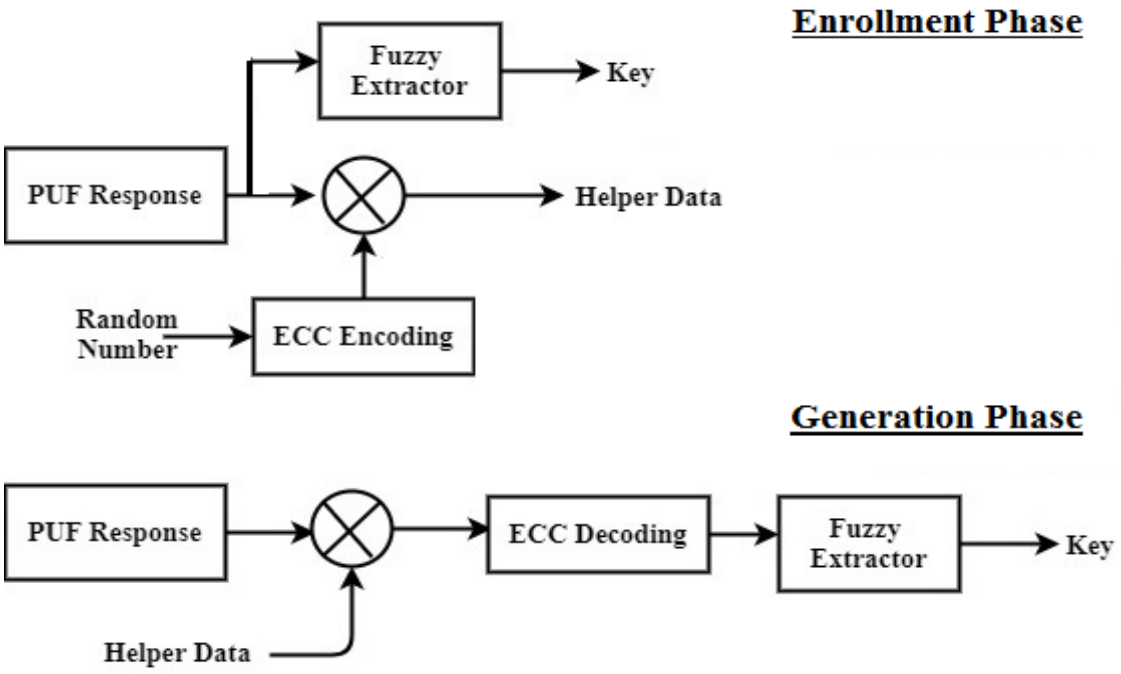

Figure 3.2 PUF based key generation [80]

During the enrollment phase, ECC encoder syndrome and PUF's response are XORed, produce helper data stored in the memory. In the key generation phase, similar PUF response (by applying the same challenge during enrollment) is XOR with helper data followed by ECC decoder extract the same key. In this way not required to store secret key, only helper data need to store. Even helper data is communicated public key has perfect secret utilizing storing secrets in its unique physical features.

3.3 Classification of PUF

PUF is a hardware security module whose output response is a complex function of challenge input and unique physical properties. It finds application in random secured authentication and cryptographic key generation. The expectation from PUF discussed in [35, 84] are (i) Responses generated from different FUF must different for the same challenge input, (ii) Response for each channel should consider unique, and (iii) A response should not deviate due to noise and environmental condition. PUF circuit is categorized into strong and weak PUF. Identification of unique features [85] makes PUF different from others; according to [86] three categories of PUF are (i) delay-based PUF (ii) frequency variation PUF and (iii) initial value of SRAM PUF. Table 3.1 shows the classification of PUF as strong and weak PUF.

Table 3.1Strong vs. Weak PUF [84]

Weak PUF	Strong PUF
Requires a lesser number of CR pair	Requires a large number of CR pairs
The response is unaffected from noise	The response is stable in noise
The response is strongly depending on intrinsic variability	Responses do not maintain the correlation with a previous response
Response processes through the error-correcting core	Error-correcting core not required
Output response must preserve	Preserving of response not required
It is susceptible to attack	Not susceptible to attack
Example are SRAM PUF	Examples are Arbiter and RO PUF

3.3.1 Delay based Arbiter PUF

Delay based PUF is based on random delay row (path) implemented with 2:1 multiplexers, and a decision device usually arbiters or D-latch or XOR gate. In this PUF circuit, output Y is a function of challenge input C and delay associated with AND-NOT-OR cell. Delay of multiplexer path due to selection input high is lower than selection input low, due to additional requirement of NOT cell [87]. Figure 3.3 shows the input signal running through the parallel row with the application of challenge input at select. There is the race for the rising input information between the upper and lower path to reach a destination. [88] shows a pair of the multiplexer directed by the same input C[i]. When C[i] is low, multiplexers select the upper path in the upper row and lower path in a lower row; for the high value of C[i] lower path in the upper row and upper path in the lower row. Internal cell induces a different pair of delay for each input C[i]. Challenge input C switches the path multiple times, provides a different delay path. Since delay is additive at arbiter end upper and lower path has a different delay. Arbiter decides with the path is faster, the output is one if upper path faster otherwise zero.

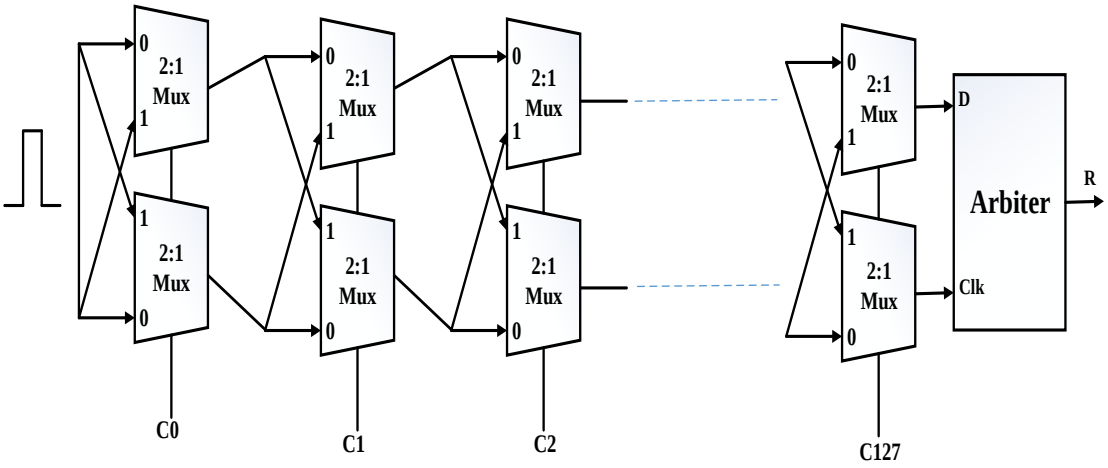

Figure 3.3Arbiter PUF

The multiplexer is a delay module consists of two internal paths with different delays shown in [37]. Let multiplexer delay in upper path a_i and b_i. Let $H_i = (a_i + b_i)/2$ and $y_i – (a_i-b_i)/2$ signal going through upper path is delayed by $H_i + (–1)^{ci}y_i$ assuming there are total n/2 stage than an overall delay in upper path calculated via equation (3.1)

$$D_H = \sum_{i}^{n/2} H_i + (-1)^{C_i} Y_i \quad (3.1)$$

Similarly, delay of the lower path in each mux is d_i and f_i Let $L_i = (d_i+f_i)/2$ and $u_i = (d_i-f_i)/2$ signal going through the lower stage is delayed by $L_i + (-1)^{ci}u_i$ Therefore, the total delay in the lower path calculated via equation (3.2)

$$D_L = \sum_{i}^{n/2} L_i + (-1)^{C_i} u_i \quad (3.2)$$

A signal traveling through the upper and lower path will interact decision device at the end. Condition to determine output bit response given in equation (3.3)

$$D_H < D_L \rightarrow R=1 \quad (3.3)$$
$$D_H > D_L \rightarrow R=0$$

Limitation of Arbiter PUF is it requires a long chain of cascaded structure and accurate decision device. The response generation rate is very low. Authors in [25, 39] have analyzed Slender-PUF with the non-linear arbiter, DFF, and XOR gate has obtained a random set of responses. LOOP-PUF is an alternative method of using the non-linear delay stage [88] to have high randomness. The width of the response bit requires the number of parallel paths, with multiple arbiters acts parallel. An alternative way to design delay PUF discussed in [89] a current starved inverter chain with a control element sum up with a different delay. Arbiter continuously produces a response bit based on the arrival time of a parallel path. PUF with non-linear delay implemented in [90] as a Feed-forward cascade and Feed-forward overlap structure to provide reconfigurability.

3.3.2 Frequency variation-based RO PUF

Ring oscillator (RO) based PUF circuit requires a large number of the oscillator. An odd number of delay stages produces frequency $f= 1/NT_d$; oscillation frequency deviates from its deal frequency ($f\pm\Delta f$) due to manufacturing variation (at the different corner) shown in [19]. ROPUF with frequency variation is presented in [91] figure 3.4, a group of N frequency whose active edge occurring at a different instance of time. A fixed pair of frequency chosen with large size of the multiplexer. Challenge input at the selection line

selects two particular frequencies allow two independent counters to start upward count [92, 93]. Select two particular frequency f_1 and f_2 based on challenge input; frequency is slightly different result into the counter counts varies. Size of the counter should keep large to have sufficient entropy; after counting many cycle differences in the count value amplified, if the upper counter is faster output, a response is high otherwise low [30].

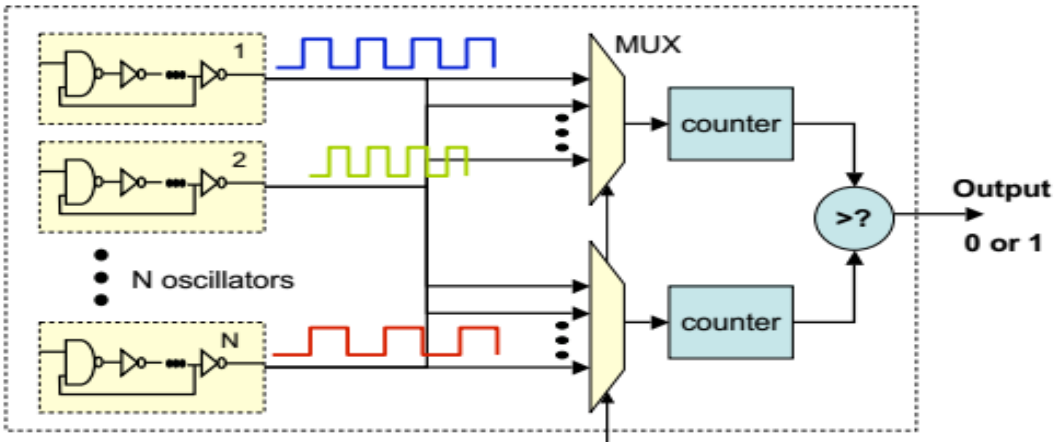

Figure 3.4 RO PUF [90]

PUF suffers from the limitation of a large number of ring oscillators leads to more power consumption, the large size of the multiplexer and a large number of frequencies for counting purposes to have a fair comparison. The entropy of RO-PUF depends on the random selection of oscillation frequency, [21] has analyzed the transient effect based ring oscillator (TERO) measuring the difference in the number of transient oscillations per seconds between two identical oscillators. Two obtain multiple responses in comparator is replaced by two multi-output functions, i.e., adder, subtractor, etc.

3.3.3 Initial value-based SRAM PUF

An SRAM cell comprises two CMOS inverter connected back to back controlled by access transistor. SRAM cell has two stable states Q and Q_b. During the write operation, if data bit is zero Q transition to low and Q_b transition to high and vice-versa. When SRAM is powered up, and no data write procedure performed cell enters into a metastable state. If all-transistor is identical, (no mismatch), the cell remains in metastable indefinitely [95-97]. However, in actual implementation, there must exist a threshold voltage mismatch in the

transistor due to process variation [29, 96]. Noise in the circuit triggers the feedback loop and force the state to enter in a high or low state; the final state is cell is non-deterministic due process variation induces fluctuation into output bit. SRAMPUF in figure 3.5 [98] suffers from the limitation of only those SRAM cell selected that provides stable behavior, Exhibit 10% error.

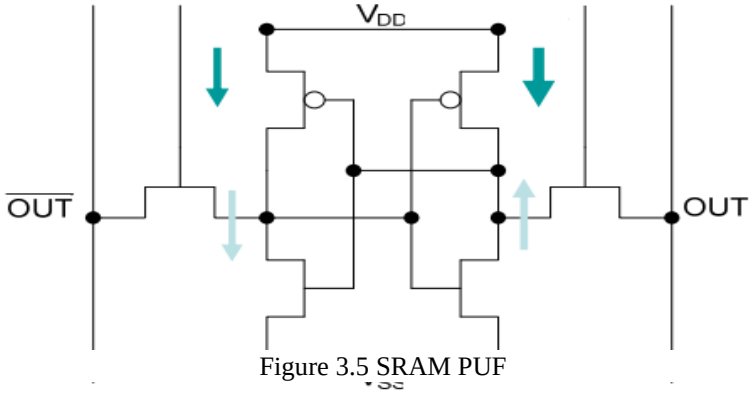

Figure 3.5 SRAM PUF

3.4 Schmitt trigger PUF

Schmitt trigger is one of the electronics circuits generally used for wave shaping circuits. A Schmitt trigger circuit contains two threshold voltage. A variable input signal compared twice in one time period known as the upper and lower threshold voltage. The significant difference between the comparator and trigger circuit are multiple threshold points; a comparator circuit has only one threshold point, while Schmitt trigger possesses two threshold points to compare the analog signal [99]. Schmitt triggers show the different comparison points for rising and falling input. Hysteresis of the trigger circuit eliminates the noise generation and produces a more consistent signal. The output of Schmitt trigger varies their state when a rising input crosses the upper threshold voltage (UTP), and a falling input crosses the lower threshold point (LTP) voltage [100]. Hysteresis width measured as the voltage difference between UTP and LTP. Hysteresis width of the Schmitt trigger is a unique feature, possesses a fixed value. Schmitt triggers feature doesn't include the circuit behaviour near transition point; If noise signal is lesser, the switching voltage Schmitt trigger will not respond; hysteresis reduces the sensitivity to noise and disturbance. In this work, the threshold voltage of Schmitt trigger and hysteresis used to generate

Schmitt trigger PUF circuit response. Schmitt trigger has a fast transition time compared to the conventional comparator circuit, besides this circuit finds application to derive the load with high-speed switching, low power dissipation, and small DC supply voltage [101]. The present work of Schmitt triggers designed with a Cadence virtuoso schematic composer with CMOS 90nm technology at 1V supply voltage. The conventional Schmitt trigger circuit shown in Figure 3.6(a) contains 3 NMOS and 3 PMOS transistors. Schmitt triggers the circuit is designed with a double transistor inverter. The threshold voltage of the trigger circuit set by varying the width ratio of PMOS and NMOS. The threshold of NM1 and PM3 is set higher than NM0 and PM4, and additional feedback transistor PM2 and NM2 provide hysteresis width.

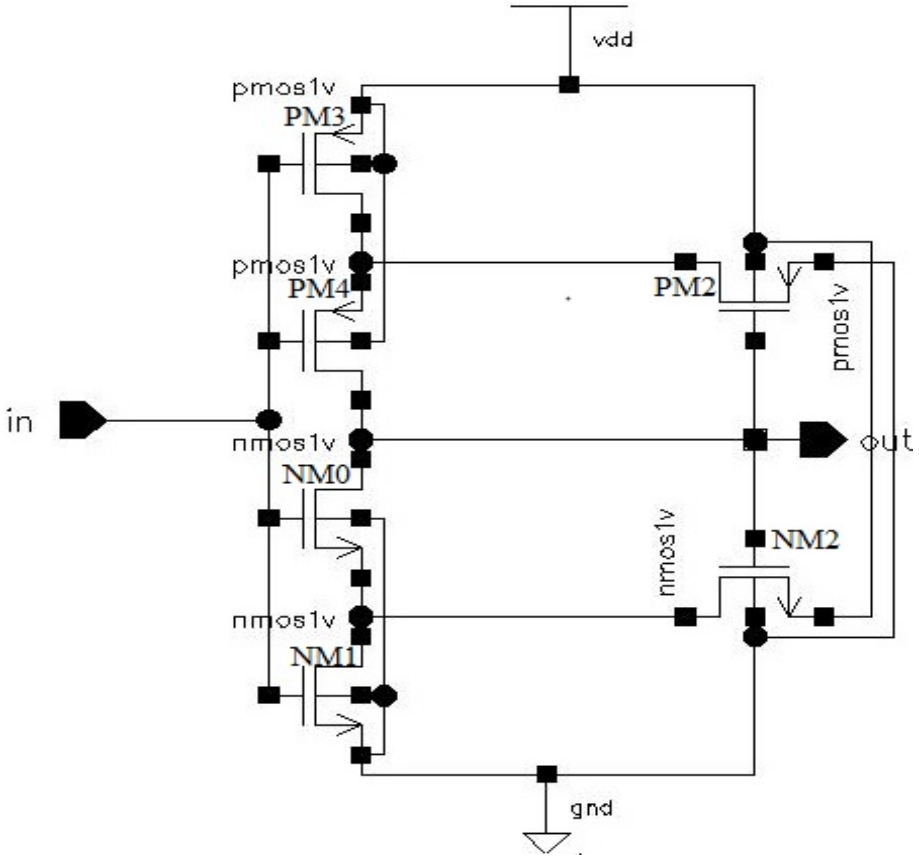

Figure 3.6 (a) STPUF circuit

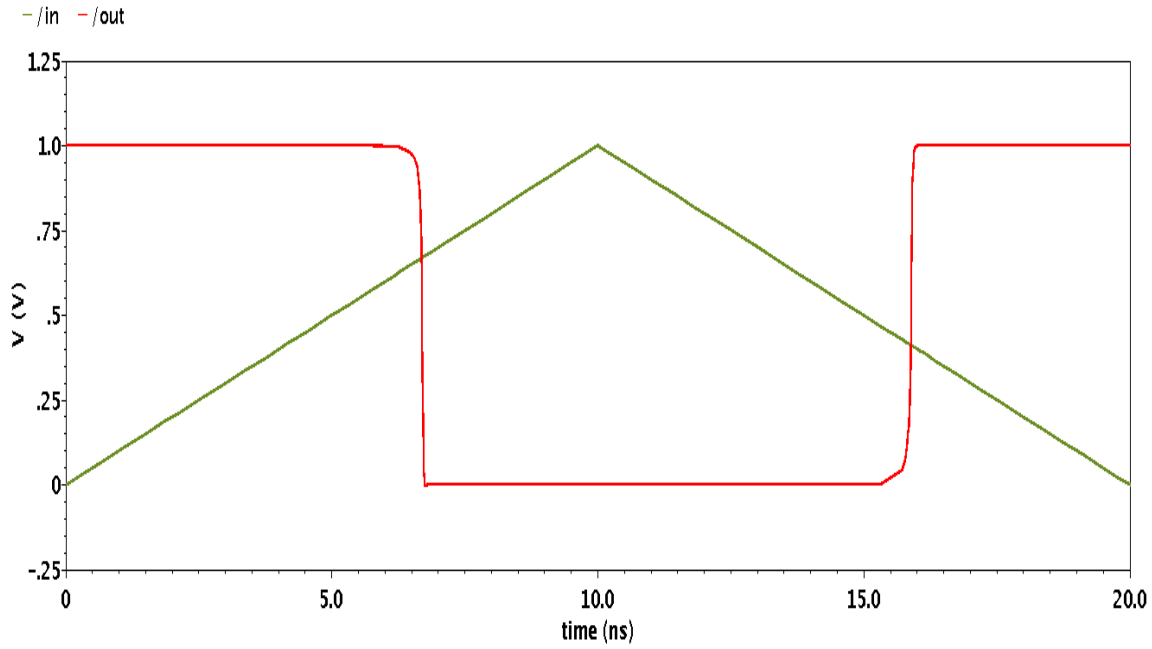

Figure3.6 (b)Schmitt trigger transient response

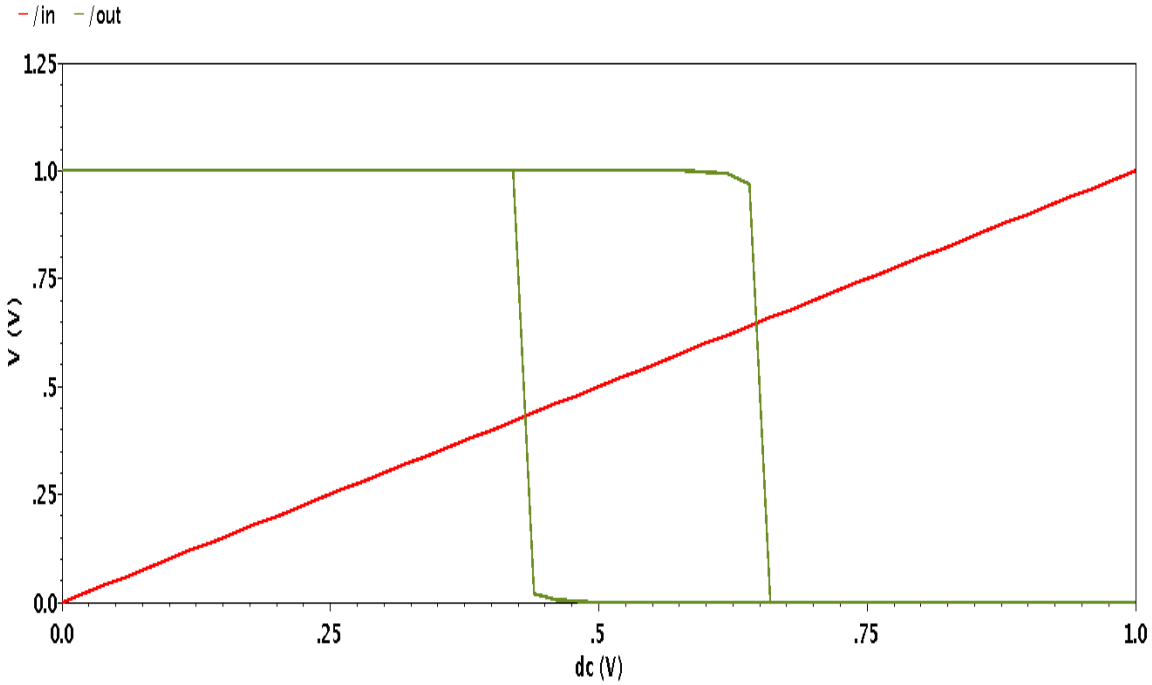

Figure 3.6 (c) Schmitt trigger DC response

The hysteresis width adjusted by varying the W/L ratio of the transistor. When the input voltage $V_{in} < V_{th}$, transistor NM0 and NM1 are OFF and PM3, PM4 is ON, pull the output

to the high level, turn ON the feedback transistor NM2. Forces the output level to pull down by NM0 and pull up by NM2. As V_{in} increase near V_{th} NM1 turn ON output level pulls down to V_{DD}-V_{th} give lower switching point V_{LPT} Similarly for input $V_{in} > V_{th}$; NM0 and NM1 turn ON, PM3, and PM4 turn OFF force the feedback transistor PM2 in ON state. When V_{in} falls around the threshold of PM4, PM4 turns ON, and PM3 turns OFF. Output level pulls up by PM4 and pull down by PM2. Output level maintained contains to V_{th} low level. When input V_{in} approaches the threshold voltage (V_{th}) of PM3, it pulls up the output provides upper threshold point V_{UTP}. The simulation result shows that V_{LTP} and V_{UTP} adjusted to 15% of the supply voltage. Figure 3.6(b) shows the transient response of varying signals from one stable point to others when a Schmitt trigger operates in linear mode. Rising input sample at 0.742V and falling input trigger at 0.357V. Ramp input crosses 0.742V output falls too low, and input pulse falls beyond 0.357V output of ST switches low to high. Figure 3.6(c) presents the difference between V_{UTP} and V_{LTP} is hysteresis, which refers to the extra voltage added to the low logic level at the output or subtracted from a high level of output. Output changes from high to low NM0 in cutoff and NM1 and NM3 in saturation. Output switches from low to high PM4 operate in cutoff PM2 and PM3 in a saturation region. The threshold point of the Schmitt triggers calculated as equation (3.4) and (3.5).

$$V_{T+} = \frac{V_{DD} + V_{Tp} + \sqrt{\frac{\beta_n}{\beta_p}} V_{Tn}}{1 + \sqrt{\frac{\beta_n}{\beta_p}}} + \frac{\sqrt{\frac{\beta_n}{\beta_p}} (V_{DD} - V_{Tn})}{(1 + \sqrt{K_n})(1 + \sqrt{\frac{\beta_n}{\beta_p}})} \tag{3.4}$$

$$V_{T-} = \frac{V_{DD} + V_{Tp} + \sqrt{\frac{\beta_n}{\beta_p}} V_{Tn}}{1 + \sqrt{\frac{\beta_n}{\beta_p}}} - \frac{\sqrt{\frac{\beta_n}{\beta_p}} (V_{DD} + V_{Tp})}{(1 + \sqrt{K_p})(1 + \sqrt{\frac{\beta_n}{\beta_p}})} \tag{3.5}$$

Where V_{Tp} and V_{Tn} are the threshold voltage of NMOS and PMOS transistors, respectively [16]. Hysteresis width increases with the load capacitance. The hysteresis voltage is given by

$$V_H = \frac{1}{(1+\sqrt{\frac{\beta_n}{\beta_p}})}\left(\frac{\sqrt{\frac{\beta_n}{\beta_p}}(V_{DD}-V_{Tn})}{1+\sqrt{K_n}}+\frac{(V_{DD}+V_{Tp})}{1+\sqrt{K_p}}\right) \quad (3.6)$$

Here we have presented a delay & hysteresis based Schmitt trigger based PUF circuit; the architecture of STPUF is similar to delay-based arbiter PUF. Schmitt trigger controlled by a multiplexer substitute for each delay unit. A STPUF presented in figure 3.7 [21] contains two parallel paths; there is a race to arrive input analog signal to the far end(arbiter). In between the path input signal is triggered up and down multiple times. Multiplexer route the racing signal to upper and lower path controlled by input challenge. Each stage of Schmitt trigger compares with threshold voltage multiple time and adds delay into the input signal. The signal arriving at the arbiter end is a function of delay of Schmitt trigger stage and hysteresis width. Arbiter at the final end takes a decision on which path is faster if the upper path is arriving earlier generates a response bit high else low. An ST generates stable response beyond the switching point [102]; during hysteresis, it is non-deterministic. The generated response is a function of input challenge as well as delay and hysteresis width. It is the first time in literature where responses bit is the function of multiple hardware features.

$$R = f(C, Delay, Hystersis) \quad (3.7)$$

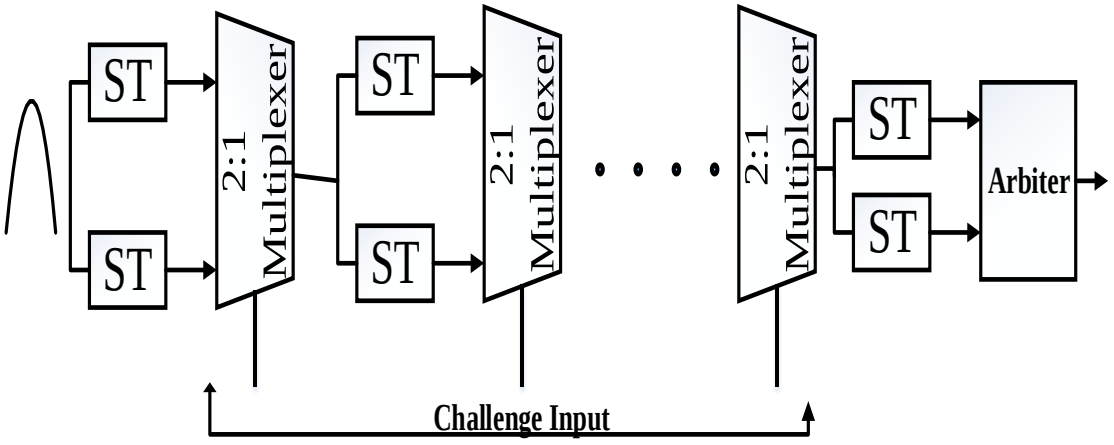

Figure 3.7 STPUF module

3.4.1 Different Topology of Schmitt Trigger

To enhance the entropy, eight different architecture of ST has used in between the path; each ST stage modulates incoming signals at different threshold levels and unusual delays. Each Schmitt trigger sample the input signal at the separate upper and lower trigger points. Since the delay and hysteresis width of each ST are different, each ST provides different trigger points for the same input signal. Eight different architectures of ST have utilized to design STPUF. Figure 3.8(a) present NOR based architecture [103] of Schmitt trigger (NORST) series NMOS are connected in parallel to form a NOR circuit. Figure 3.8(b) presents the dual threshold-based architecture of Schmitt trigger (DTST) in [104] offers a high threshold for UTP and low threshold for LTP. Figure3.8(c) present Hex architecture MC14584B of Schmitt trigger (HexST) finds primary usage for low power requirement and high noise immunity. Figure 3.8(d) presents self-adjusting hysteresis (SAHST) architecture shown in [107], where the feedback path into ST configured to determine the threshold voltage. Figure 3.8(e) offers Standard CMOS Schmitt trigger (STDST) circuits in [104] with controllable hysteresis Schmitt trigger with switching threshold adjustment. Two combinations of feedback devices to have UTP and LTP trigger edges following the preferred embodiment. Figure 3.8(f) present a LadderST [105] compensated ST architecture that provides a monotonic hysteresis response. Figure 3.8(g) introduces an SOI equivalent ST circuit [106]; it is 3 stage form of SOIST shown in Figure 3.8(e)[107].

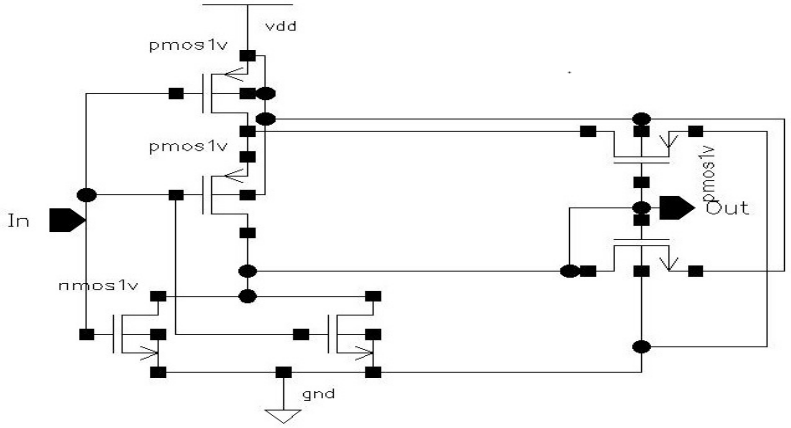

Figure 3.8 (a) NORST

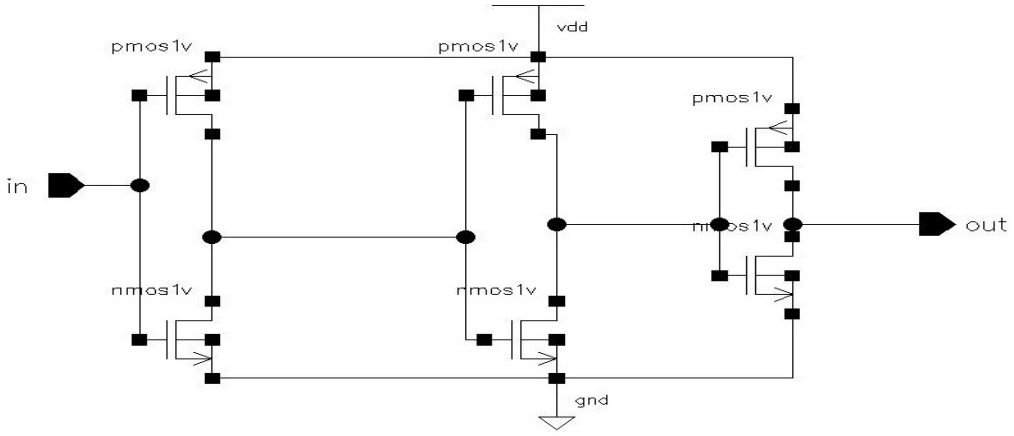

Figure 3.8 (b) DTST

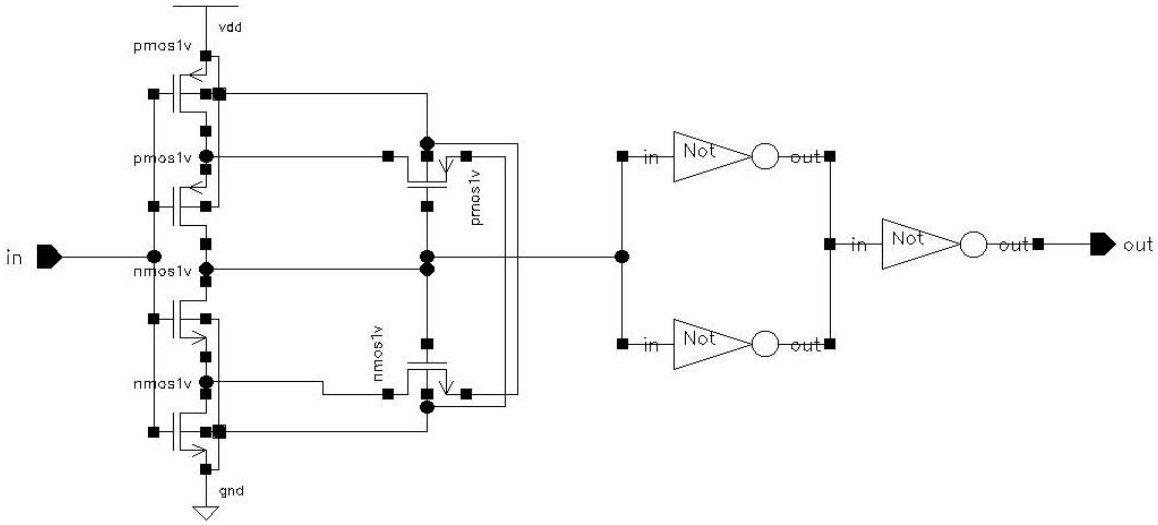

Figure 3.8 (c) HexST

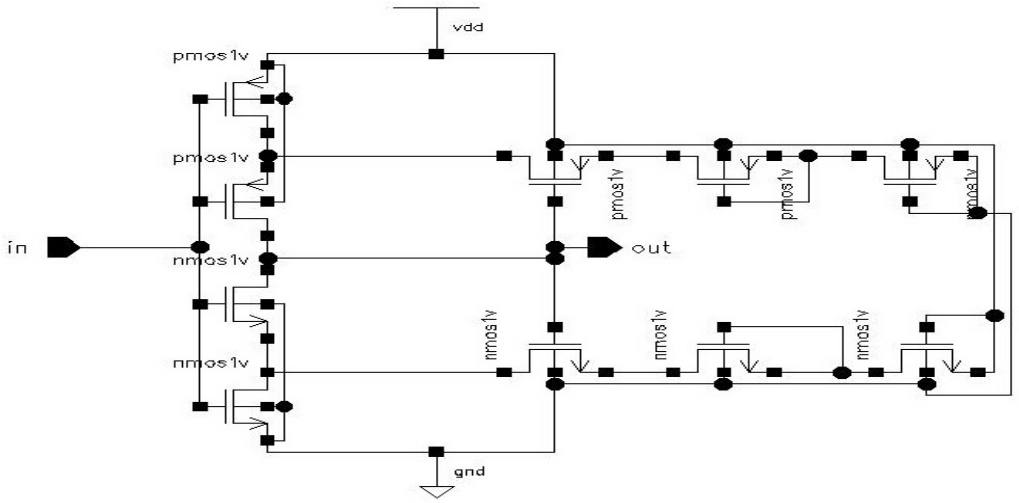

Figure 3.8 (d) SAHST

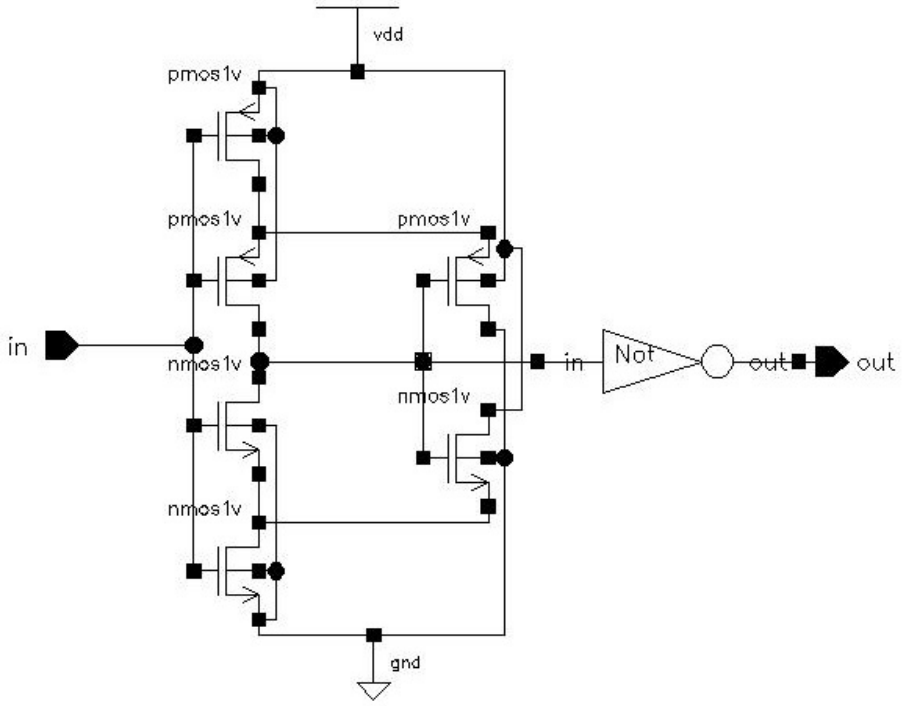

Figure 3.8 (e) STDST

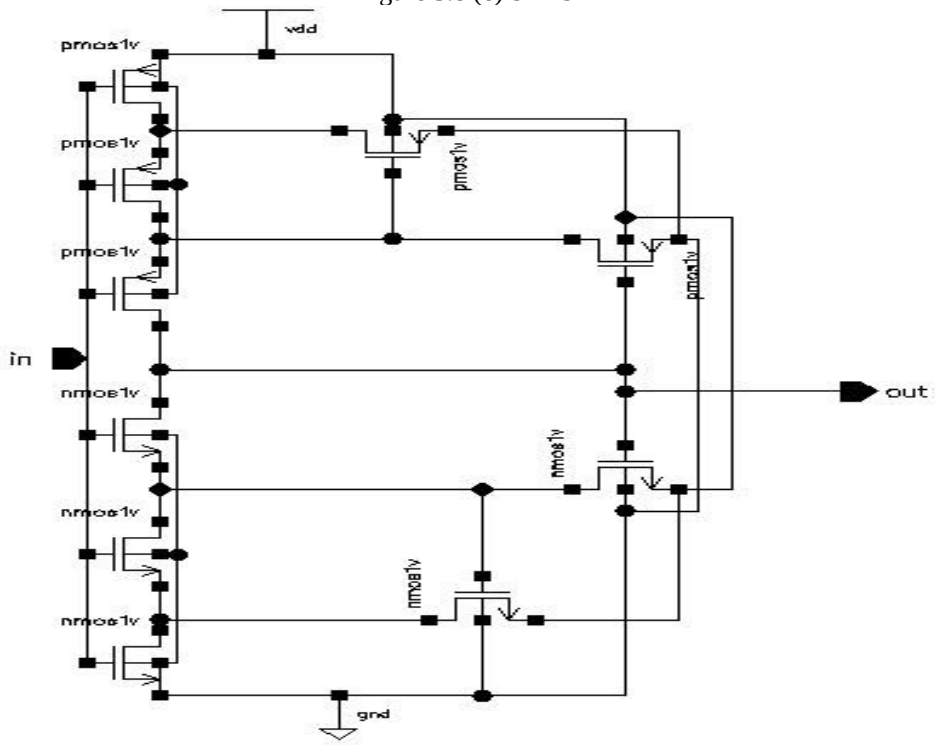

Figure 3.8 (f) LadderST

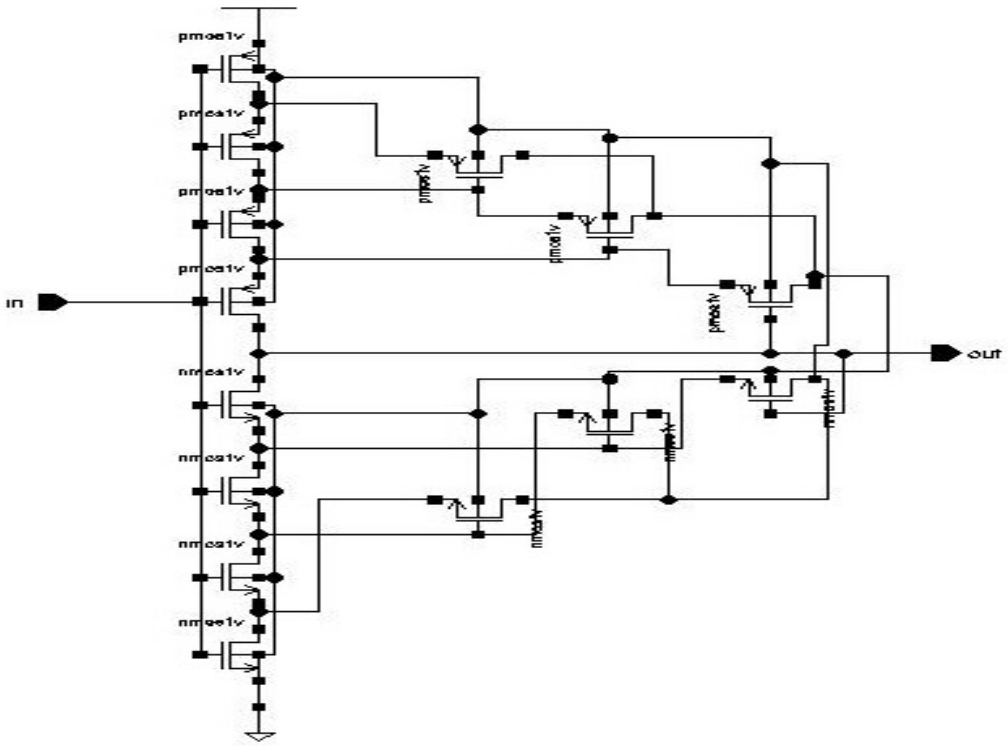

Figure 3.8 (g) SOIST

Table 3.2 Comparison of Schmitt Trigger

Sr No.	Architecture	LTP(v)	HTP(v)	Hysteresis (v)	Power (W)	Delay (s)
1	ST	0.357	0.742	0.385	1.869u	53.43p
2	DTST	0.42	0.6	0.234	9.976u	531p
3	NORST	0.286	0.311	0.025	1u	49.19p
4	HexST	0.358	0.592	0.234	3.47u	96.74p
5	SAHST	0.415	0.466	0.051	1.422u	45.97p
6	SOIST	0.28	0.6	0.32	2.31u	101.2p
7	STDST	0.368	0.6	0.232	2.407u	69.96p
8	LadderST	0.294	0.597	0.303	6.919u	139p

Table 3.2 lists a comparison of different architecture of Schmitt trigger concerning the trigger point, hysteresis width, power, and delay. Switching time (delay) and hysteresis width of Schmitt trigger resistance during pullup and pull down can reduce by the

addition of PMOS in parallel and NMOS in series. Delay of circuit intensity by addition of PMOS in series and NMOS in parallel. Since resistance increases by adding NMOS in series and PMOS in parallel minimizes resistance. NOR gate-based Schmitt triggers emphasis resistance, NORST offers higher delay. LTP is minimum in the case of NORST and maximum for SAHST, seed minimum, and maximum strength to rising input.

Similarly, for falling input, DTST/SOIST/STDST offers high HTP, and NORST has a lower value of HTP. Hex design of ST (HEXST) provides more full hysteresis width and estimated delay with ST and STDST. Silicon on Insulator design of Schmitt trigger (SOIST) available or use with two-level of feedback resistance offers better control of threshold voltage. SOIST possesses considerable delay and hysteresis width. CMOS SOIST circuit consists of 2 additional feedback for both V+ and V- trigger edges. SAHST increases the MOS in series in the feedback path to add more resistance (3 PMOS and 3 NMOS), has a configurable threshold voltage. Dual threshold ST circuit possesses three layers of inverting stage, here upper and lower threshold voltage determined by the inbuilt switching voltage of the inverting stage. Hysteresis width varies in a range of 0.025 for ST and 0.385V for LadderST. NORST samples the variable input to a small period rectangle pulse while LadderST results in a more significant period. ST, DTST, HEXST, STDST sample the information into a similar kind of rectangle width. Propagation delay attends a minimum value of 53.43ps for the ST circuit. DTST circuit determines the trigger voltage due to the inbuilt switching voltage of the inverting stage offers maximum speed 531ps. Power requirement is the cumulative effect of static and dynamic power, static power increase with many components available over-circuit and progressive power increase with switching activity. The power requirement of DTST consumes the highest to 9.976μW, and NORST consumes minimum to 1μW.

3.4.2 Arbiter

Farther end of PUF circuit consists of the arbiter; a decision device is shown in Figure 3.9(a) takes decision '1' if the upper path faster and '0' if the lower path is faster. It

compares the edge of the upper and path of PUF [87] whose arrive earlier, as discussed in Figure 3.9 (b).

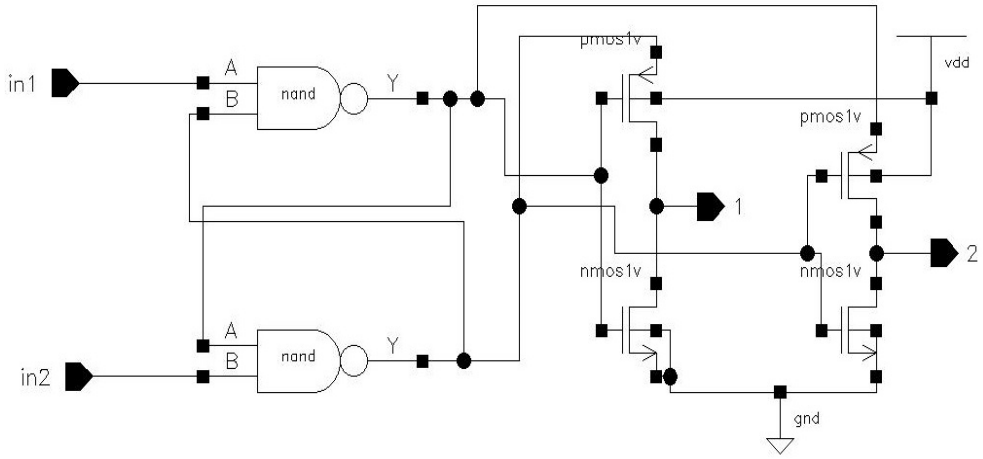

Figure 3.9 (a) Arbiter Circuit

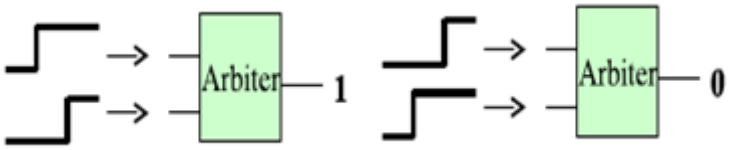

Figure 3.9(b) Arbiter response

3.4.3 Schmitt trigger PUF (STPUF)

Earlier designed PUF response is dependent on the input challenge and single CMOS property; this is the first time where the response depends on input challenge and two unique property delay and trigger voltage. Figure 3.10 presents a STPUF circuit; it consists of 8 rows and eight columns. Each row found a race between a group of Schmitt trigger controlled by a multiplexer. Table 3.2 shows the different ST architecture, their hysteresis width, and the delay. Each row of STPUF has a parallel path composed of a unique trigger circuit followed by the multiplexer. For example, PUF0 contains eight paths of STDST and NORST. The selection of trigger made concerning the threshold voltage, one Schmitt trigger threshold is maximum, and another trigger threshold on minimum such that same hysteresis width is maintained.

Each path contains a 2:1 multiplexer to apply the challenge input on the select line. When input challenge bit is low, multiplexer changes, top, and bottom path else input pass through the bottom and top signal. For the input challenge C= 8'b10101010 the multiplexer selects the path generates by STD, NOR, STD, LADDER, STD, STD, HEX, SOI in PUF0. Each PUF contains a different combination of the trigger with some hysteresis. For same input PUF1 select SOI, NOR, STD, LADDER, STD, ST, HEX, ST; in PUF2 select ST, LADDER, LADDER, SAH, NOR, SOI, SOI, SAH; in PUF3 select LADDER, STD, LADDER, ST, ST, HEX, LADDER, SAH. Two identical paths are in the race to reach the destination (arbiter), in-between path signal route through top and bottom row multiple times controlled by challenge input. Random process variation and controlled trigger point induce delay difference of ΔD to the input of arbiter. The same signal applied to different Schmitt trigger causes unusual delays and sample the variable input at a different level; additionally, trigger output is routed by common standard multiplexer induce delay due to AND-OR or AND-NOT-OR gate in the path. After eight trigger-mux stages at the arbiter end input arrives at a different instance of time. Since the delay of each trigger is a distinct signal reaches at arbiter at different times. Arbiter response is high if the top signal arrives earlier and low if the bottom signal arrives fast, i.e., if Logic '1' of a top signal is faster else logic '0'. In each row, the input signal sampled and inverted multiple times generate different voltage levels. A2D function in the cadence spectre calculator determines a digital response.

Table 3.3Placements of Schmitt trigger circuit into Row0 of PUF0.

PUF0	STD	LADDER	STD	NOR	STD	SAH	HEX	ST	Arbiter
	NOR	NOR	HEX	LADDER	LADDER	STD	LADDER	SOI	

Table 3 4 Placements of Schmitt trigger circuit into Row0 of PUF1.

PUF1	SOI	STD	STD	HEX	STD	STD	HEX	SAH	Arbiter
	SAH	NOR	HEX	LADDER	SAH	ST	SOI	ST	

Table 3.5 Placements of Schmitt trigger circuit into Row0 of PUF2.

PUF2	ST	NOR	LADDER	STD	NOR	HEX	SOI	LADDER	Arbiter
	STD	LADDER	STD	SAH	STD	SOI	ST	SAH	

Table 3.6 Placements of Schmitt trigger circuit into Row0 of PUF3.

PUF3	LADDER	SAH	LADDER	NOR	ST	STD	LADDER	SOI	Arbiter
	SAH	STD	NOR	ST	STD	HEX	SOI	SAH	

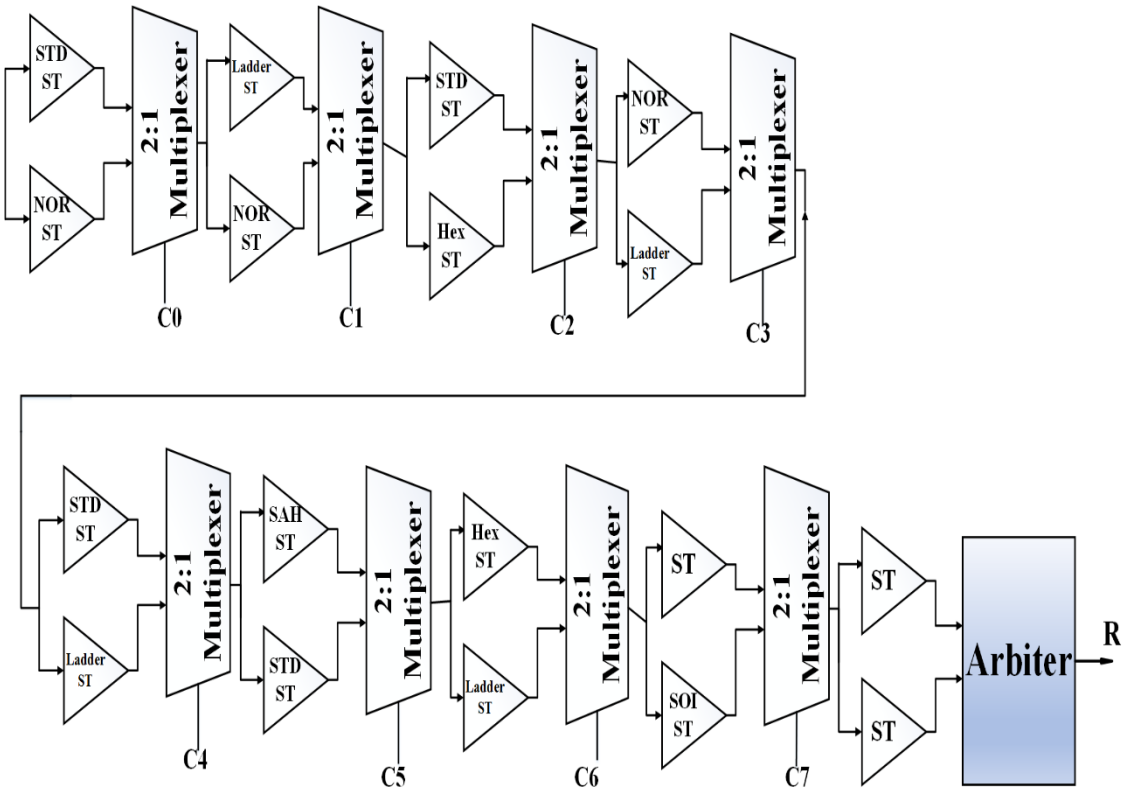

Figure 3.10 PUF0 first-row schematic

A ramp wave at input applied through each row samples 16 times, creates eight levels at arbiter response R0-R7. A storage device D flip flop (DFF) stores the arbiter output at $1/8^{th}$ of input ramp wave frequency shown in Figure 3.10. For each input challenge, half of ST trigger and half remains in idle state, generated output response in function of input challenge, trigger voltage, and delay. The responses are stored as a database of challenges. The response pair contains nonlinearity such that PUF architecture cannot be modelled with machine learning. Schmitt triggers sample ramp waves and generates random responses at different interval times. Generated response is a function of not only input challenge as well as delay and sampling nature of Schmitt triggers.

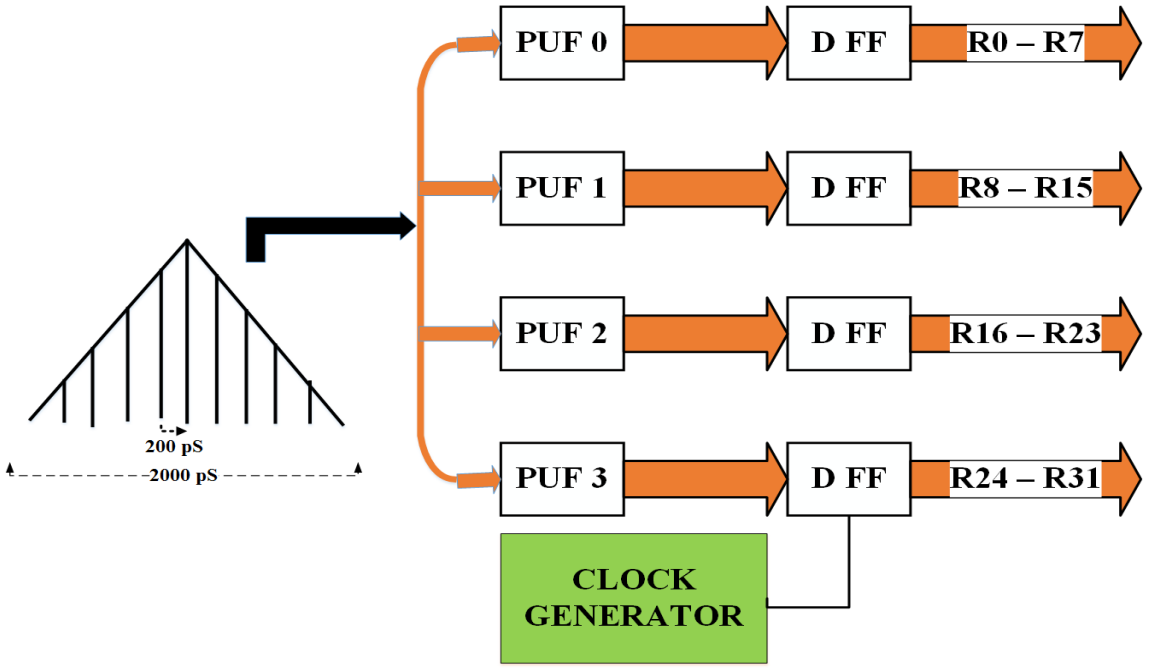

Figure 3.11 Simulation setup for PUF

In this work, a novel architecture of the challenge-response generation scheme presented with the Schmitt triggers circuit. 8-bit input challenge is mapped onto 32-bit output responses. A ramp signal applied to the input used to 4 PUF(PUF0-PUF3) in parallel, followed by four units of 8-bit Delay flip flop working in parallel output mode operates at the standard clock. The period of triangular input 2000 ps sampled ten times by clock signal of period 200 ps. The small width of ramp signal 200 ps of a variable input is sufficient to trigger and generate a constant level. Since different threshold voltages of Schmitt trigger selective ST trigger input whose HTP or LTP lies in the range, each PUF-MUX stage samples the input ten times in a period with a clock frequency of 200 ps. Each ST triggers during this small change and induces different delays into the signal, thus arrival times to arbiter must different. DFF activates after 200 ps; arbiter responses latched to 8-bit DFF. Transient response of STPUF is presented in figure 3.12, obtained with Cadence spectre for challenge input C=8'b10101010. During the first clock period response of PUF0, PUF1, PUF2, PUF3 are 5CH,4CH, B3H, and72H, respectively. Rising and falling input sampled five times each, a total of 10 responses obtained during one

input one cycle through each PUF. In this work, 40 responses received in one clock cycle, intensify the throughput by ten times.

3.5 Result & Discussion

Expectation form PUF circuits are response bit from different PUF should be unique for same challenge input; Response bits should constant with operating conditions, distributions of bits should uniform. Performance of PUF circuit evaluated with inter and intra variation of response bit. The author has proposed multiple parameters, [93] measure qualities in terms of randomness, correctness, steadiness, diffusionism uniqueness, [31] measures qualities as uniformity, uniqueness, bit aliasing, reliability. Standard parameters of evaluation are uniformity, uniqueness, and reliability explained in [108].

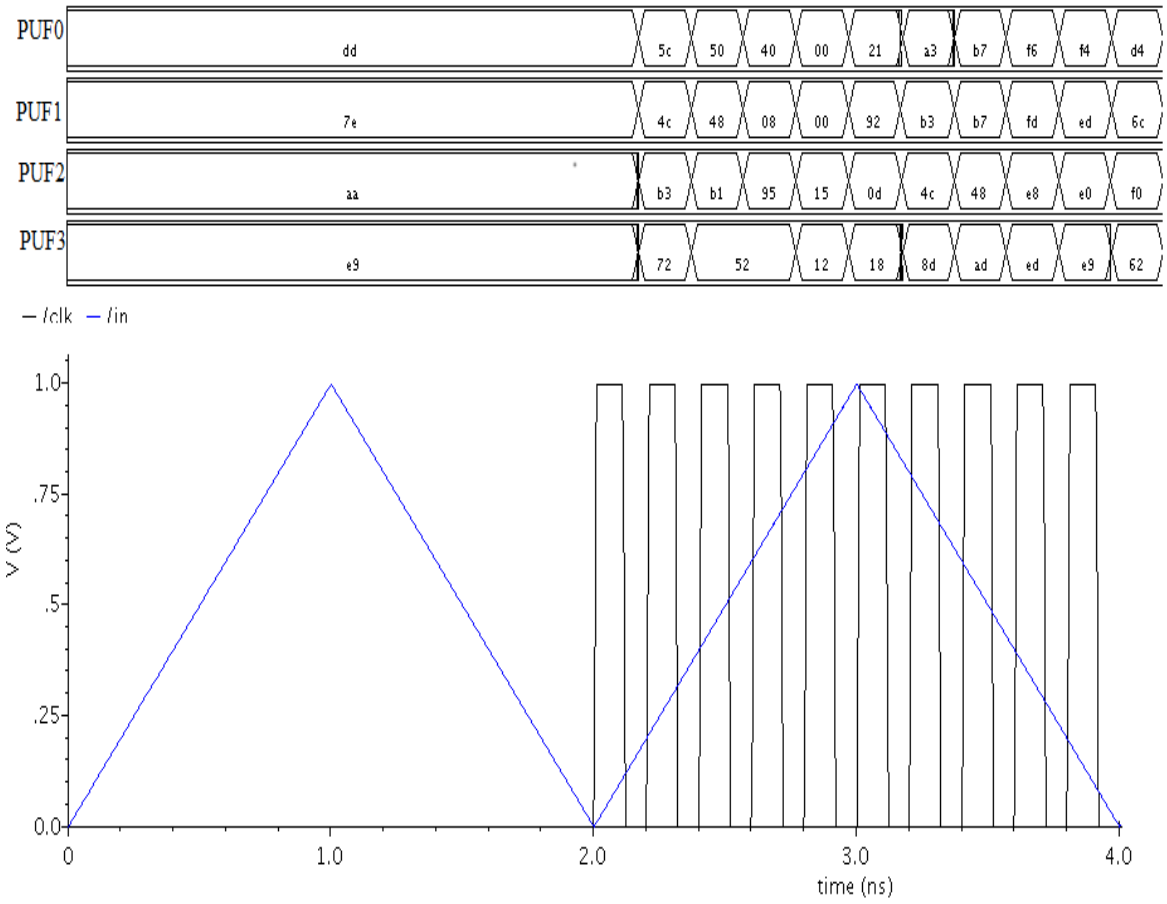

Figure 3.12 Transient simulation result of PUF

3.5.1 Inter-PUF Variation

The expectation from PUF is; for the same challenge, input to multiple PUF responses must be different. Inter-PUF variation estimates the number of bit differences between the continuous response of two PUF for the same input challenge. Hamming distance measures the number of bit changes in the response; ideally, 50% if the bit should change for application of the same input to two different PUF [28]. For n-number of bits per response from k number of devices; typical measurement of inter PUF variation calculated from equation (3.7)

$$InterHD = \frac{2}{k(k-1)} \sum_{i=k}^{k-1} \sum_{j=i+1}^{k} \frac{HD(R_i, R_j)}{n} \times 100\% \tag{3.7}$$

Where HD(R_i, R_j) is the Hamming distance(HD) between any two responses of R_i and R_j obtained from different PUF for the same challenge input.

3.5.2 Intra-PUF Variation

Intra variation measures the separation of response bit among responses originated from the same PUF when one-bit changes at input. Ideally, 50% of the bit should changes for one-bit flips at the input terminal. Intra PUF variation calculated as a hamming distance of the response generated from an intervention. Intra PUF variation is an estimate of the stability of the response bit must be stable. For n number of bits per response obtained from k number of devices intra hamming distance is measure as equation (3.8)

$$Intra\ HD = \frac{1}{K} \sum_{i=1}^{k} \frac{HD(R_{i1}, R_{i2})}{n} \times 100\% \tag{3.8}$$

Where HD (R_{i1}, R_{i2}) is the hamming distance between 1st and 2nd sample of response bit obtain from the same PUF.

3.5.3 Uniformity

It measures the distribution of response bit evenly. A PUF response should be random (chip-specific), and response bits are unbiased. Uniformity measures the ability of PUF to

produces response '1' and '0' with equal probability. Consistency is measured with hamming weight presented in equation (3.9), counts the number of high bits in the PUF's response. Its ideal value is 50%.

$$Uniformity = \frac{1}{n}\sum_{j=1}^{n} R_{i,j} \times 100\% \qquad (3.9)$$

Where $R_{i,j}$ is the j^{th} bit of n-bit response obtained from the i^{th} chip.

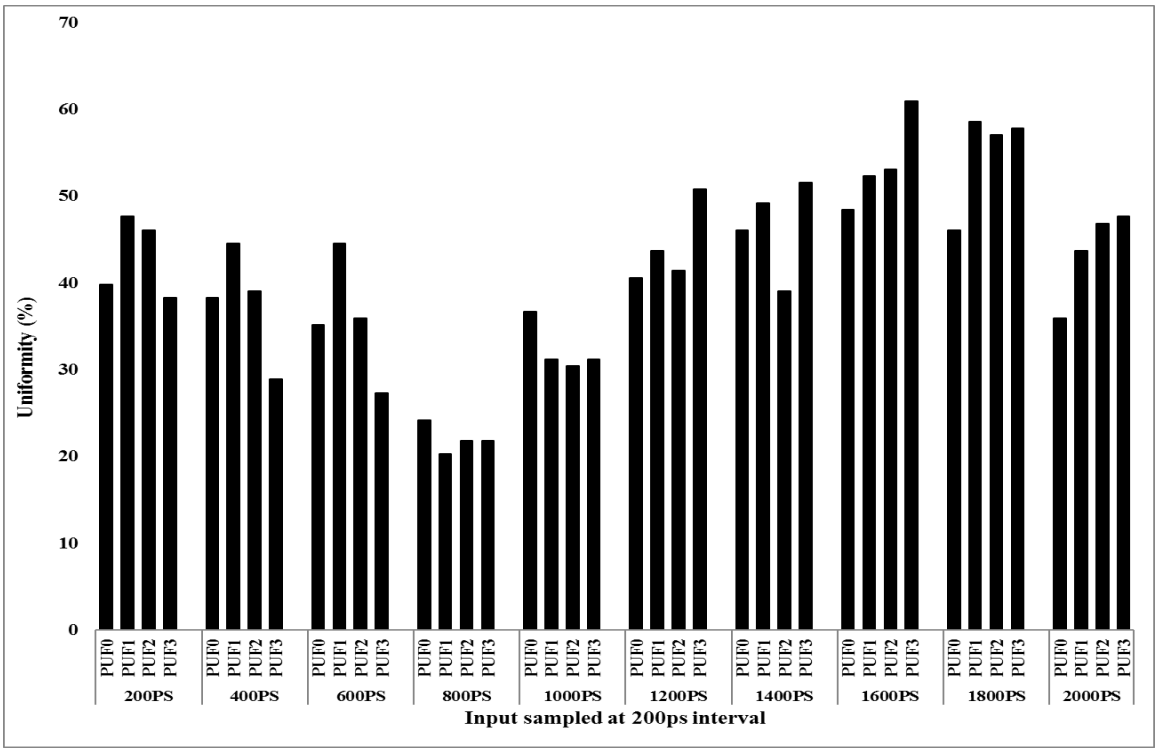

Figure 3.13 Uniformity of ST-PUF

Uniformity of STPUF measure the occurrence of bits in responses, complete ramp wave is sampled with rate 200 ps; at the starting and falling edge of input ramp signal uniformity falls. From figure3.13 it observed that for 7^{th} clock cycle uniformity is close to the ideal value i.e., 50%. The average value of uniformity for 4 PUF lies in the range of 20-60.93%.

3.5.4 Bit Aliasing

Bit aliasing is an estimate of biasness of response bits due to static hazards. Some of the bits in response are stuck to '0' if biased to zero or ground or stuck to '1' if biased to one or power supply presented in [21]. When PUF response repeated for the same challenge among different PUF, it gives a piece of exclusive side information to the attacker to predict the response. If bit aliasing occurs, then the response will get generated from different chips. Consequently, an attacker can predict the response successfully. Figure 3.14 shows that bit aliasing for the i^{th} bit of a PUF across K single chips for a challenge calculated as equation (3.10).

$$Bit\ Alia\sin g = \frac{1}{k}\sum_{j=1}^{k} R_{i,j} \times 100\% \tag{3.10}$$

Where $R_{i,j}$ is the i^{th} response obtained from the j^{th} sample.

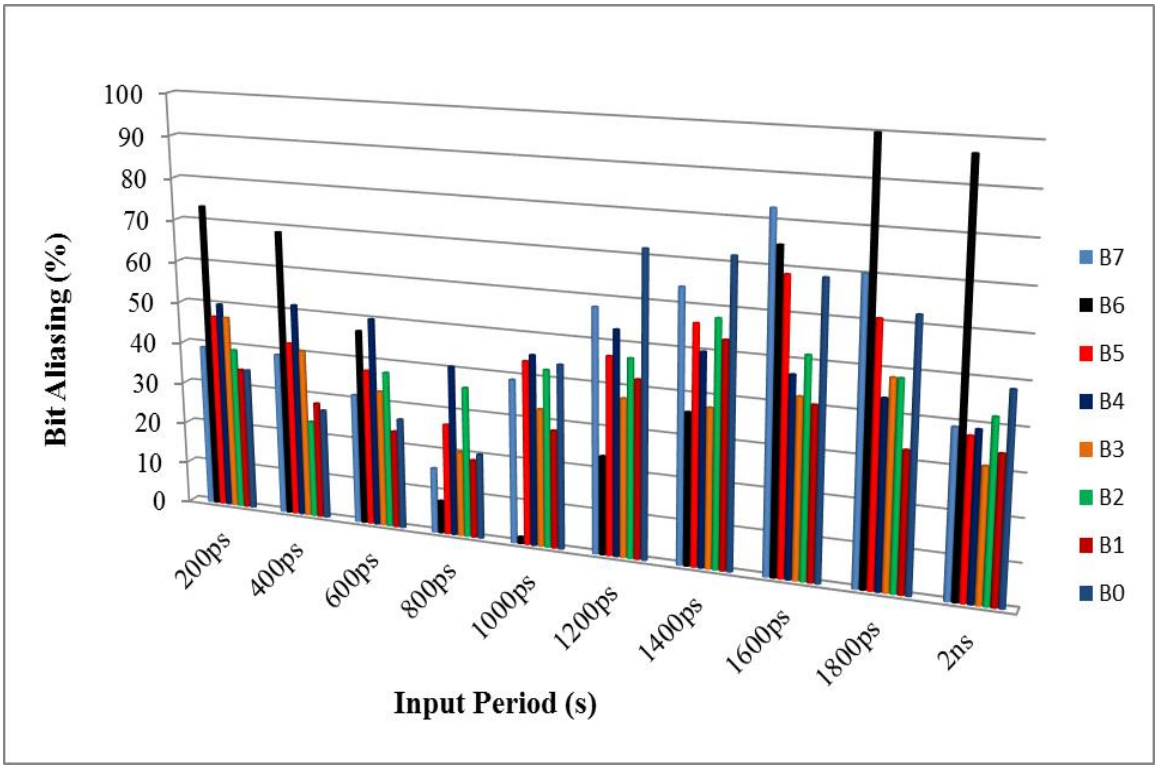

Figure 3.14 Bit aliasing of ST-PUF

Ideally, a bit aliasing metric should be 50%, and it observed that the 2^{nd} bit of PUF response in the 4^{th} sample is biased. Rising input of ramp wave is more biased towards '0'

that probable to get stuck-a-0 fault while falling input is biased towards '1,' i.e., to a stuck-at-1 fault.

3.5.5 Uniqueness

It is a measure the feature of PUF to produces new or unique responses for a challenge applies to different PUF. For the same input, different PUF must have a different response. Uniqueness calculated as Inter PUF variation (inter hamming distance); [89, 109] evaluated by comparing the hamming distance of unique response generated by different PUF shown in equation (3.11). Ideally, 50% of the response bit between two PUF, lower uniqueness presents biasness of PUF towards '0,' and higher biasness shows biasness of PUF towards '1'.

$$\text{Uniqueness} = 1 - \text{Average inter-PUF HD}$$

$$InterHD = \frac{2}{k(k-1)} \sum_{i=k}^{k-1} \sum_{j=k+1}^{k} \frac{HD(Ri, Rj)}{n} \times 100\% \tag{3.11}$$

Where Ri and Rj are the n-bit responses of two chips i and j to the same input challenge and k number of the chip.

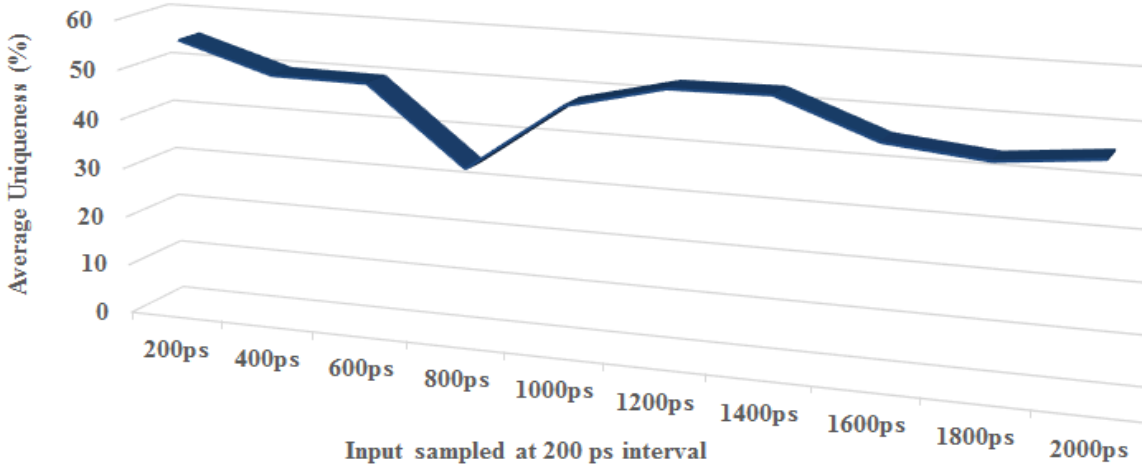

Figure 3.15 Uniqueness of ST-PUF

Figure 3.15 presents the uniqueness reduces to 34% during the fourth clock period; it is approximated to an ideal value during the falling input of ramp input and touches to 50%. The average uniqueness value ranges from 34 to 55.5%.

3.5.6 reliability

It is a measure of a PUF's response to a must stable response every time for ramp input challenge input under complex operating conditions. Environmental condition temperature and supply voltage affect the working on PUF; reliability says PUF must produce a reliable response with temperature variation and variation on the supply voltage [110]. A standard measure of reliability is intra PUF variation, measured by evaluating the hamming distance generated under different conditions. Calculated by collected k response at different voltage and temperature variation, as shown in equation (3.12). Ideally, there should be 0% intra chip variation; for 100% reliable response.

Reliability = 1 – Average intra chip HD

$$\text{Average Inter chip HD} = \frac{1}{K}\sum_{j=1}^{K}\frac{HD(Ri,Rij)}{n} \times 100\% \qquad (3.12)$$

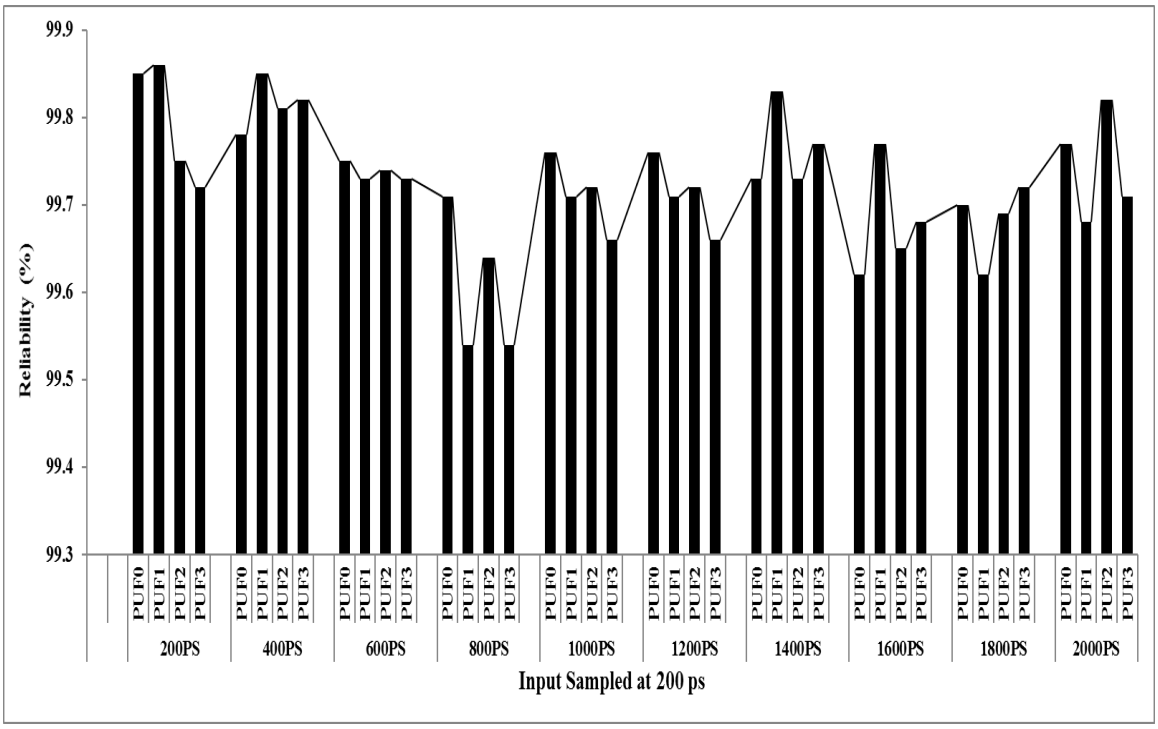

Figure 3.16 Reliability of ST-PUF with temperature variation

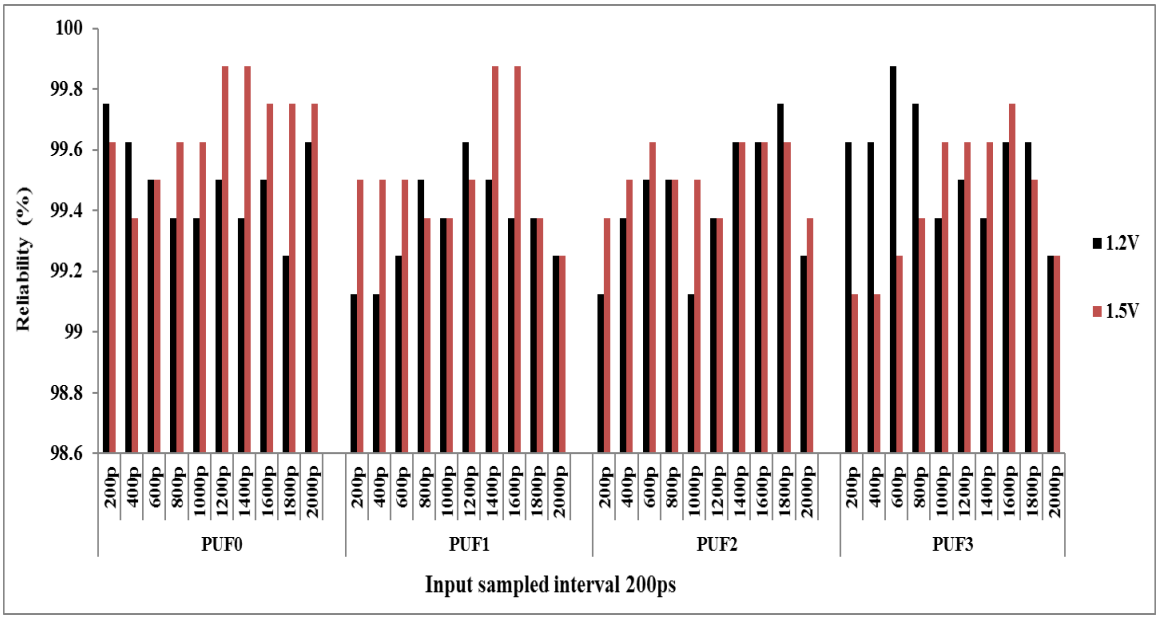

Figure 3.17 Reliability of ST-PUF with supply voltage variation

In this work, responses obtained from 4 PUF in figure 3.11 analyzed with temperature variation of 20°C to 80°C and supply voltage 1.2 V to 1.5 V. Figure 3.16 shows that reliability of PUF1 and PUF3 achieves minimum value 99.54% during 4th clock period and the maximum amount to 99.86% during 1st clock period. Reliability improves with higher voltage shown as simulation result in figure 3.17. Compare to rising a falling input (1000 ps – 2000 ps) achieve maximum value of reliability 99.875%. Table 3.7 presents the comparative result of proposed STPUF with existing MUXPUF and ROPUF. For linear input STPUF generate ten responses in one period, the parameter is presented in range, and their best result is the measure for segment number; where a segment is the time delay of 200 pS. Uniformity reaches its best value 49.8% at the 7th segment and uniqueness reaches 49.2% at 2nd segments. Best observation of reliability concerning temperature variation occurs at 1st and combination, similarly with supply voltage variation reliability reaches 99.875%. We observe that linearly falling input results in the uniform response, and linearly rising input can generate more unique responses.

Table 3.7 Comparison of PUF's Parameter

Parameter	Ideal Value [78]	STPUF		[76] Mux PUF	[17] Mux PUF	[31] RO PUF	[32] RO PUF
		Range	Best Observation For ramp input				
Uniformity (%)	50	34 – 55.5	7^{th} Segment 49.8	30.26	_ _	50.56	_ _
Uniqueness (%)	50	20.3 – 60.93	2^{nd} Segment 49.2	67.44	46.15	47.24	38.96
Reliability (%)	100	99.25 – 99.6875* 99.125 – 99.875**	$1^{st,}$ 7^{th} Segment 99.6 2^{nd},10^{th} Segment 99.8	98.01	48	99.14	98

*Temperature variation (20 – 80℃)
** Supply voltage variation (1.2V, 1.5V)
- - Not available

3.6 STPUF Evaluation

STPUF generate a unique response for each challenge based on its physical feature as delay and hysteresis. It can evaluate for authentication and key storage purpose.

3.6.1 Authentication

Four different PUF architecture is designed, and their challenge-response pair is stored in the database during design. Figure 3.18 shows a set of 3 pairs of challenge and response. For the given challenge-response of each PUF are different. PUF protects the device authentication mechanism. During the evaluation, if a person wants to authenticate the device apply challenge, for example, C= 8'b1000111 their current obtained response 32'hACB10C95. The obtained response is compared with the response stored in the CRP database. Since response matched with CRP of PUF2, the device contains PUF2 authenticate to access while others not authenticated. Since response are computed as a function of a physical feature of Schmitt trigger, a counterfeit if stores all possible CRPs they cannot evaluate the response since physical properties are unknown to them.

3.6.2 Key Storage

In section 3.2, it is highlighted that a PUF avoids directly storage the key, a trace of the key, i.e. helper data, is needed to store. The error correction process completed in two

steps; enrollment and generation phase. ECC code correct error induces due to environmental conditions. In the enrollment phase response of PUF(R) and ECC encoding is XORed to generate helper data usually a parity bit. Hamming code approach, adds a parity bit to the position of weight 2^0,2^1, $2^{2,}$ and so on. In this example input to ECC encoder S=8'b00001101, four parity bits added as P1, P2, P4, and P8 enhance the hamming code as 12-bits by stuffing four parity bits, C= 12'b000001100110. 12-bit PUF response selected as 12'b010100101001 XORed with ECC syndrome result in helper data H=12'b010101001111. Helper data are stored in non-volatile memory, and available in public, and leaks information. The adversary who has access to these data not able to derive the key since PUF's response is hidden. The helper data are necessary for error correction.

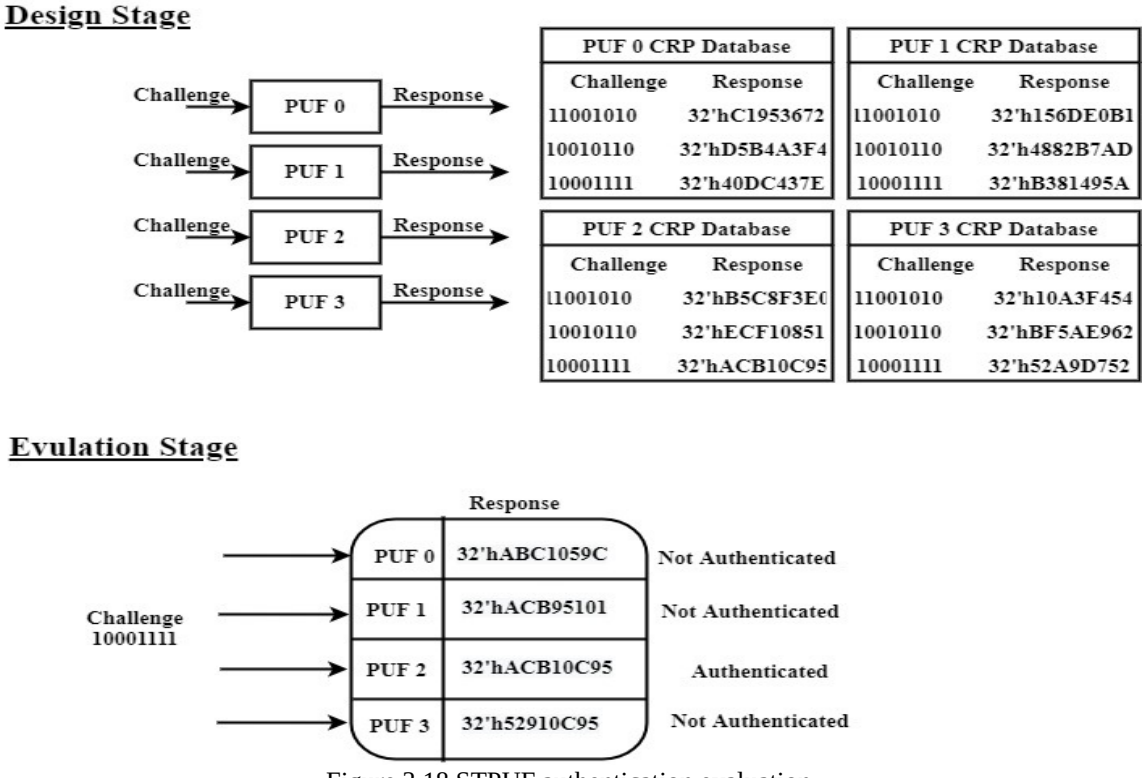

Figure 3.18 STPUF authentication evaluation

Chapter 4
Substitution BOX (SBOX)

4.1 Introduction

Advanced Encryption Standard (AES) is adopted as a secured encryption standard for privacy and security of data 2001 by NIST. AES is used to keep communication secure over digital devices like data communication, e-banking, military, mobile. AES is a symmetric key encryption standard that requires a similar key for encryption and decryption. Three varieties of AES are AES128, AES192, and AES256 depending on the key size. In AES128, 128-bit input data scrambled with a 128-bit secret key that produces ciphertext. Encryption involves iteration, repeats four internal function N-times depending on key length [111]. The key size of AES-128, AES-192, and AES-256 are 10, 12, and 16, respectively. AES starts with AddroundKey- input text mixes with the key process through SubByte- each byte is replaced by a nonlinear byte, ShiftRow- each bit of substitute byte gets shifted circularly by 1,2 or 3 positions and MixColumn- completes with matrix multiplication. MixColumn, along with shift row, is a significant source of diffusion in cipher. Substitution byte is the most crucial block in the encryption process; it includes nonlinearity. The substitution process planned to execute based on Shannon's confusion-diffusion principles [73, 112].

Confusion- induce substantial confusion required to keep the text secret in the ciphertext. It is the essential requirement of the cryptographic algorithm.

Diffusion- is a measure of necessary randomness required in the substituted byte. It required that if one bit at SBOX input changes, half of the byte in substitute byte should get a replacement. It includes unpredictability, and the complex computation requires to guess the key.

The requirement of SBOX is its output should be unpredictable. The input stage of the Subbyte process is the AddRoundKey stage, XORed the 128-bit user input with the secret key. Figure 4.1 presents a 128-bit output of AddRoundkey grouped into 16 bytes. SubByte contains 256 nonlinear bytes arranged as a 16×16 matrix known as a

substitution box or SBOX shown in table 4.1. Each SBOX operates on individual byte, parallelly 16 nonlinear bytes at the output for next stage function. To read the SBOX value, output byte of add round key acts as address. the 1st selects row address, and 2nd nibble selects the column address, which picks one out of 16 rows or columns, respectively. Typical cell specified by row and column gives the byte to be substituted explained in [70, 74].

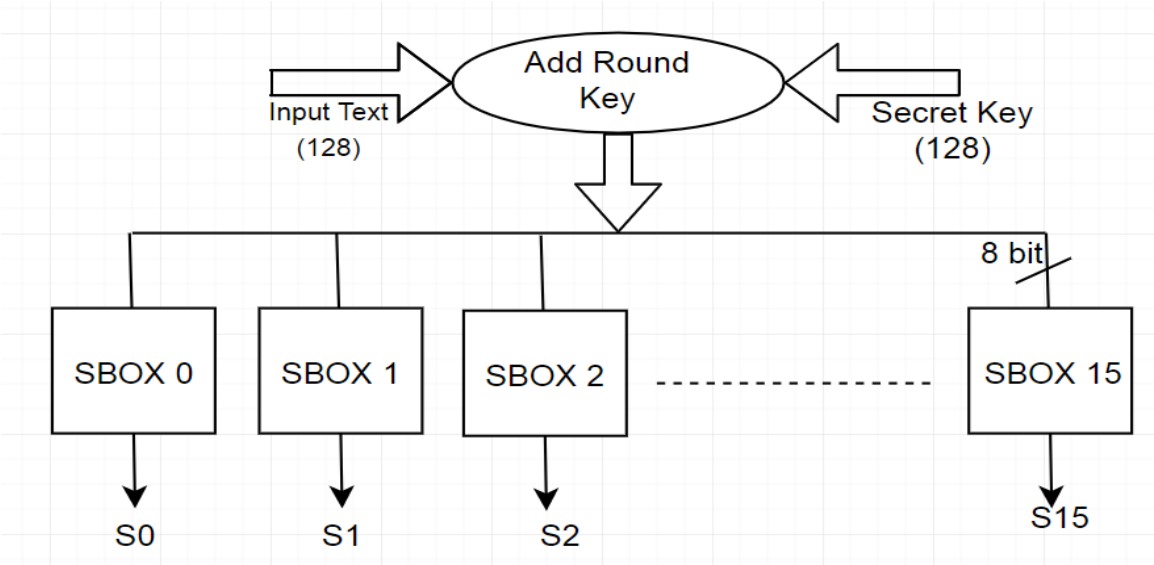

Figure 4.1AddRoundKey and SubByte Stage

Table 4.1SBOX Values

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

4.2 Architecture of SBOX

An efficient design of SBOX is rapidly growing to obtain high throughput and low area usage in smaller devices on the chip. Very-large-scale integration (VLSI) implementation of SBOX described in [113] characterized by (a) complexity of the design, (b) power requirement to find new byte, and (c) time required to produces byte. The simple and straight forward approach of SBOX shown in [114] is (i) lookup table (LUT) based memory architecture and (ii) nonmemory based computational architecture. The former approach of implementation of SBOX takes the shape of the hardware lookup table, suffers from large area requirements and static delay. The latter approach, based on composite filed arithmetic, requires large-signal integrity. The drawback of the second approach is large power consumption, but the delay is lesser than the former—the desired trade-off which architecture best suited for an application.

4.2.1 Lookup table (LUT SBOX)

The memory architecture of SBOX requires a large memory that can store 256 bytes of data, usually implemented with EPROM or lookup table (LUT) shown in figure 4.2. The pre-computed value of SBOX byte stored in a memory location; a fetching circuit requires to fetch the byte whose address provided by the input byte of SBOX. An efficient LUT-SBOX presented in [30] shows a fetching circuit implemented with the multiplexing circuit. This architecture suffers from the static delay of lookup table access time. Optimize of underlying structure AND-OR-NOT gate optimizes the performance of SBOX [47]. Additional memory is required to store pre-computed value and fetch the byte from rows and columns to bring back the desired amount. To reduce the memory requirement of LUT-SBOX, a pipelined architecture shown in [88] performance improves by several iterations needed in a byte to compute byte substitution. Two hundred fifty-six bytes divided into 16 equal-size groups, first 2 MSB select a group, next 2 MSB select the row and column in the selected group with 2:4 decoder. The last bit enables 4:1 multiplexer to fetch a required byte from select row and column.

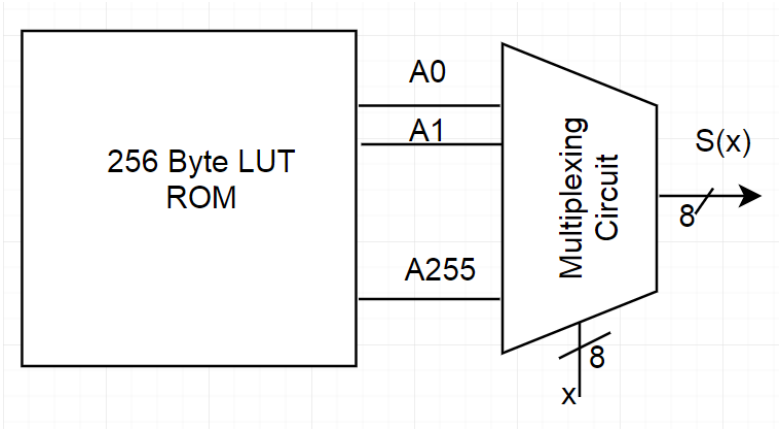

Figure 4.2 Lookup table SBOX

4.2.2 Computational Architecture of SBOX

The nonmemory approach of SBOX computes a unique value for each input in composite filed arithmetic explained in[115]. Multiplicative inverse (MI) and affine transform (AT) are two internal of SBOX. Whose implementation requires only a combinatorial logic block. Composite filed arithmetic presented in [116] decomposes the compute the multiplicative inversion in GF (2^8). Complexities further simplified into [117] by further decomposing the multiplicative inversion in GF $(2^4)^2$ and GF $(2^4)^2$ into GF (2^2). Authors in [118] shown improvement in the calculation steps by switching the representation. Byte substitution begins isomorphic mapping of the field of input into composite filed. Multiplicative inverse computes a nonlinear byte in the Galois field (GF). Inverse isomorphic remap the result in composite filed followed by an affine transform presented in [119].

Individual bit of byte in GF (2^8) element denoted the coefficient of Power in GF (2^8) polynomial. Arbitrary polynomial presented as (bx + c) in GF (2^8) with irreducible polynomial (x^2+Ax+B). AT reduce the complexity in computation power of the polynomial need to converted in lower power. The multiplicative inverse for (bx+c) can found using equation (4.1) where b is the most significant bit (MSB), and c is the least significant bit (LSB).

$$(bx+c)^{-1} = b(b^2B+bcA+c^2)^{-1} + (c+bA)(b^2B+bcA+c^2)^{-1} \quad (4.1)$$

The mapping structure of fields and the irreducible polynomials (x^2+x+k) modified as equation (4.2). Figure 4.3 presents the internal blocks in the architecture of equation (4.2) where B = λ (constant $(1100)_2$) and A in equation (4.1).

$$(bx+c)^{-1} = b(b^2\lambda + c(b+c)^{-1}x + (c+b)(b^2\lambda + bcA + c^2)^{-1} \quad (4.2)$$

It can analyze from equation (4.2) hardware architecture of multiplicative inverse requires adder multiplier, squarer, and inverse Galois field (GF). Each of these blocks simplifies into standard algebraic form while implementing the circuit for computation.

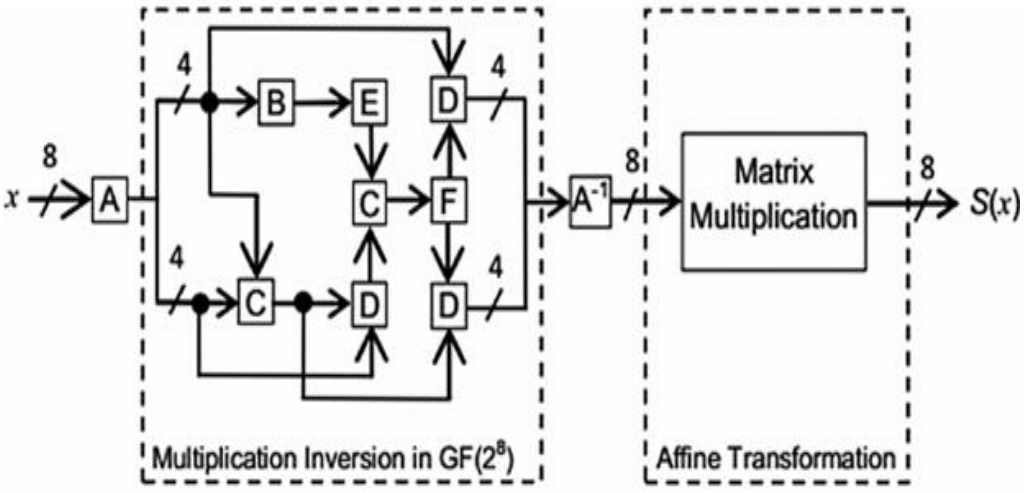

Figure 4.3 Multiplicative inverse and Affine transform in SBOX [88,113, 120]

A is isomorphic mapping convert value of GF (2^8) to GF (2^4).

B is squarer in GF (2^4)

C is a sum in GF (2^4)

D is multiplication in GF (2^4)

E is multiplication with constant in GF (2^4)

F is an inversion in GF (2^4)

A^{-1} is an inverse isomorphic map.

The composite field represented as $GF(2^n)^m$ is isomorphic to the Galois field $GF(2^k)$ where k=nm. Computation of multiplicative inverse in GF (2^8) requires substantial resources, to reduce the complexity GF (2^8) can be built with lower field GF (2^1), GF (2^2) and GF $(2^2)^2$ as presented in equation (4.3).

$$GF(2^2) = GF(2^1) \quad : x^2 + x + 1$$
$$GF((2^2)^2) = GF(2^2) \quad : x^2 + x + \phi \qquad (4.3)$$
$$GF(((2^2)^2)^2) = GF((2^2)^2) \quad : x^2 + x + \lambda$$

4.3 Internal Architecture of SBOX

Details architecture diagram of SBOX presented in figure 4.3; each internal block designed with a combinatorial logic block. Implementation requires a group of XOR, AND, and NOT primitive. Here cadence virtuoso schematic composer has been used to develop the transistor level diagram and cadence spectre for validating the simulation result at CMOS 90nm technology node. Two different designs of SBOX presented with static CMOS logic and dynamic CMOS logic. Each design logic has its pros and cons. Static logic requires a large area; dynamic CMOS logic reduces the number of a transistor requires at the cost of power consumption.

4.3.1 Isomorphic mapping

A finite field element mapped to composite filed employing isomorphic mapping (δ) function. After performing the computation, the result needed to plan back to the finite field with the help of inverse isomorphic function (δ^{-1}). δ and δ^{-1} represented as 16×16 matrices decided by a polynomial in GF (2^8). If q is the element of isomorphic mapping in GF (2^8) presented as $\delta\times q$ and its inverse is $\delta^{-1}\times q$ in equation (4.4) and (4.5), respectively.

$$\delta\times q = \begin{pmatrix} 1&0&1&0&0&0&0&0\\ 1&1&0&1&1&1&1&0\\ 1&0&1&0&1&1&0&0\\ 1&0&1&0&1&1&1&0\\ 1&1&0&0&0&1&1&0\\ 1&0&0&1&1&1&1&0\\ 0&1&0&1&0&0&1&0\\ 0&1&0&0&0&0&1&1 \end{pmatrix} \times \begin{pmatrix} q_7\\ q_6\\ q_5\\ q_4\\ q_3\\ q_2\\ q_1\\ q_0 \end{pmatrix} = \begin{pmatrix} q_7 \oplus q_5\\ q_7 \oplus q_6 \oplus q_4 \oplus q_3 \oplus q_2 \oplus q_1\\ q_7 \oplus q_5 \oplus q_3 \oplus q_2\\ q_7 \oplus q_5 \oplus q_3 \oplus q_2 \oplus q1\\ q_7 \oplus q_6 \oplus q_2 \oplus q_1\\ q_7 \oplus q_4 \oplus q_3 \oplus q_2 \oplus q_1\\ q_6 \oplus q_4 \oplus q_1\\ q_6 \oplus q_1 \oplus q_0 \end{pmatrix} \qquad (4.4)$$

$$\delta^{-1}\times q=\begin{pmatrix}1&1&1&0&0&0&1&0\\0&1&0&0&0&1&0&0\\0&1&1&0&0&0&1&0\\0&1&1&1&0&1&1&0\\0&0&1&1&1&1&1&0\\1&0&0&1&1&1&1&0\\0&0&1&1&1&0&0&0\\0&1&1&1&0&1&0&1\end{pmatrix}\times\begin{pmatrix}q_7\\q_6\\q_5\\q_4\\q_3\\q_2\\q_1\\q_0\end{pmatrix}=\begin{pmatrix}q_7\oplus q_6\oplus q_5\oplus q_1\\q_6\oplus q_2\\q_6\oplus q_5\oplus q_1\\q_6\oplus q_5\oplus q_4\oplus q_2\oplus q_1\\q_5\oplus q_4\oplus q_3\oplus q_2\oplus q_1\\q_7\oplus q_4\oplus q_3\oplus q_2\oplus q_1\\q_5\oplus q_4\\q_6\oplus q_5\oplus q_4\oplus q_2\oplus q_0\end{pmatrix} \tag{4.5}$$

Schematic of isomorphic mapping presented in figure 4.4, matrix multiplication is required only XOR to implement. 8-bit finite filed input [7:0]o mapped to an 8-bit composite filed number [7:0]q given in simulation result in figure 4.5.

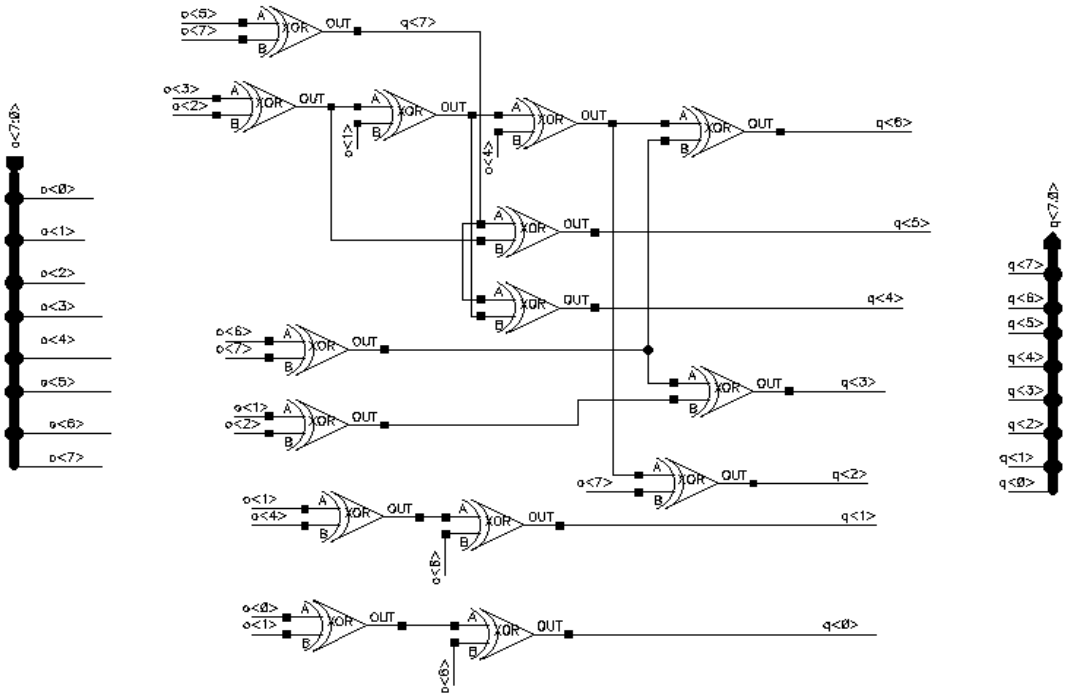

Figure 4.4 Schematic of Isomorphic Mapping

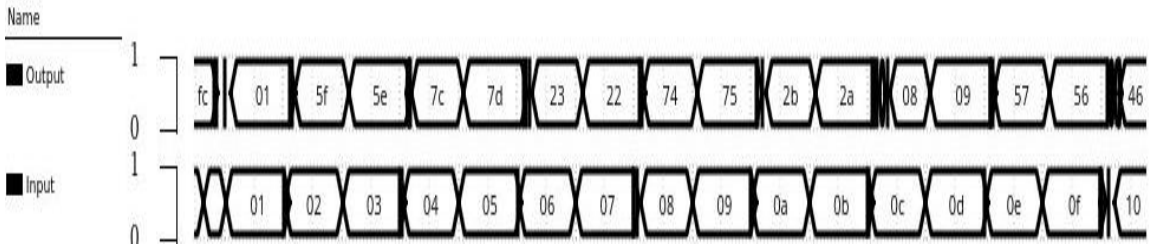

Figure 4.5 Simulation result of Isomorphic Mapping

In composite filed arithmetic, an arbitrary polynomial presented as (bx+c) a binary number in Galois filed q showed $(q_H x + q_L)$. Where q is a 4-bit number $(q_3 q_2 q_1 q_0)_2$ further decomposing in lower Galois filed $(q_H = q_3 x + q_2)$ $(q_L = q_1 x + q_0)$ and. The logical computation of the internal component of the multiplicative inverse below.

4.3.2 Addition in GF (2^4)

Addition in Galois filed can implement with bitwise XOR operation

4.3.3 Squaring in GF (2^4)

Squaring in GF (2^4) implemented by substituting $k=q^2$ where k and q are the 4-bit element in GF(2^4); k presents $k_3k_2k_1k_0$, q presents $q_{3q}q_2q_1q_0$

$$k = k_H x + k_L = (q_H x + q_L)^2$$

$$k = {q_H}^2 x^2 + q_H q_L x + q_H q_L x + {q_L}^2$$

$$k = {q_H}^2 x^2 + {q_L}^2$$

Substitute $x^2=x+\phi$, GF (2^4) further decomposes into GF (2^2) modifies as

$$k = {q_H}^2 (x + \phi) + {q_L}^2$$

Decomposition of the higher term of K (K_H)

$$k_H = {q_H}^2 x \quad and \;\; K_L = {q_H}^2 \phi + {q_L}^2$$

Substitute $x^2=x+1$, GF (2^2) further simplifies into GF (2^1) modifies as

$$k_H = {q_H}^2 = (q_3 x + q_2)^2 = {q_3}^2 x^2 + 2.q_3 q_2 x + {q_2}^2 = q_3 x^2 + q_2$$

Substitute $x^2=x+1$

$$k_H = q_3(x + 1) + q_2 = q_3 x + (q_3 + q_2)$$

Thus $k_3=q_3$ and $k_2=(q_3+q_2)$ where + symbol stands for XOR operation in GF(2^1) field. Similarly, the lower term of k (K_L) decomposes as by putting $\phi=(10)_2$

$$k_L = {q_H}^2 + {q_L}^2 = (q_3 q_2)^2 (10)_2 + (q_1 q_0)^2$$

$$k_L = (q_3 + q_2)^2 (1_2 x + 0) + (q_1 x + q_0)^2$$

$$K_L = ({q_3}^2 x^2 + 2.q_3 q_2 x + {q_2}^2) x + ({q_1}^2 x^2 + 2.q_0 q_1 x + {q_0}^2)$$

$$k_L = q_3 x^3 + q_2 x + q_1 x^2 + q_0$$

Substitute $x^2 = x + 1$

$$x^3 = x^2.x = (x+1).x = x^2+x = x+1+x = 1$$

$$K_L = q_3.1 + q_2x + q_1(x+1) + q_0$$

$$K_1x + K_0 = (q_2 + q_1)x + (q_3 + q_1 + q_0)$$

$$k_1 = (q_2 + q_1) and\ k_0 = (q_3 + a_1 + q_0)$$

where (+) symbol stands for XOR operation in the GF field.

Reduced expression presented into the standard algebraic form given in equation (4.6). Final expression to map squarer with a hardware structure shown in figure 4.6. Simulation result in figure 4.7 is squarer output in composite filed arithmetic.

$$k_3 = q_3$$

$$k_2 = q_3 \oplus q_2 \qquad (4.6)$$

$$k_1 = q_2 \oplus q_1$$

$$k_0 = q_3 \oplus q_1 \oplus q_0$$

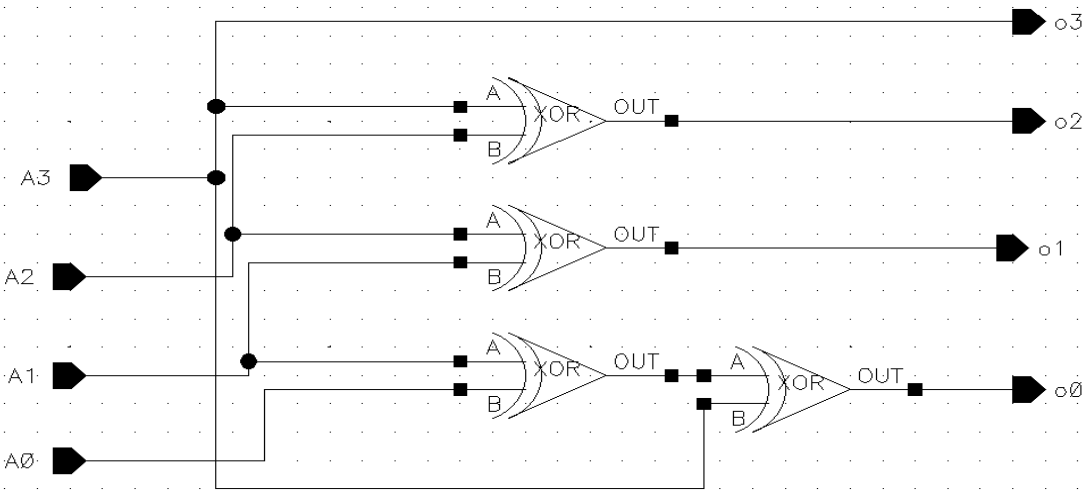

Figure 4.6 Schematic of Squarer

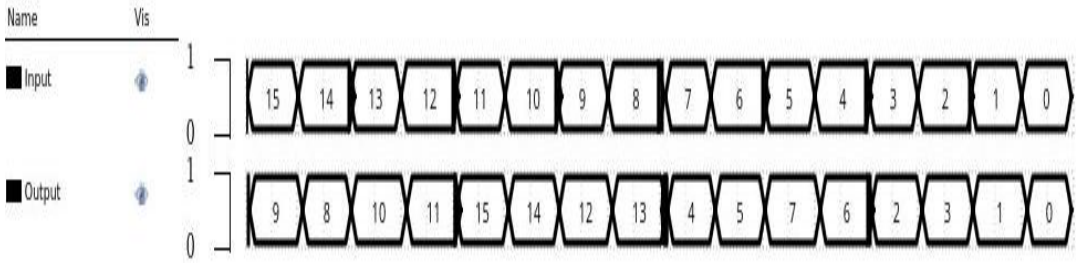

Figure 4.7 Simulation Result of Squarer

4.3.4 Multiplication with constant (λ)

Let k=qλ where k presents $k_3k_2k_1k_0$, q presents $q_3q_2q_1q_0$ and λ= $(1100)_2$

$k = k_H x + K_L = (q_H x + q_L)(\lambda_H + \lambda_L)$ $\lambda_H=(11)_2$ and $\lambda_L=(00)_2$

$K = q_H\lambda_H x^2 + q_L\lambda_L x$

Substitute $x^2=x+\phi$, the irreducible polynomial in GF(2^2) modifies as

$$K = q_H\lambda_H(x + \phi) + q_L\lambda_H x$$

$$k = (q_H\lambda_H + q_L\lambda_L)x + q_H\lambda_H\phi$$

Module reduction in GF(2^1) performed by decomposing the higher term (k_H) and lower term (k_L) separately.

Decomposition of k_H

$$k_H = q_H\lambda_H + q_L\lambda_L$$

$$k_H = (q_3q_2)(11)_2 + (q_1q_0)(11)_2$$

$$k_H = (q_3x + q_2)(x + 1) + (q_1x + q_0)(x + 1)$$

$$k_H = q_3x^2 + (q_3 + q_2)x + q_2 + q_1x^2 + (q_1 + q_0)x + q_0$$

Put $x^2=x+1$

$$k_H = q_3(x + 1) + (q_3 + q_2)x + q_2 + q_1(x + 1) + (q_1 + q_0)x + q_0$$

$$k_H = (q_3 + q_3 + q_2 + q_1 + q_1 + q_0)x + (q_3 + q_2 + q_1 + q_0)$$

$$k_3x + k_2 = (q_2 + q_0) + (q_3 + q_2 + q_1 + q_0)$$

Thus $K_3=(q_2+q_0)$ and $k_2=(q_3+q_2+q_1+q_0)$ where + presents bitwise XOR.

Decomposition of k_L

$$k_L = q_H\lambda_H\phi$$

$$k_L = (q_3q_2)(11)_2(10)_2$$

$$k_L = (q_3x + q_2)(x + 1)x$$

$$k_L = q_3x^3 + q_2x^2 + q_3x^2 + q_2x$$

Put $x^2 = x + 1 \; and \; x^3 = 1$

$$k_L = q_3(1) + q_2(x+1) + q_3(x+1) + q_2 x$$

$$k_L = (q_3 + q_2 + q_2)x + (q_3 + q_3 + q_2)$$

$$k_1 x + k_0 = q_3 x + q_2$$

Thus $K_1=q_3$ and $k_0=q_2$

Final algebraic expression of multiplication with λ given in equation(4.7). These expressions mapped the hardware structures presented in figure 4.8. Simulation result in figure 4.9 is multiplication with a static value λ in the composite filed.

$$k_3 = q_2 \oplus q_0$$

$$k_2 = q_3 \oplus q_2 \oplus q_1 \oplus q_0 \qquad (4.7)$$

$$k_1 = q_3$$

$$k_0 = q_2$$

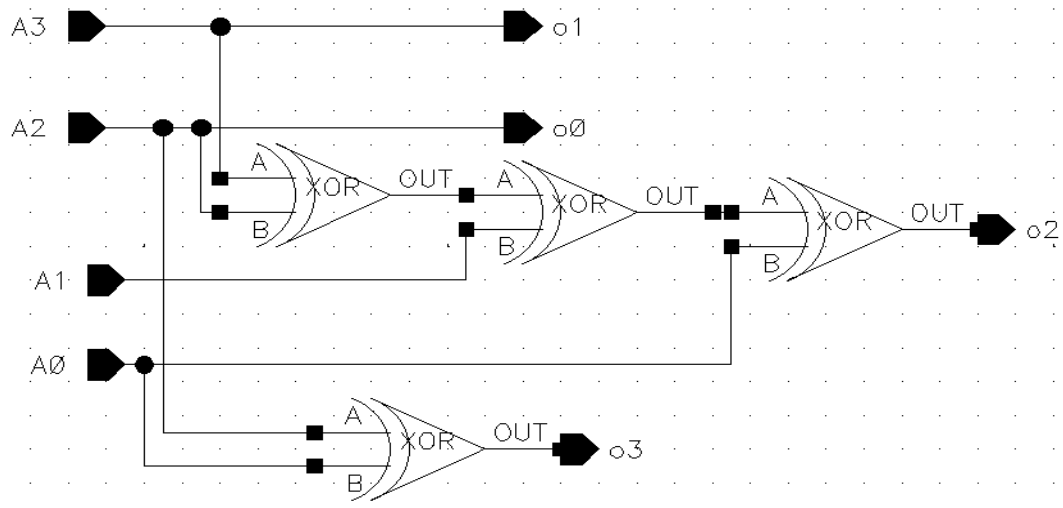

Figure 4.8 Schematic of Multiplication with Constant (λ)

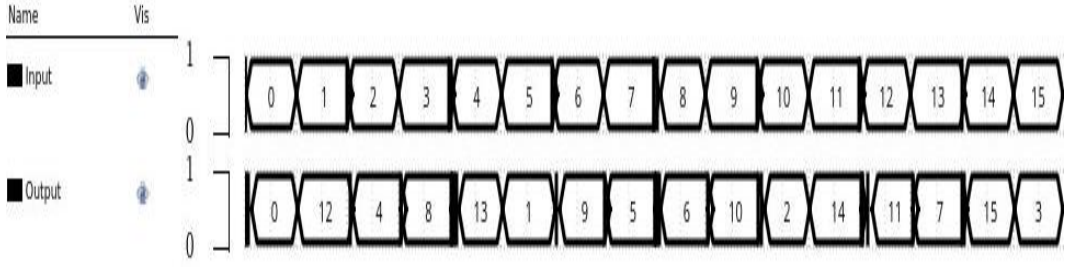

Figure 4 9 Simulation Result of Multiplication with Constant (λ)

4.3.5 GF (2^2) in multiplication

Let k=qw are the 2-bit element of GF (2^2); k presents k_1k_0, w present w_1w_0, and q presents q_1q_0

$$k = k_1x + k_0 = (q_1q_0)(w_1w_0) = (q_1x + q_0)(w_1x + w_0)$$

$$k = q_1w_1x^2 + q_0w_1x + q_1w_0x + q_0w_0$$

$$\text{Put} x^2 = x + 1$$

$$k = q_1w_1(x+1) + q_0w_1x + q_1w_0x + q_0w_0$$

$$k_1x + k_0 = (q_1w_1 + q_0w_1 + q_1w_0)x + (q_1w_1 + q_0w_0)$$

Thus, algebraic expression of GF (2^2) multiplication is

$$K_1 = q_1w_1 \oplus q_0w_1 \oplus q_1w_0$$
$$k_0 = q_1w_1 \oplus q_0w_0 \qquad (4.8)$$

Algebraic expression of multiplication of two 2-bit numbers [1:0]A and [1:0]B presented in equation(4.8). From the expressions, we can map the hardware structure of multiplication in GF(2^2) filed shown in figure 4.10 The figure 4.11 is the simulation result of multiplication in GF(2^2).

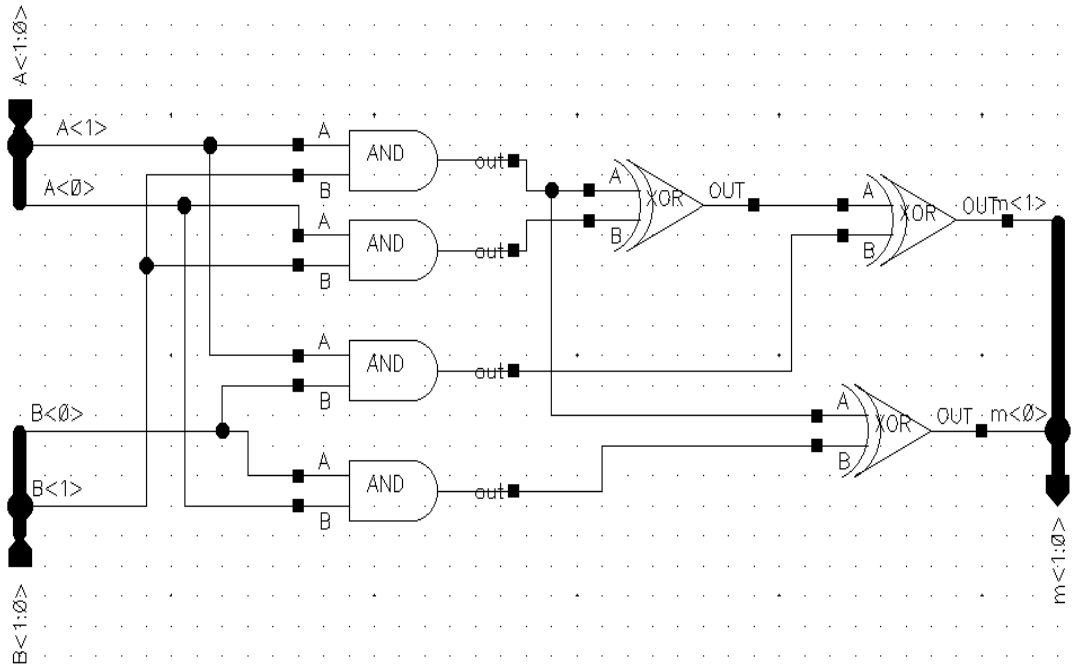

Figure 4.10 Schematic of Multiplication in GF (2^2)

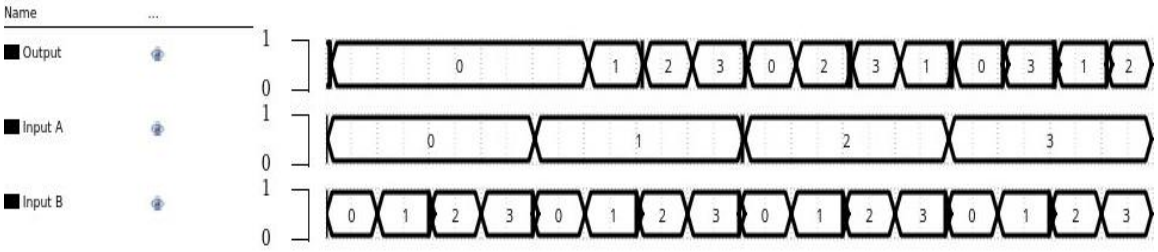

Figure 4.11 Simulation Result of Multiplication in GF (2^2)

4.3.6 Multiplication with constant (φ)

Let k=qϕ are the 2-bit element of GF (2^2); k presents k_1k_0, q presents q_1q_0

$$k = k_1x + k_0 = (q_1q_0)(10)_2 = (q_1x + q_0)x$$

$$k = q_1x^2 + q_0x$$

$$Put x^2 = x + 1$$

$$k = q_1(x+1) + q_0x \qquad (4.9)$$

$$k = k_1x + k_0 = (q_1 + q_0)x + q_1$$

$$Thus k_1 = q_1 \oplus q_0 and\ k_0 = q_1$$

Equation(4.9) is the boolean expression of multiply with a 2-bit constant value in the composite filed. It requires only one XOR gate, shown in figure 4.12.

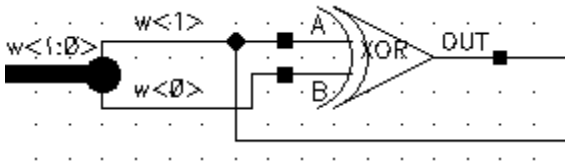

Figure 4.12 Schematic of Multiplication with ϕ

4.3.7 GF(2^4) Multiplication

Multiplication of two 4-bit numbers in GF (2^4) is a complicated step in the computation in multiplicative inverse. It required multiplication in GF (2^2) and multiplied with ϕ as an internal block to multiples 4-bits number in composite filed. Two 4-bit input grouped into 2-bit size applies to reduce sized Galois field ($GF2^2$) two field multiplication.

Let k=qw are the element in GF (2^4).

$$k = k_H x + k_L = (q_H x + q_L)(w_H x + w_L)$$

$$k = (q_H w_H)x^2 + (q_H q_L + q_L w_H)x + q_L w_L$$

$$Put x^2 = x + \phi \qquad (4.10)$$

$$k = (q_H w_H)(x + \phi) + (q_H w_L + q_L w_H)x + q_L w_L$$

$$k = k_H x + k_L = (q_H w_H + q_H w_L + q_L w_H)x + q_H w_H \phi + q_L w_L$$

The hardware structure of the equation (4.10) implements with addition and multiplication in GF (2^2) shown in figure 4.13 and its simulation result is. present in figure 4.14

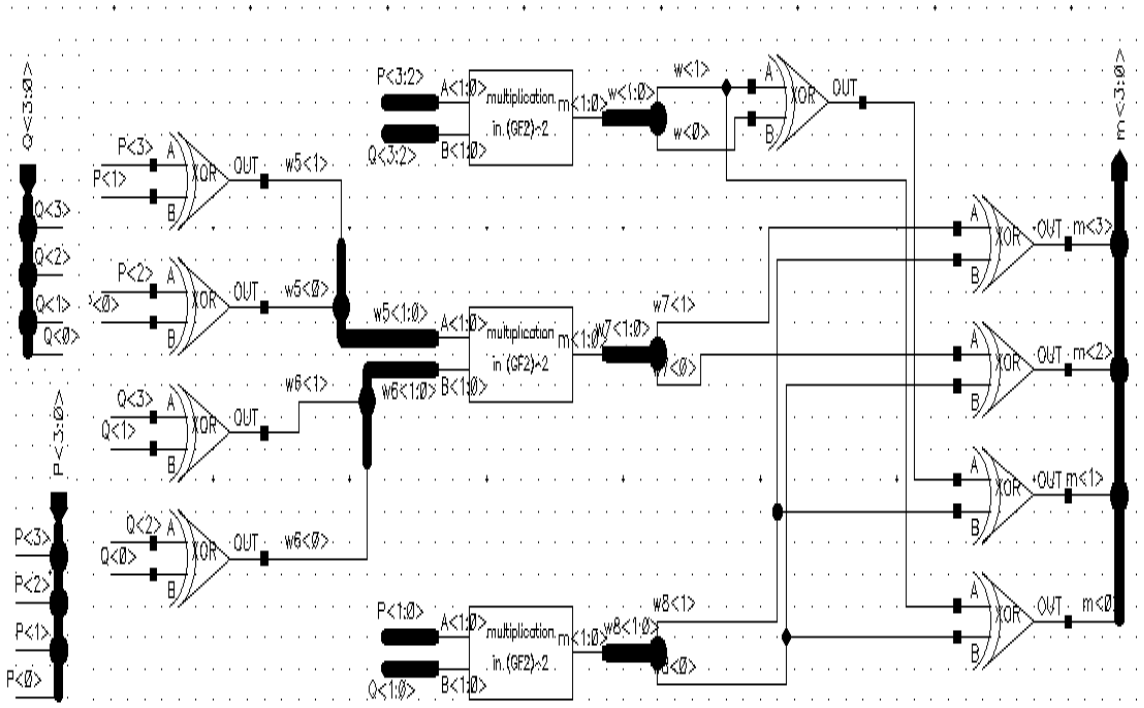

Figure 4.13 Schematic of Multiplication in GF(2^4)

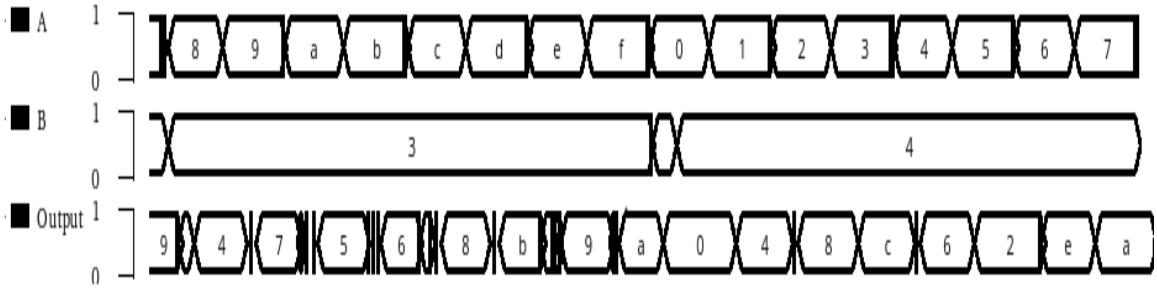

Figure 4.14 Simulation Result of Multiplication in GF(2^4)

4.3.8 Multiplicative Inversion in GF (2^4)

Inversion of [3:0]q in GF (2^4) implement with the Boolean expression in equation (4.11) of pre-computed value in table 4.2. Gate level schematic presented in figure 4.15 requires XOR and AND cell, and their simulation result in figure 4.16 shows the inverse in composite filed.

$$q_3^{-1} = q_3 \oplus q_3q_2q_1 \oplus q_3q_0 \oplus q_2$$

$$q_2^{-1} = q_3q_2q_1 \oplus q_3q_2q_0 \oplus q_3q_0 \oplus q_2 \oplus q_2q_1$$

$$q_1^{-1} = q_3 \oplus q_3q_2q_1 \oplus q_3q_1q_0 \oplus q_2 \oplus q_2q_0 \oplus q_1$$

$$q_0^{-1} = q_3q_2q_1 \oplus q_3q_2q_0 \oplus q_3q_1 \oplus q_3q_1q_0 \oplus q_3q_0 \oplus q_2 \oplus q_2q_1 \oplus q_2q_1q_0 \oplus q_1 \oplus q_0 \quad (4.11)$$

Table 4.2 Precomputed value of multiplicative inverse

Q	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
q-1	0	1	3	2	F	C	9	B	A	6	8	7	5	E	D	4

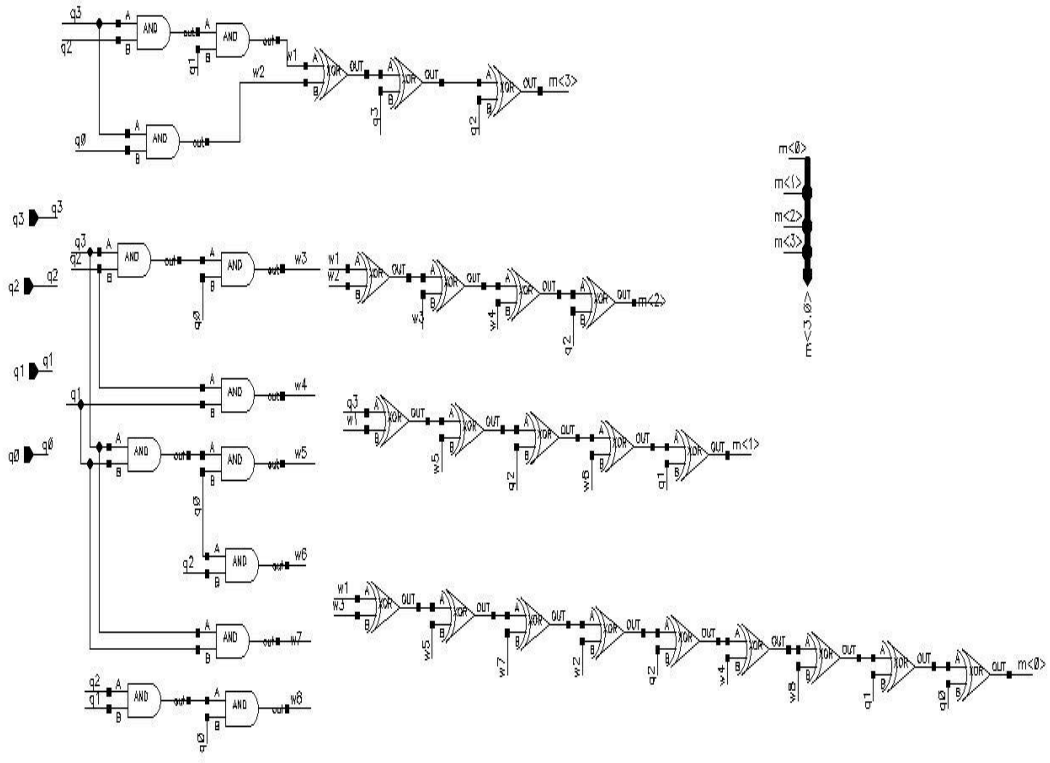

Figure 4.15 Schematic of Multiplicative Inverse in GF(2^4)

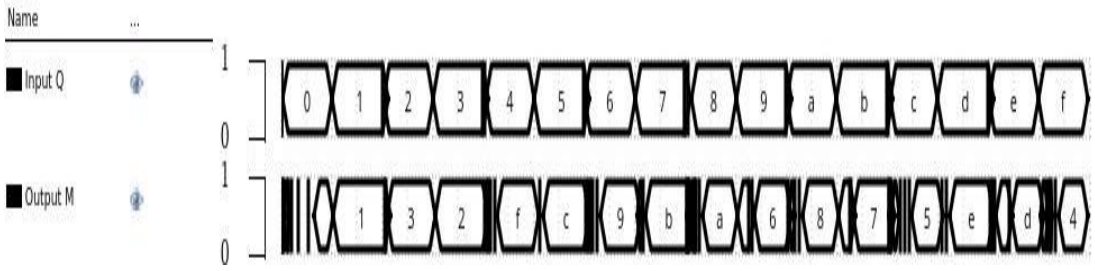

Figure 4.16 Simulation Result of Multiplicative Inverse in GF(2^4)

4.3.9 Inverse Isomorphic Mapping

The schematic of inverse isomorphic mapping presented in figure 4.17, requires only XOR to implement. 8-bit composite filed input [7:0]x mapped back to 8-bit finite field number [7:0]q given in simulation result in figure 4.18.

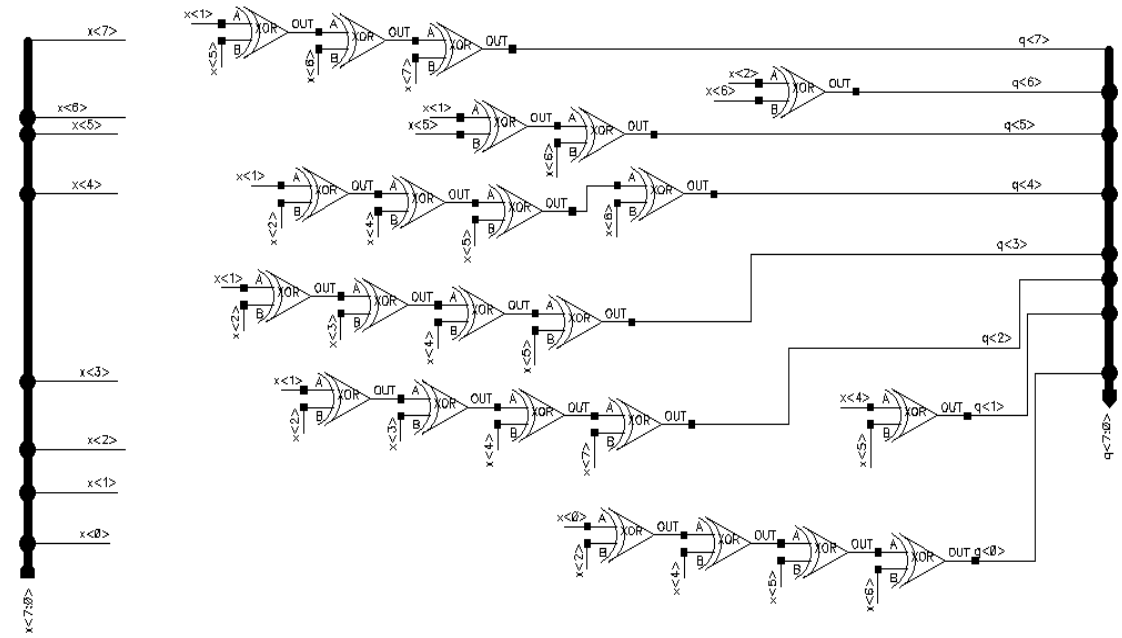

Figure 4.17 Schematic of Inverse Isomorphic Mapping

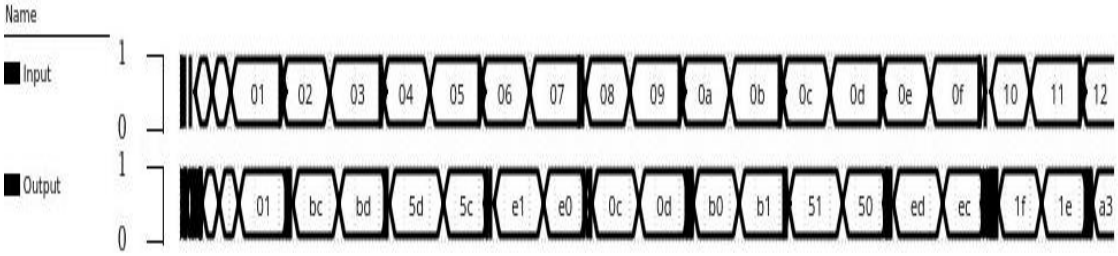

Figure 4.18 Simulation Result of Inverse Isomorphic Mapping

4.3.10 Affine Transform

Affine Transformation (AT) and inverse affine transform (AT^{-1}) defined by 8×8 matrix multiplication followed by XOR with constant. AT and AT^{-1} is defined as

$$y=ax+b$$

$$x=a^{-1}y+a^{-1}b$$

where a is 8×8 matrix and b is a constant byte $(01100011)_2$

$$AT(x)=\begin{pmatrix}1&1&1&1&1&0&0&0\\0&1&1&1&1&1&0&0\\0&0&1&1&1&1&1&0\\0&0&0&1&1&1&1&1\\1&0&0&0&1&1&1&1\\1&1&0&0&0&1&1&1\\1&1&1&0&0&0&1&1\\1&1&1&1&0&0&0&1\end{pmatrix}\times\begin{pmatrix}x_7\\x_6\\x_5\\x_4\\x_3\\x_2\\x_1\\x_0\end{pmatrix}\oplus\begin{pmatrix}0\\1\\1\\0\\0\\0\\1\\1\end{pmatrix} \tag{4.12}$$

$$AT^{-1}(x)=\begin{pmatrix}0&1&0&1&0&0&1&0\\0&0&1&0&1&0&0&1\\1&0&0&1&0&1&0&0\\0&1&0&0&1&0&1&0\\0&0&1&0&0&1&0&1\\1&0&0&1&0&0&1&0\\0&1&0&0&1&0&0&1\\1&0&1&0&0&1&0&0\end{pmatrix}\times\begin{pmatrix}x_7\\x_6\\x_5\\x_4\\x_3\\x_2\\x_1\\x_0\end{pmatrix}\oplus\begin{pmatrix}0\\0\\0\\0\\0\\1\\0\\1\end{pmatrix} \tag{4.13}$$

Hardware connection diagram of the affine transform present in figure 4.19 implement with only the XOR gate and the 8-bit simulated result shown in figure 4.20.

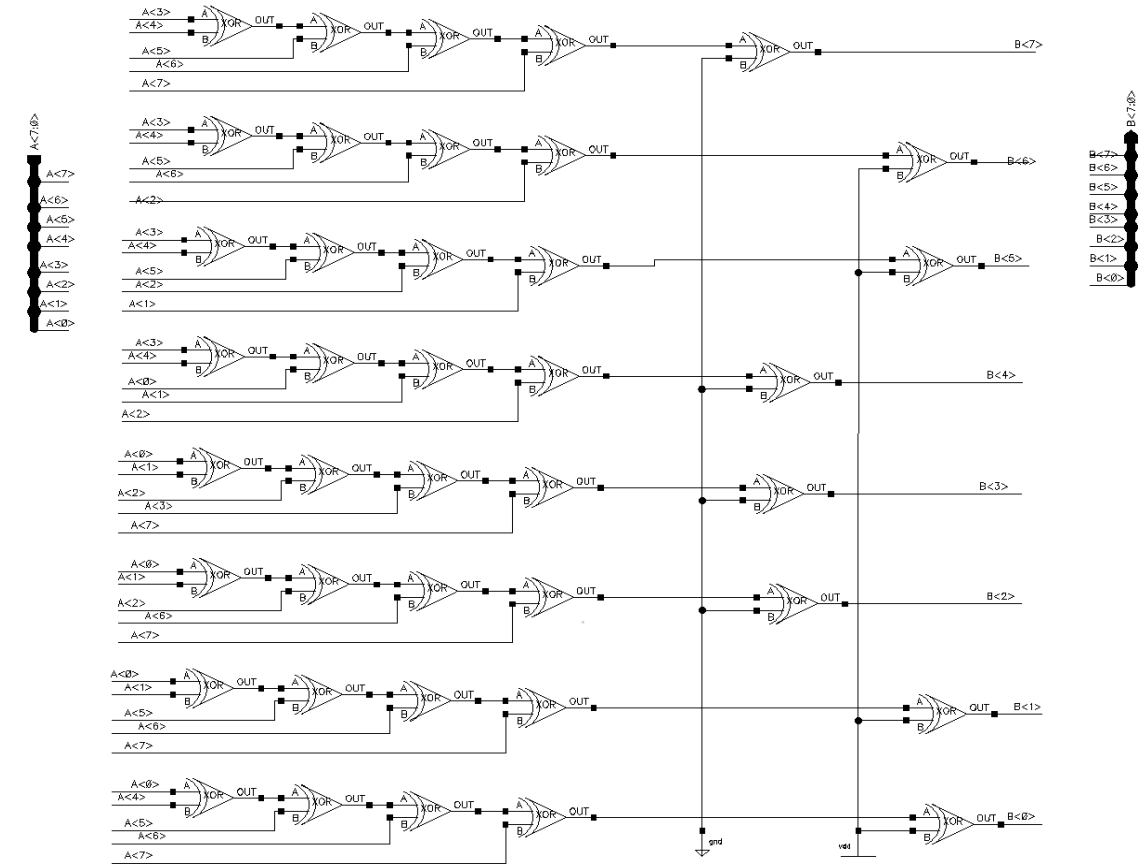

Figure 4.19 Schematic of Affine Transform

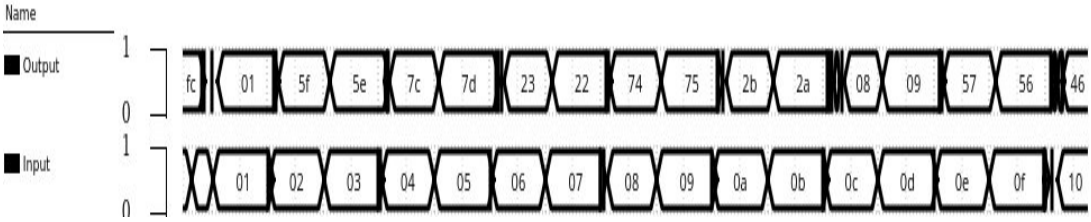

Figure 4.20 Simulation Result of Affine Transform

4.4 SBOX implementation with static CMOS logic

The performance parameter of each block of SBOX presented in table 4.3, multiplication in GF(2^4), is the most power-consuming block followed by isomorphic mapping and affine consume minimum power. Multiplication with constant(λ) takes maximum delay, among other components in SBOX. An interfacing diagram of SBOX with a worked example is given in [53, 93, 120] the figure 4.21, to compute the substitution of input

byte 05H isomorphic mapping map to composite filed 7DH further split into nibbles. Higher nibble 7H applies to squarer produces output 4H also multiply with constant(λ) $(1100)_2$ delivers output DH; low nibble DH XORed with upper nibble 7H yield AH. Multiplication in GF(2^4) multiple 7H and AH yield 06H. Modulo-2 addition of squarer output and GF(2^4) multiplier carried out with XORed output BH. The inverse block is the most consuming block of multiplicative inverse in terms of power and area. It adds nonlinearity into computation with constant value mentioned in table 4.2. The output of multiplicative inversion 07H multiplied with two units of GF(2^4). Multiplier another input to it is an output of isomorphic mapped value; input for upper GF(2^4) multiplier is 0H, and 7H produces output 0H.

Similarly, input to low GF(2^4) multiplier is DH, and 7H produces output 9H. Input to inverse isomorphic mapping stage 09H computes multiplicative inverse output to 1AH. By combining upper and lower nibble of multiplier output. Affine transform calculates the value to substituted as 6BH. The transient response of SBOX in figure 4.22 validates the design by comparing it with SBOX byte.

Table 4.3 power and delay report of an individual block of SBOX

SBOX Internal Block	Delay (s)	Leakage Power Consumption (w)	Average Power Consumption (w)
Isomorphic Mapping	418.3p	489.4n	20.26 µ
Squarer	89.68p	204.6n	6.02µ
Multiplication with λ	1.005n	69.02n	8.807µ
Multiplication in GF (2^2)	51.15p	170.2n	7.573µ
Multiplication in GF (2^4)	219.1p	6.12µ	121.7µ
Multiplicative Inverse	23.87p	910.9n	50.52µ
Inverse Isomorphic Mapping	486.8p	670.6n	32.36 µ
Affine Transform	357.4p	222.8f	201.9n

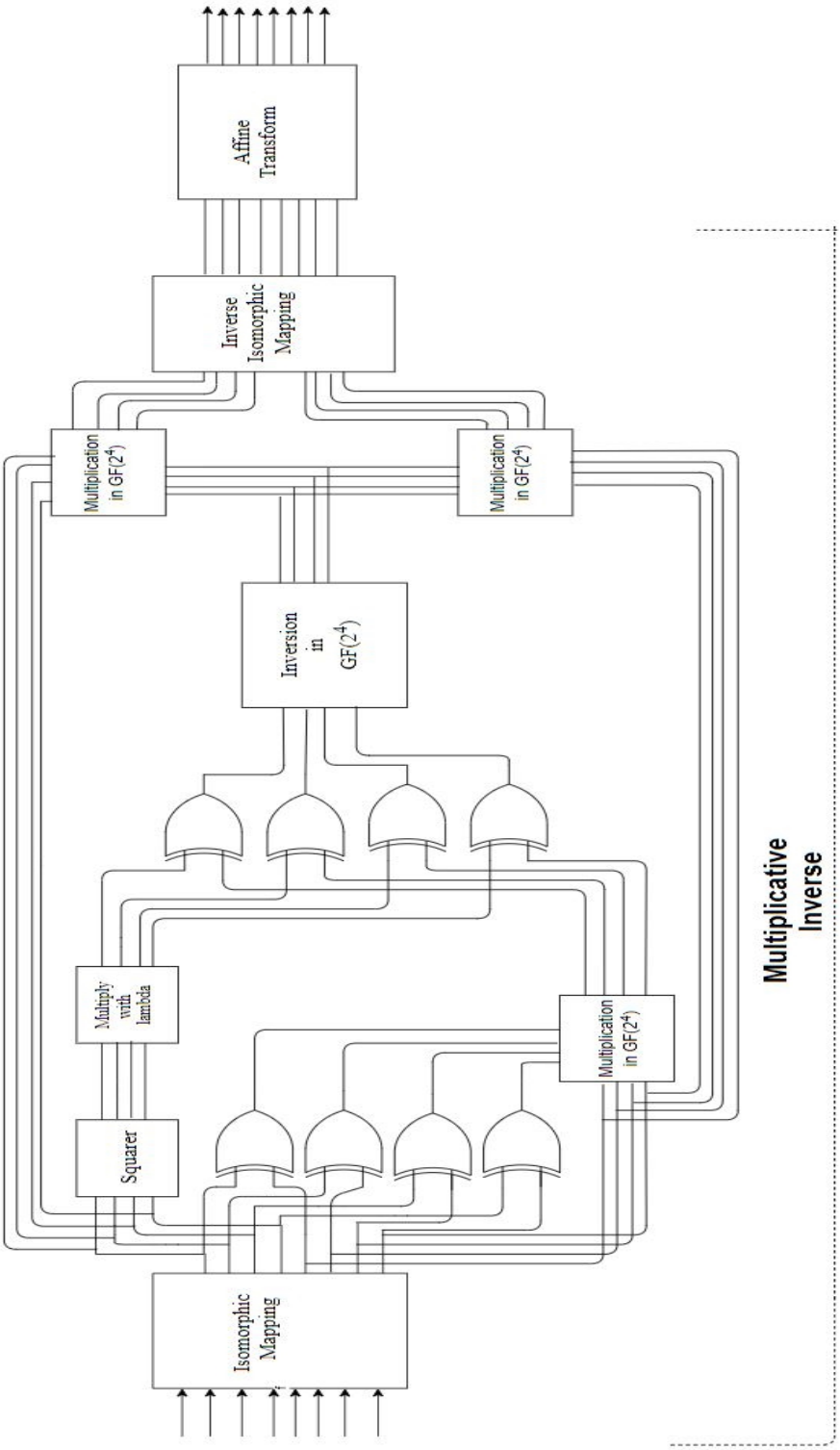

Figure 4.21 Interfacing diagram of SBOX

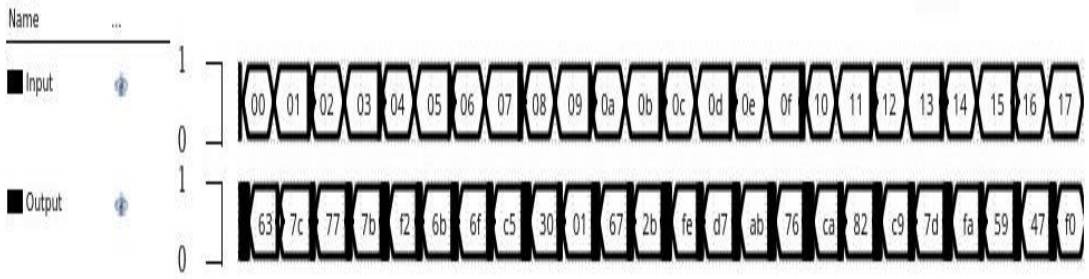

Figure 4.22 Simulation result of SBOX

4.5 SBOX implementation with hybrid CMOS logic

Conventional CMOS design requires a large number of the transistor to increase the area requirement. CMOS transistor-based logic gate is the default standard for low power digital design, while dynamic logic is faster than static logic requires a quiet area. Since the internal block of SBOX can implement with XOR and AND gate, the performance of SBOX can improve with the implementation of a cell with different logic design.

4.5.1 Pass Transistor Logic (PTL) XOR

Pass transistor-based circuit design used to reduce circuit complexity at the cost of voltage swing. The XOR gate's various topology, as illustrated in [60, 121], with a reduced number of transistors. Powerless and groundless PTL architecture achieves functionally of XOR limitation that occurs in output swing. PTL XOR presented in figure 4.23 requires only four transistors, compared to static XOR architecture transistor count drop by 66.67%. The output voltage degraded by threshold voltage; XOR output achieves V_{tp} for input 00, V_{dd} for 01, and 10 and ground for 11, respectively. Since degraded output doesn't use to control in subsequent stage threshold loss, don't degrade production further. It depends on they cause a further drop in the following stage or not. As shown in table 4.4, PTL XOR is slow bit shows a 4.7 fold improvement in power consumption.

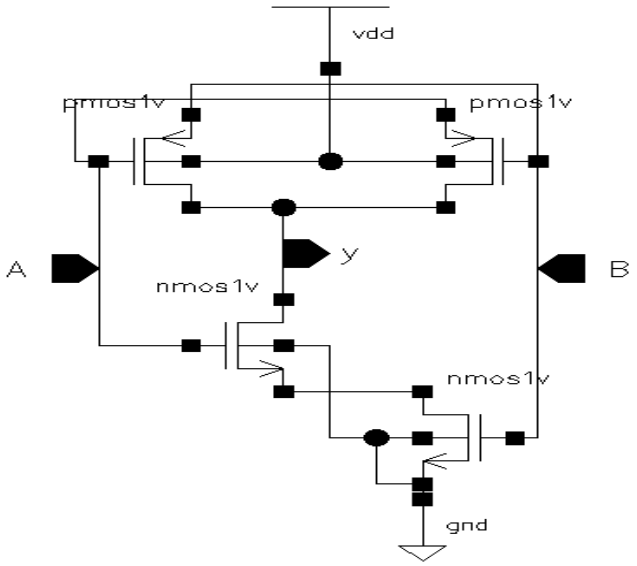

Figure 4.23 PTL XOR

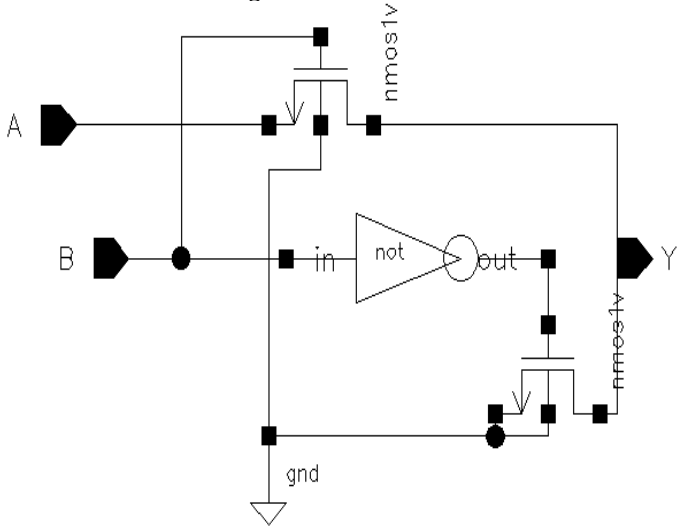

Figure 4.24 PTL AND

4.5.2 PTL AND

PTL, shown in figure 4.24 [122], implements boolean function with only NMOS transistors. PTL implementation with AND function with the same input at source and gate and output derive from drain terminal. For high input at a gate, terminal B passes An input to output. For low input at B, NMOS turns off, and the output terminal is floating. To correct the result, additional transistors in parallel provide low output. Compare to

static CMOS, PTL AND gate result in 33.33% area saving. PTL AND gate faster and save 15% power consumption compare to static CMOS.

4.5.3 Transmission Gate (TG) XOR

Implementation with a CMOS transmission gate shown in [123] provides a compact structure that requires a lesser number of transistors than conventional CMOS logic but larger than PTL. Transmission gate topology is more comfortable in developing the XOR gate. The transmission gate is a combination of P-type and N-type MOSFET control by a switch input shown in figure 4.25. Gate input of NMOS and PMOS are complementary signal; provides a parallel path between the input to output. A transmission gate acts as a bidirectional switch controlled by gate input S. When S=0 and S_bar=1 both transistor turns ON offers a low resistance path, PMOS passes strong '1', and NMOS passes strong '0'. For S= 1 and S_bar=0, both MOS turns off, high impedance state.

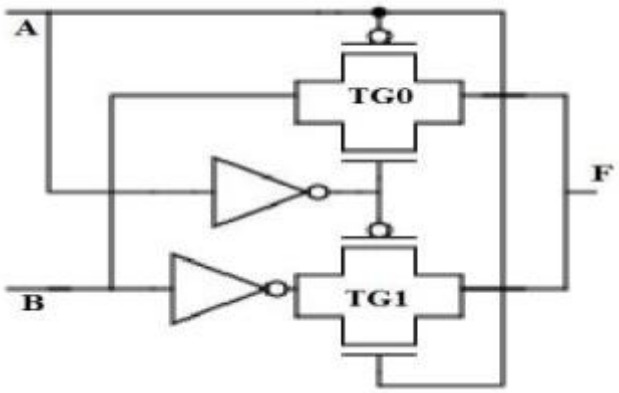

Figure 4.25 TG-XOR

XOR implementation of TG requires eight transistors compare to 12 in static CMOS logic. A TG-XOR requires two transmission gate and to the inverter when B=0 TG1 turns on, and TG0 turns OFF to produce half of the result (~AB). Similarly, B=1 TG1 turns off, and TG0 turns off to create another half result (~BA). The shorted drain terminal of both TG combines both results employing hidden wired-OR logic implement ~AB+~BA.

4.6. Result and Analysis

In this work, we have carried out a substitution box schematic with a different CMOS logic style. Implementation of internal cell XOR-AND with different logic yields a comparative summary report. Table 4.4 found that PTL and TG logic is an excellent choice to minimize the number of transistors compared to static architecture. Delay is minimal in static architecture since power and ground are available, while in other logic, it exceeds. More specific delay of XOR cell is more significant than AND cell. Hybrid logic combination of TG and PTL logic reduces the static power and dynamic power consumption due to the reduction in gate count and no direct short path found between the power to the ground rail. PTL logic significantly reduces the static current.

Table 4.4 Comparison of XOR-AND cell in different CMOS logic style

Parameter	Static Architecture		Pass Transistor Logic		Transmission Gate
	XOR	AND	XOR	AND	XOR
Delay (ns)	0.005	0.01929	1.016	0.0103	1.017
Total power (nw)	2342	745.6	495	629.9	211.446
Static power (nw)	1.152	25.58	1.037	15.51	7.146
Dynamic power (nw)	2340.848	745.58	493.963	614.39	204.3
Transistor Count	12	6	4	4	8

A comparative result of SBOX implementation with static CMOS and TG-XOR and PTL AND presented in table 4.5. SBOX design with TG-PTL logic is more expensive in the subject of power requirement and delay; it shows an advantage in terms of transistors count compared to static SBOX architecture. The number of BSIM3v3 model transistors requirement to implement SBOX with the static CMOS logic, PTL logic, and TG+PTL logic is 2316, 868, and 1544 respectively. The presented design of SBOX verified at 1000 MHz frequency; power consumption is increased approximately 4-fold compared to [29], SBOX delay is lower hybrid logic and highest for PTL logic. PTL and TG logic effectively minimize cell counts in the design but hurt power consumption and delay. Compare to the static logic, power consumption of PTL logic decreases from 216.2 μW

to 133.2μW due to the absence of supply rails and increases to 1339 μW for the hybrid approach. Hybrid save the delay 0.43 nS compare to static logic but enhance to 1.256 nS for PTL logic. Trade-off exits between power delay concerning the area. Measured PDP and EDP of presented SBOX design are lower for SBOX with static cell, PTL and TG reduces the area but sacrifices the other.

Table 4.5 Comparison of SBOX in different CMOS logic style

	Static XOR-AND	PTL XOR-AND	TG XOR-PTL AND	[30]	[118]	[56]	[63]	[71]	[8]
Technology (nm)	90nm	90nm	90nm	65nm	65nm	130nm	65nm	250nm	130nm
Supply (v)	1	1	1	--	--	1.2	0.8	2.5	1.5
Frequency (MHz)	1000	1000	1000	763	610	10	10	10	10
Delay (ns)	0.767	1.256	0.734	1.31	1.64	3.3	7.5	9	2.53
Average power (μw)	216.3	133.2	1339	54.99	44.39	12.1	0.037	140	179
PDP (fJ)	165.9	167.3	982.8	72.04	72.8	39.93	0.278	1260	452.87
EDP (zJs)	127.24	210.13	721.4	94.37	119.39	131.77	2.085	11340	1145.8
Transistor Count	2316	868	1544	--	--	--	--	--	--

-- Not Available

4.6.1 Dependency of CMOS power consumption with processed data

The power analysis attack at SBOX starts with analyzing the peak value of power to extract the secret key during computing the nonlinear byte during the substitution byte process presented in [130]. A designer's objective is not only to reduce power consumption but also to include security parameters at the design level. The circuit should not reveal the power consumption; i.e., power consumption should free from the input data explained in[65]. The power attack resistant feature of the VLSI circuit is measure with and two energy-based parameters discussed in [49] Normalized Energy

deviation (NED) and normalized standard deviation (NSD) calculated from equation (4.14). NED is the percentage between maximum energy consumption (E_{max}) and minimum energy consumption (E_{min}) for all input combinations of all possible transitions [131]. NSD measure how much energy consumption varies for each input. NSD is it presents how consumed energy distributed around the mean, the larges value of NSD indicated energy widely spread across mean smaller value of NSD indicates they are close to the way. Ideally, NED-NSD approaches zero for better resistance to a power analysis attack.

$$NED = \frac{E_{\max} - E_{\min}}{E_{\min}} \quad NSD = \frac{\sigma_E}{E_{avg}}$$
$$\sigma E = \sqrt{\sum_{i=1}^{n} \frac{(E_i - E_{avg})^2}{n}} \quad E_{avg} = E_{\max} - E_{\min} \tag{4.14}$$

NED-NSD measures presented in [52] how much energy value changes for each bit change in input. It offers a complexity level for an attacker to guess the secret correctly. Lower value gives a more complicated measurement setup. Table 4.6 compares the energy resistance feature of static, TG-PTL, and PTL logic SBOX. SBOX with TG XOR shows the minimum value of NED, while PTL and static logic have a wide range of energy distribution. Deviation of energy around average cost is almost similar in static and TG-PTL logic, while PTL SBOX scores the maximum value of NSD. PTL SBOX shows a significant amount of energy for even a single bit change in input. TG-PTL based SBOX power profile shows low dependency on the processed data.

Table 4.6 Comparison of energy parameters in SBOX topology

Parameter	Static SBOX	PTL SBOX	TG-PTL SBOX
Emin (fJ)	251.2633	4.6845	2.219
Emax(fJ)	1734.2256	1762.72	1664.78
Eavg(fJ)	1482.9623	1758.04	1667
E_{SD}(fJ)	39.47	93.3382	34.6
NED	0.855	0.9973	0.4391
NSD	0.0266	0.0531	0.0208

Chapter 5
Power attack analysis

Cryptanalysis is a method to identify the weakness of the cryptographic algorithm and to learn the secret key. Here, knowledge of algorithms is unnecessary; the attacker tries to gain access over devices based on information obtained from hardware limitation rather than the algorithm. The CMOS-based low-power circuit design is the default standard of the standard cell library of the EDA tool [132, 133]. In addition to primary output devices do emit secondary information in terms of power, time, electromagnetic radiation, fault, etc. shown in figure 5.1, known as side-channel information. Analysis of this information to obtain the hidden secret is the side-channel attack. Since the cryptographic algorithm is known to everyone, input plain text message and ciphertext message may or may not be known, but the secret key must be secured. The attacker wants to guess the secret key from side information and the available input message.

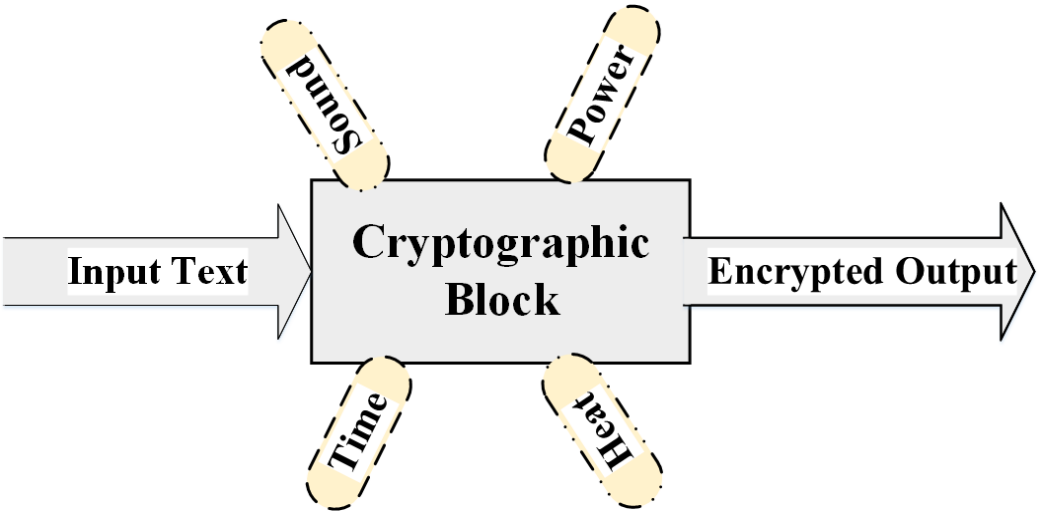

Figure 5.1Source of side-channel information

Power attacker analysis is the most explored side-channel attack. Power attack analysis is a category to side-channel analysis that monitors the hardware's external power measures while a cryptographic operation carried out. The goal of the attack is obtaining information result in comprising of security. As MOS transistor shrinks to the nanometer

region, leakage power increases. The switching/dynamic power of the CMOS transistor is dependent on the input data to be processed. A device consumes power while it performs computation. Power consumption is unique for each input since the power consumption profile maintains relations with the input message. SBOX is identified as a point of interest to obtain to store the [134] power consumption for corresponding input and output. From the SBOX output and their power value, adversaries try to guess the input of SBOX, and on XORed with input message, it is possible to predict the secret key. It is a challenging task to store the power trace for each possible input. A small value of resistor 50 ohm connected in series to the power supply. Current flows through the resistor are measures and analyses with a waveform analyzer or oscilloscope. Power attack analysis completed in 2 phases, 1^{st} phase is data gathering where the power consumption database created for each possible input and output and 2^{nd} phase to data analysis where the input of SBOX tries to obtain from the power trace and output of SBOX.

5.2 Overview of CMOS Power consumption

CMOS implementation of digital circuit preferred due to low power consumption features. The circuit consumes a definite amount of power when it performs computation. The author in [128] presents the power consumption pattern depends on input data. Power consumption of the CMOS circuit classified into 3 categories, static power (P_s), dynamic power (P_{dyn}), and short circuit power (P_{sc}). As shown [43, 64], CMOS, not gate consists of PMOS and NMOS in series. For low input PMOS on and NMOS cutoff, the load capacitor charges up to maximum value and stores energy. For high gate input NMOS conducts and PMOS cutoff stored energy in capacitor discharges through NMOS.

5.2.1 Static Power

Static power refers is the amount of power requires to withstand the circuit in the given condition. Power supply and ground rail applied to the circuit, but not performing

switching activity a small current flow in IC known as static power and power consumption due this current is static power. It also termed as leakage power. P_s=0 Ideally.

$$P_s = I_s \times V_{DD} \tag{5.1}$$

Where I_s is due to reverse bias current between the source and drain to the substrate.
Static current is given as

$$I_s = K\frac{W}{L}e^{(\frac{V_{GS}-V_{TH}}{\eta V_T})}(1-e^{\frac{V_{DS}}{V_T}})(e^{\frac{V_{SB}}{V_T}}-1) \tag{5.2}$$

Where V_{TH} is the threshold voltage, V_T is the thermal voltage, V_{GS}, V_{DS}, and V_{SB} present the operating voltage at a corresponding junction. The device's static current depends on circuit topology and static voltage at the terminal [129].

5.2.2 Short circuit power

When input falls from low to high, PMOS changes its operating region from saturation to cutoff, and NMOS changes from cutoff to saturation. Since MOS transistor cannot change region quickly offers non zero rises or fall time. During switching, both PMOS and NMOS conducts for a small period. A DC path found between power and ground rail, a peak current flow between a short-circuited path for a small period. Power consumes due to this current is short circuit power.

$$Energy\,consumed\ per\ switching\ activity\ period\ E_{sc} = V_{DD} I_{peak} t_{sc} \tag{5.3}$$

$$Power\ consumption\ per\ cycle\ P_{sc} = V_{DD} I_{peak} t_{sc} f_{clk} \tag{5.4}$$

5.2.3 Dynamic power

Source of power analysis attack is information leaks through dynamic power. Dynamic power is data-dependent maintain relation with power peak and input bits. The amount of power requires a circuit to switch from one state to another described in [71]; when the output node switches, high node capacitor charges up to V_{DD} draw energy from the supply. Half of the energy spent across PMOS as heat and remaining half energy stores in the capacitator. During high to low switching of output C_L discharges and stored energy dissipates through NMOS to ground.

$$\text{Energy is drawn from the supply } E_{V_{DD}} = \int_0^{\infty} i_{V_{DD}}(t) V_{DD}(t)\,dt$$

$$= V_{DD} \int_0^{\infty} C_L \frac{dvout}{dt}\,dt$$

$$= C_L V_{DD} \int_0^{V_{DD}} dvout$$

$$= C_L {V_{DD}}^2 \qquad (5.5)$$

$$\text{Energy stored in capacitor } E_c = \int_0^{\infty} i_{v_{DD}}(t) V_{out}(t)\,dt$$

$$= \int_0^{\infty} C_L \frac{d_{vout}}{dt} v_{out}\,dt$$

$$= C_L \int_0^{V_{DD}} v_{out}\,dv_{out}$$

$$= \frac{C_L {V_{DD}}^2}{2} \qquad (5.6)$$

A switching cycle takes a fixed amount of power $C_L V_{DD}^2$. If the gate is switching ON to OFF α time per second dynamic power is given as $P_{dyn} = \alpha C_L V_{DD}^2$ where α is switching activity at output terminal measure as the probability of several transitions leads to energy dissipation to the number transition. Its maximum frequency of input is f_{clk} dynamic power modifies as equation (5.7)

$$P_{dyn} = \alpha f_{clk} C_L {V_{DD}}^2 \qquad (5.7)$$

Switching activity factor α measures the number of time input changes per second. The activity factor is a data-dependent parameter. An n input circuit takes 2n input combinations; dynamic power is consumed by a circuit for a small period when the input changes one state to another; the remaining input combination consumes only static power or leakage power. For CMOS inverter are four transitions occur, and corresponding consumption shown in table 5.1. Small static power dissipated for 0-0 and 1-1 transition [51]. During 0-1 transition load capacitor charges with PMOS and discharges through NMOS while 1-0 transition. Charging and discharging phenomena of capacitor consumes substantial dynamic power. Static $P_{0\text{-}0}$ and $P_{1\text{-}1}$ are lower than dynamic power $P_{0\text{-}1}$ and $P_{1\text{-}0}$. Power requires to transition 0-1 is more significant than to transit 1-0; $P_{0\text{-}1} > P_{1\text{-}0}$.

Table 5.1 Power consumption pattern of CMOS Inverter with respective bit

Input	Output	Power
0	0	P_s
0	1	P_s+P_d
1	0	P_s+P_d
1	1	P_s

5.2.4 Power delay product (PDP)

PDP measures energy consumption per switching event in joule. If circuit switches with maximum possible rate $f_{max}=1/2t_p$.

$$PDP = C_L V_{DD}^{\ 2} f_{\max} t_p = \frac{C_L V_{DD}^{\ 2}}{2} \tag{5.8}$$

5.2.5 Energy delay product (EDP)

PDP measures the energy needed to switch; this value is small by reducing the supply voltage. Since energy and delay related to supply voltage, high supply reduce delay but increase energy and vice-versa.

$$EDP = PDP \times t_p = \frac{C_L V_{DD}^{\ 2}}{2} \times t_p \tag{5.9}$$

5.3 Classification of Power Analysis Attack

Power attack analysis is classified [135, 136] as simple power analysis (SPA), differential power analysis (DPA), and correlation power analysis (CPA).

5.3.1 Simple power analysis

SPA participates directly interpreting power traces, collected by a cryptographic module, and try to deduce the information. Power trace is a 2-dimensional waveform of power consumption measurement concerning time. It consists of power measurements taken across the cryptographic operation. Power trace AES is shown in figure 5.2. Since AES has ten rounds, the centre portion of the curve repeats for ten rounds, it's quite a

problematic guess information by only visual inspection. SPA not considered a full-power attack. The key used in encryption is determined using the probability analysis method.

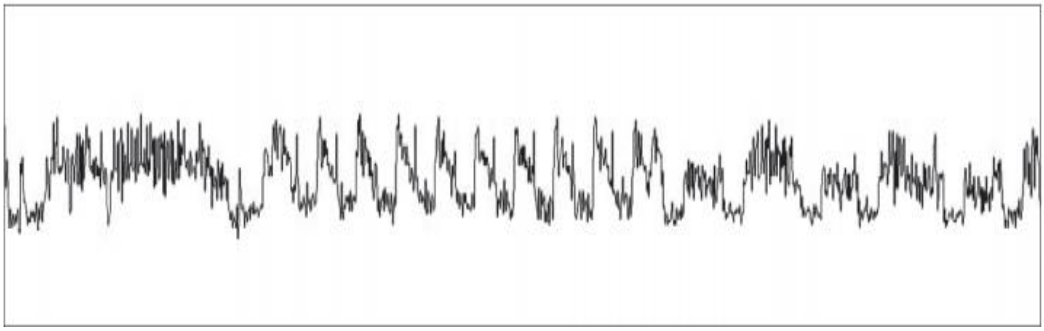

Figure 5.2 Power trace in SPA [103]

5.3.2 Differential power analysis

DPA has considered a more vigorous attack that needs a large number of power traces to compare to SPA. DPA utilizes the data-dependent power consumption pattern in the power trace. DPA can reveal the secret key, even in a noisy environment. The power requirement of the circuit is the function of internal state voltage. Power trace is a 2D waveform concerning time, presents the power amplitude variation for different input. DPA daring feat the relation between power traces and internal node voltage level due to the applied input. DPA starts with the assumption that input plain text or output of SBOX is known, accomplished in 2 step data acquisition and data analysis. Figure 5.3 presents a flow chart of the data acquisition and analysis phase. In the data acquisition phase, a large number of power traces need to store by performing a large number of simulations.

During encryption [111, 137], a large number of plain text data inputted to the SBOX circuit with a fixed secret key and power waveform analyzed with an oscilloscope or waveform analyzer. During data analysis, attackers assume a secret key between 00-FFH and compute the internal state voltage by comparing it with ciphertext. DPA attacks explained in [138] can be applied to SBOX first round or last round; since implementing the attack on the previous round, it is difficult to store and analyze power traces on each round. Convenient to apply DPA attack on the first round of SBOX. During the data

analysis phase, the collected trace divided into two groups corresponding to internal state voltage. The average waveform is computed in each group and finds the difference between both. The differential curve of average power waveform observed to deduce the information; If spike was seen in the differential waveform, then the assumed key is correct else waveform is constant for the incorrect key [42]. This procedure repeated for all possible 256 keys, here focused on an 8-bit partial key of a 128-bit key.

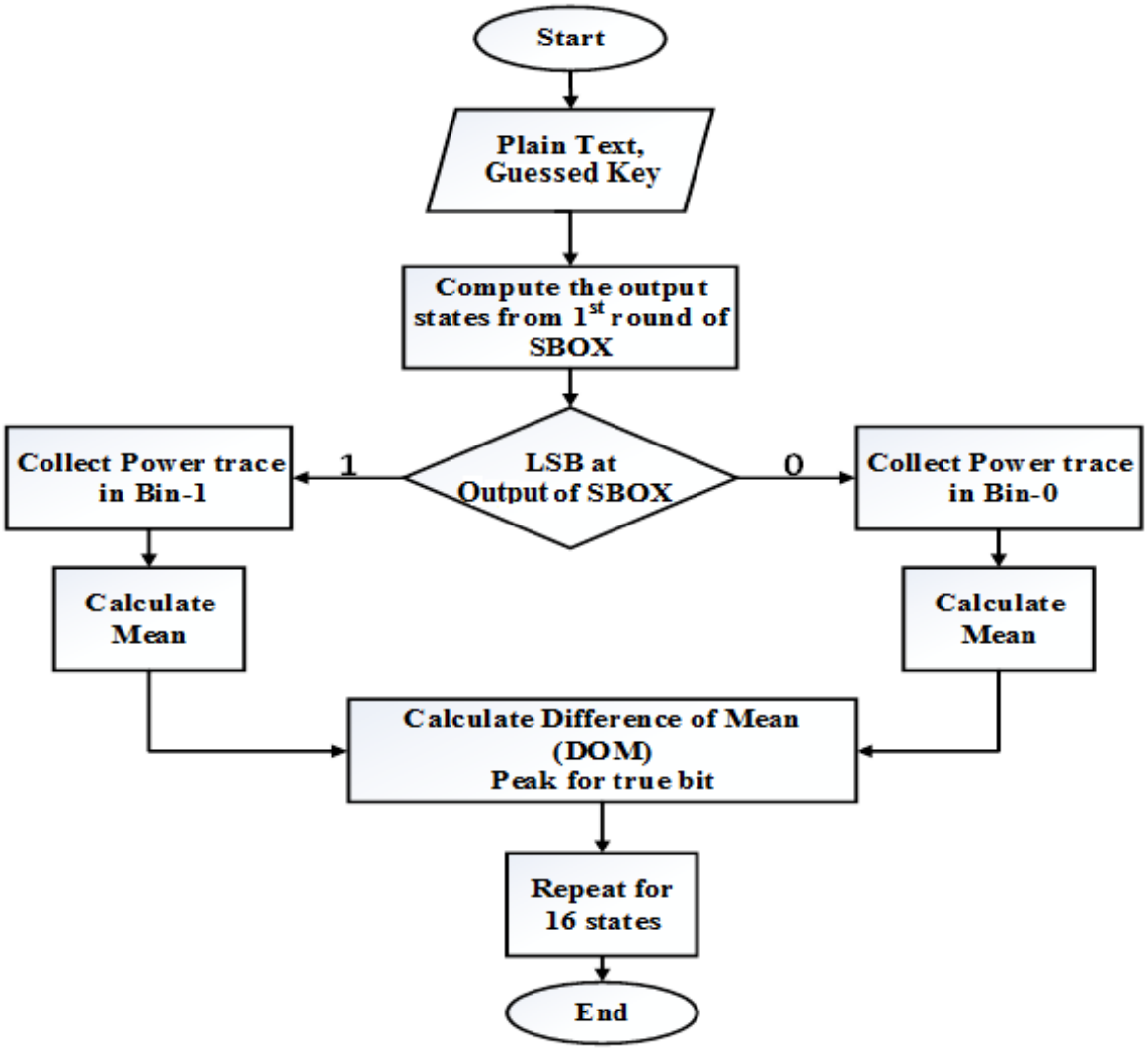

Figure 5.3 Flow chart of DPA [159]

5.3.3 Correlation power Analysis

The correlation power attack introduced in [42,54 and 153]; the Pearson correlation coefficient is used instead of the mean difference. CPA attack is a more precise and

accurate attack compare to DPA; Correlation power attack utilizes a mathematical power model to predict power consumption. These predictions are compared to real power consumption to exploit the secret key: Hamming weight and hamming distance currently used to have an efficient power model. CPA finds the relation between actual power consumption and hypothesized power model, i.e., hamming weight or hamming distance power model. The hamming weight power model estimates the power consumption of a circuit with the assumption that 0 doesn't lead to power consumption while one involves a significant amount of power. Here power is proportional to the number of high bits in the processed data. HW measures the number of 1 inputting to the SBOX, and HD is the measure of number bit changing from 0-1 and 1-0 at the input of SBOX. The waveform classified according to HD calculated from key shows dependency on the HD. The increasing value of HD shows the high-power consumption, and the lower value of HD implies low power consumption. The correct key revealed by searching the key with the highest correction between actual power consumption and model power value. A higher correlation between actual power trace with hypothesized power value implies it is highly probable to predict the secret key. Figure 5.4 presents the flow of data during the CPA experiment.

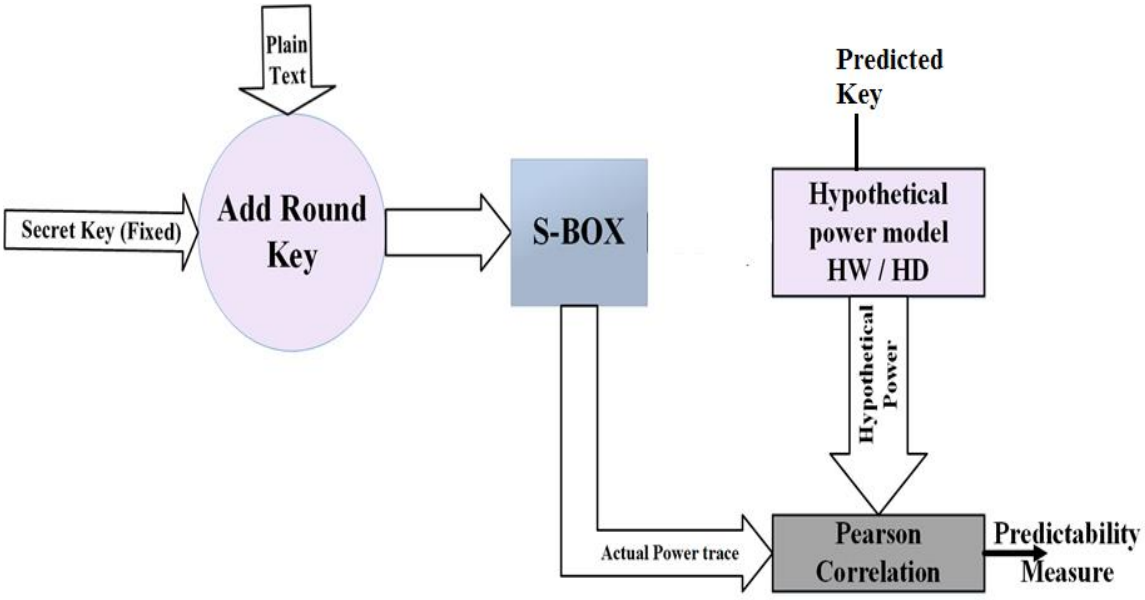

Figure 5.4 Flow chart of CPA

The correlation coefficient between actual and model power consumption calculated as equation (5.10)

$$\rho(P_{actual}, P_{predicted}) = \frac{\text{cov}(P_{actual}, \text{P}_{predicted})}{\sqrt{\text{var}}(P_{actual})\sqrt{\text{var}}(P_{predicted})} \quad (5.10)$$

Where cov(P_{actual}, $P_{predicted}$) is the covariance between actual and predicted power consumption, var(P_{actual}) and var($P_{predicted}$) is the variance of actual and predicted power value for all possible input. The coefficient of correlation measures the relation between P_{actual} and $\rho_{predicted}$ ranges between -1 to +1. ρ = -1 measure inverse relationship between P_{actual} and $\rho_{predicted}$, ρ = 0 measure P_{actual} and $P_{predicted}$ are uncorrelated, ρ = +1 measure positive and the relation between P_{actual} and $P_{predicted}$, for correct key guess the high value of ρ expected. Correlation coefficient reduces the possibility to select the wrong key, in an 8-bit key if a single bit is wrong; correlation is reduced by 1/4. If all bit wrong, anticorrelation observed. For a positive value, a higher correlation observed.

5.4 Power Model

Differential power analysis attacks briefed in [56, 127] based on determining the key used for encryption from the mathematical model, which can approximate the power requirement of the circuit. Implementation of a power attack depends on the accuracy of the hypothesis power model. Since cryptographic architecture is known, the simulation result provides a precise prediction power consumption of the circuit. Cadence ADEL browser window plots the power consumption for each input; a software environment needs to designed to shows the power consumption for corresponding input and output. Two widely used model is hamming weight and hamming distance power models.

5.4.1 Hamming weight Power model (HWPM)

HWPM approximates the power requirement of the circuit from the input data. Since power is proportional to the number of high bis in the data shown in figure 5.5, this model helps approximate the power value of a circuit when the consecutive input data is

unknown. Here low bit in input data don't consume power and high bit contribute to power consumption value. If input data have all bit zero, power consumption is minimum, and power consumption is maximum for all input bits are high. HW model extracts minimal facts about the circuit; weekly describes the power utilization of the circuits. Let R is the input data at the input terminal if circuit power consumption approximated as equation (5.11).

$$W = aHW(R) + b \tag{5.11}$$

Where a and b are constant.

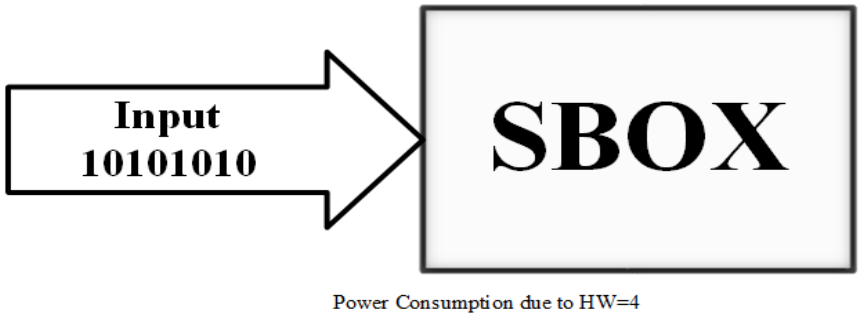

Figure 5.5 Power approximation with hamming weight

5.4.2 Hamming distance power model (HDPM)

The hamming distance power model is a convenient method of approximate power consumption. Hamming distance is a prolonged form of the hamming weight power model based on the number of bits switching at the input terminal to estimate power consumption. HD model of [53, 68] approximates the power utilization of the circuit proportional to logic transition 0-1 and 1-0. The number of transitions measured as the hamming weight of XOR of two values. Assumption of hamming distance power model is bit transition 0-0 and 1-1 does not contribute the consumption power, and 0-1 and 1-0 consume an equal amount of power. For an n-bit circuit hamming weight presented as the number of bits pull to 1, $HD = \sum_{i}^{n-1} D_i$ HD. D consists of n independent and uniformly distributed number of bits; average hamming weight = n/2 and variance = m/4.

Assume a constant value of the present state of register R_0 (should not zero) with the application of other input to XOR register bits flip their value R_1 presented in figure 5.6. As register value is updating hamming weight of XOR of initial and final value is proportional to approximate power consumption value If R_0 and R_1 is the two-consecutive value at the input terminal of circuit, power value approximated as equation (5.12)

$$W = HD(R_0, R_1) = aHW(R_0 \oplus R_1) + b \tag{5.12}$$

The hamming distance power model presents the relationship between power or current consumption and hamming weight of transition at the input. This model doesn't represent the total power utilization of chip but only power due to the data-dependent part. In CMOS circuit power consumption due to change in input, usually dynamic power fits quite well with other parts of power, ie. Static power, power due to noise, crosstalk, or wire isolation is a constant value denoted term b. Where a is scaler gain between hamming distance (HD) and power consumption and bis constant due to noise present in the circuit if all bit R_0 or R_1 =0 hamming distance model is equivalent to the hamming weight power model, consider a=1 and b=0 equation (5.12) simplifies to equation (5.13).

$$W = HW(R_0 \oplus R_1) \tag{5.13}$$

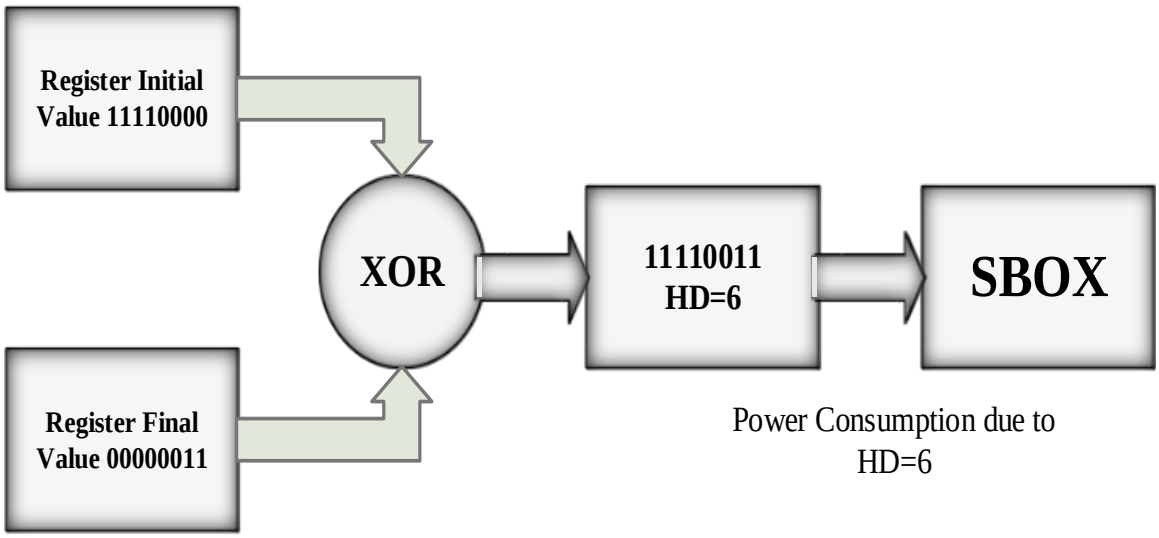

Figure 5.6 Power approximation with hamming distance

The dependency of power utilization of CMOS circuit with hamming weighs and hamming distance is identified from figure 5.7 and figure 5.8 [62]. CMOS cell draws current from the supply when switching takes place; almost zero current drawn in case of no switching. Drawn current increases as the number of bits in input data (HW) is increased, similarly current increase as the number of bits switching (HD) increases. HW and HD value eight curves are falling since the occurrence of HW=8 once and changing of 8 bits is once in a combination.

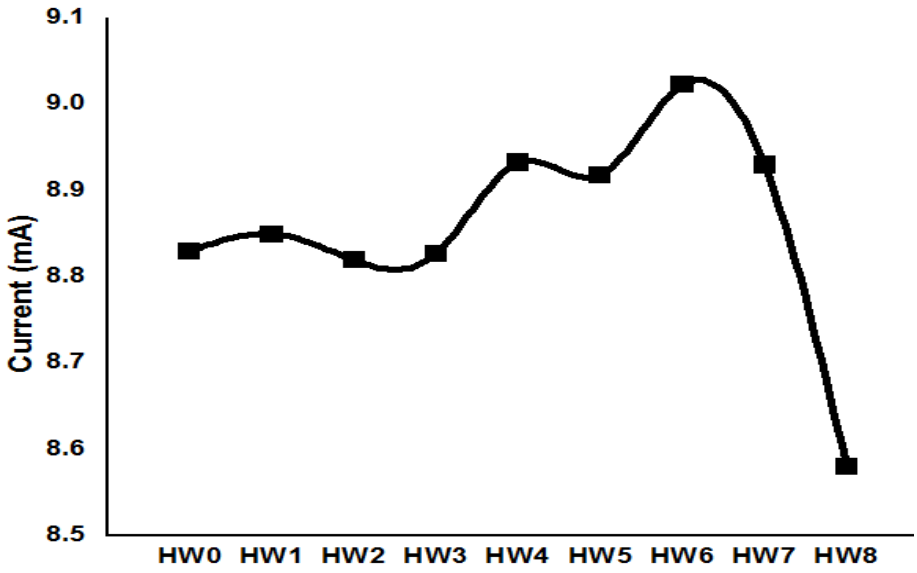

Figure 5.7 Current vs Hamming weight (HW)

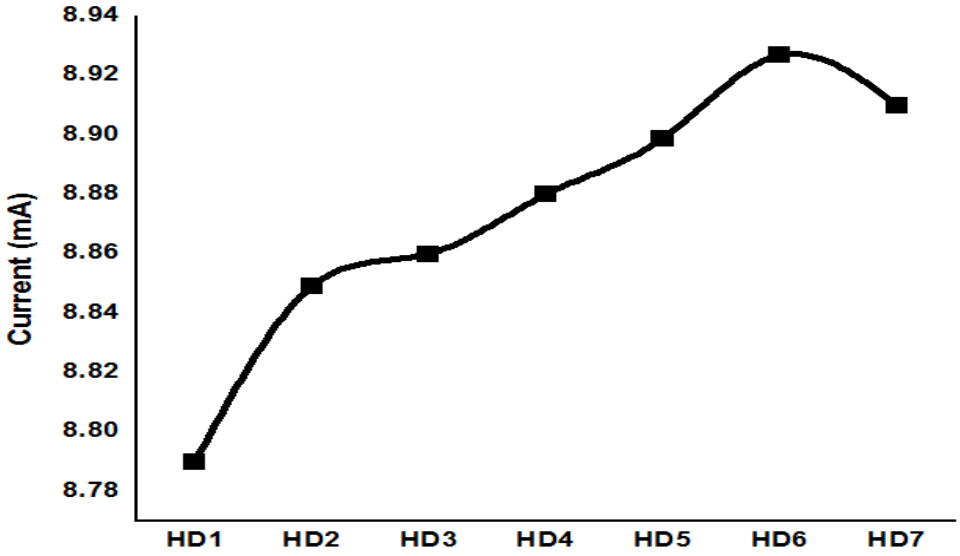

Figure 5.8 Current vs Hamming distance (HD)

5.5 Experiment with DPA

AES subsystem can attack from the first round or last round. In the first-round attack, the plain text applies to round key and SBOX power trace capture for analysis. In contrast, in the previous round attack, the previous round key and last SBOX need additional steps to reverse the encryption process. From [46, 139, 140] ciphertext reverses encryption, power trace since the previous round to first-round SBOX computes the information used for capturing and analysis. It presented in [141] the first round of SBOX is used to perform the DPA attack. Another reason to choose the first-round attack is simulation time. In the first-round, it necessary to simulate the whole encryption process. Schematic of Add round key and SBOX designed on CMOS 90nm technology with Cadence virtuoso schematic composer at 1-volt supply voltage.

Cadence spectre used for simulation and waveform analyzer needed to analyze the necessary 2056 power traces to implement the DPA attack. 128 bits of key and plain text divided into independent 16 SBOX. Each SBOX operates on separate byte reduce power trace from 2^{128} to $16\times2^{8}=4096$. Hence to break the security of an SBOX, it requires only a 4096-power trace. DPA attack applies statistical technique difference of mean to deduce information from the power trace. The power consumption of CMOS circuits is proportional to the number of high bits in the input. Assume input to SBOX is known, perform simulation of all possible key, and the plain text was chosen randomly from 00-FFH. Store the power trace for each input and the corresponding output. In the data analysis phase, the attacker tries to guess the secret key statistically.

In this experiment, we have assumed the secret key 25H and applies all possible 256 combinations of input text 00H-FFH to the AddRound key stage. The output of AddRoundkey grouped to 16 bytes; each byte applied to SBOX. In the SubByte stage, SBOX substitutes the output byte and corresponding power trace. Figure 5.9 is the power trace of the first SBOX with secret key 25H and plaintext 00-FF. The simulation runs for 2.56 μs for each plain text; each input applied for 10 ns. Clip function in the cadence virtuoso calculator selects the waveform at a given time. Clip function used 256-times on each power traces for 0-10 ns, Clip 256 power traces from a long power trace for each

corresponding plaintext separately. Figure 5.10 presents the power trace of SBOX corresponding to plain text 0-FH, i.e., first 16-byte. The device under test (DUT) consumes more power for output's LSB 1 compare to LSB 0. Power traces need to group into two bins, with their LSB. Plain text with LSB 0 stored in bin-0 and LSB 1 stored in bin-1, respectively. Calculate the mean of power trace in each bin separately. The average power trace is shown in figure 5.11 and 5.12 on each bin respectively. Calculate the mean of the waveform, power trace should be of the same length, point to point mean calculated for each input for a simulation time of 10 ns. Bin-0 and Bin-1 follow the same pattern. The differential power curve presented in figure 5.13 is not similar to the original power traces. The DPA curve should approximate to straight-line, except where the circuit performs computation, a spike in the waveform appears in case the successful key has been a guess. With differential power traces, one tries to deduce whether vital information can obtain for the proposed hypothesis. If the spike in the waveform observed presents, the key is the correct and straight-line in the difference of mean curve for the wrong key.

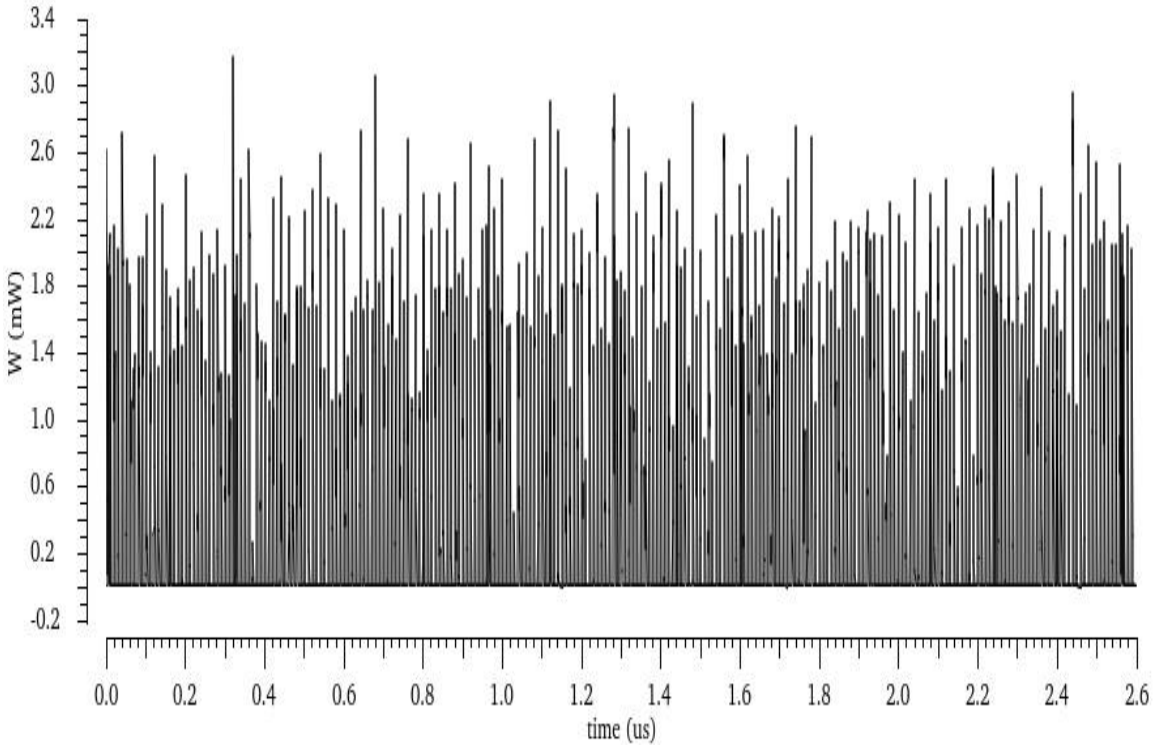

Figure 5.9 Power trace of SBOX during DPA

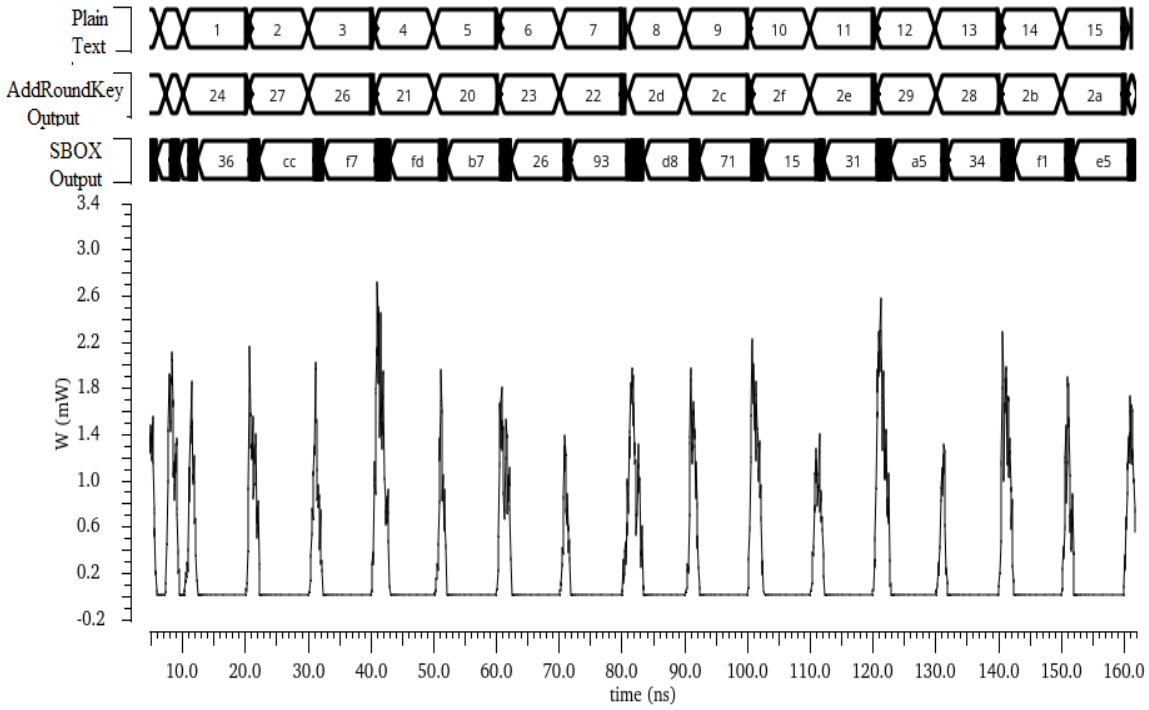

Figure 5.10 Power trace of SBOX during DPA for first 16 input

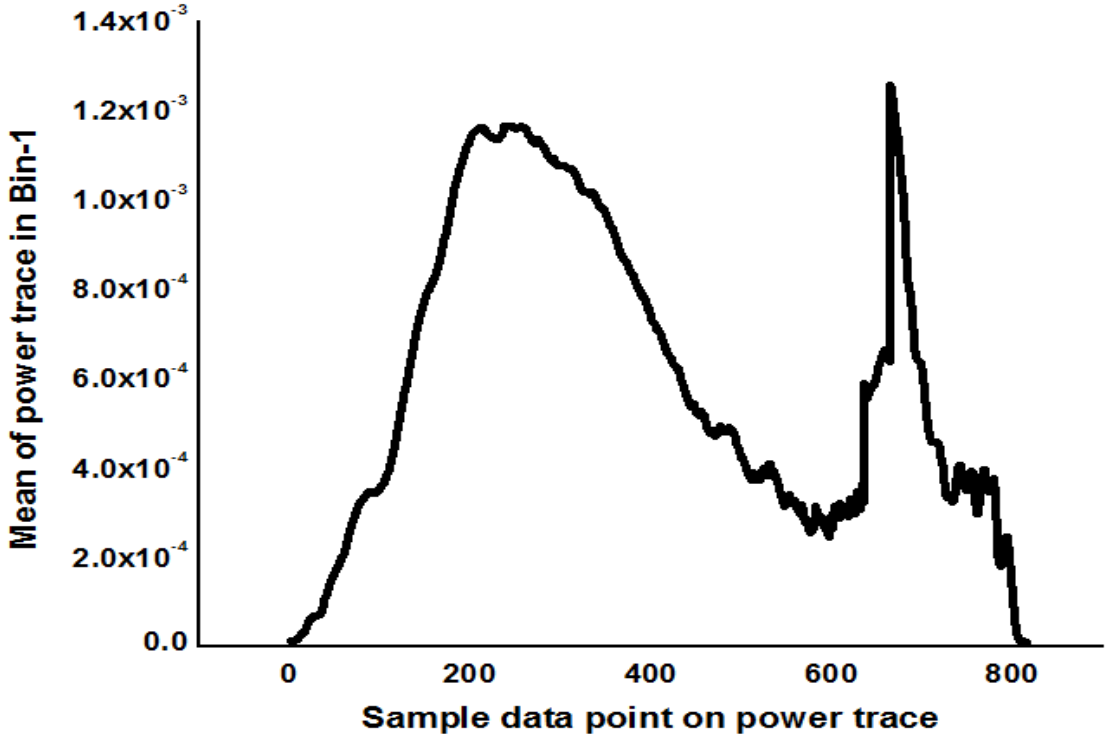

Figure 5.11 Average of power trace in bin 1

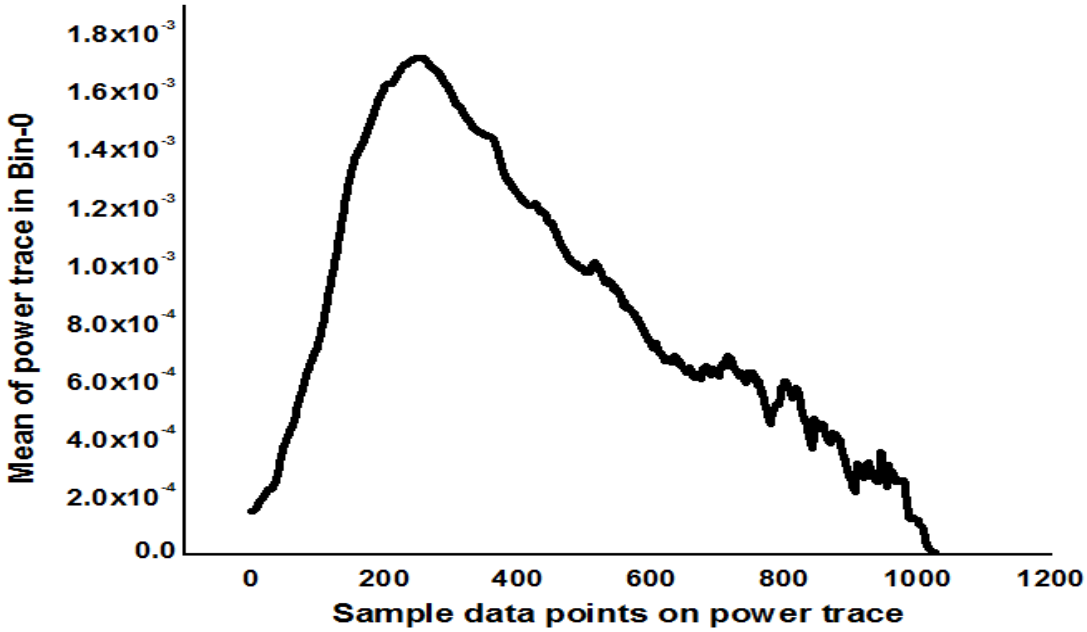

Figure 5.12 Average of power trace in bin 0

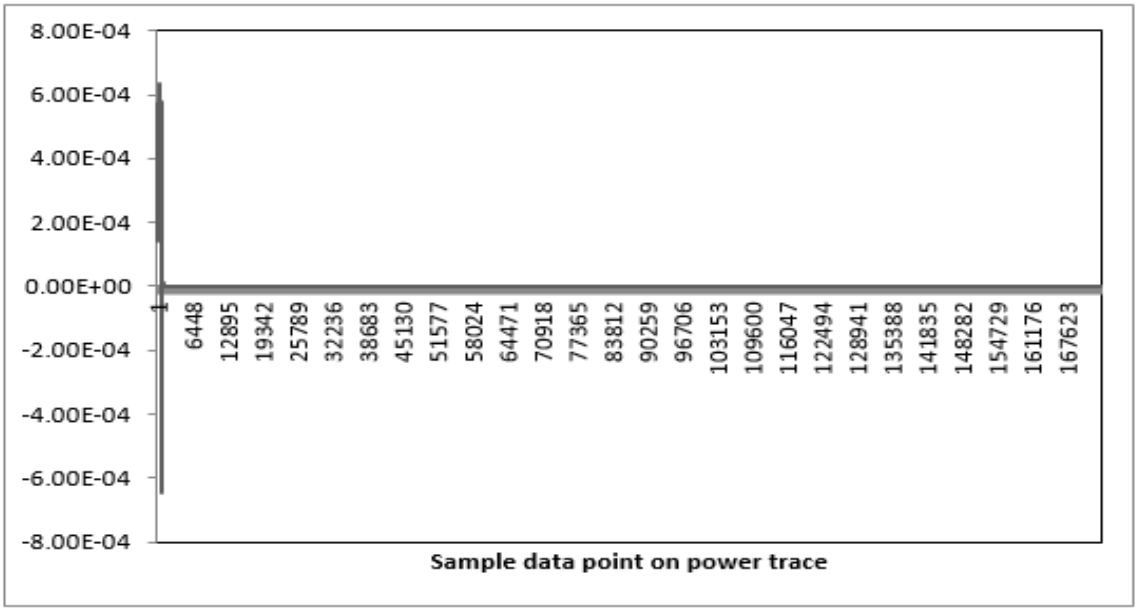

Figure 5.13 Differential curve of average power trace for LSB1 and LSB0

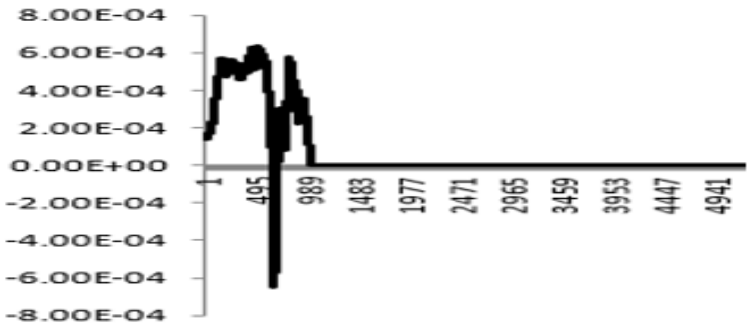

Figure 5.14 Zoomed view of the differential curve for lower sampled value

The shape of the mean of power trace is similar to each other. Here the key is 25H, and input plain text varies from 00-FFH. Each input applied for 10ns; total transient time is 2560 ns. Power traces clipped for 10ns difference of average power plotted concerning samples; a total of 1000 samples obtained for each input. Figure5.13 doesn't show a spike; for a higher sample point by scaling the power trace on lower sample points, the highest peak observed at sample point 495 corresponds to the guessed key zoomed view shown in figure 5.14. Considering the simulation setup 0-3ns circuit in an ideal mode not operating, a peak below 3 ns does not attribute to the DPA experiment. Still, a peak may occur at this time due to noise. In between switching of input vector and delay due to the AddRoundkey stage, correlation needs to found between the sample point and guessed key. DPA guessed all possible key and resulting differential key decide highest spike hits for which assumed key.

DPA attack implemented in the first round, the following limitation of DPA attack identified as [142]

a. The contribution of non-target bit creates curve 0 is independent of the target bit, which creates curve 1.

b. The predicted value of the target bit for the incorrect key doesn't reflect the correct key guess.

c. A peak in the DOM waveform does exist known as ghost peak as a result of noise. Sometimes correct peak may be smaller than ghost peak, leading to confusion for the attacker to make an accurate prediction.

5.5 Experiment with CPA

CPA attack on SBOX depends on how many traces required to predict key used in AddRoundkey input. To perform the CPA attack adversary tries to find a correlation between simulated power trace and hypothesis power model developed by hamming weight and hamming distance [143]. Guess a key randomly 25H, and apply plain text 00-FFH, collect the data-dependent dynamic power consumption value. To develop a hypothesis power model from the equation concerning the input of SBOX initially assume the key is 0, find the power value of each plain text 00-FFH. Repeat the above process by assuming the key 256 times. The last step is to find the Pearson correlation coefficient between simulated power value and 256 hypothesis power value. Figures 5.15 and 5.16 present plots the graph of the Pearson correlation coefficient between actual power value and predicted power value obtained through hamming weight and hamming the distance power model, respectively. The curve flat for the incorrect key while peak occurs for the correct key. With hypothesized key 25H, the correlation coefficient peak obtained with HWPM and HDPM is 0.3245 and 0.39644. The correlation coefficient attends a peak value when guessed key in the hypothesis model matches a simulation model. A higher amount of HDPM during the CPA implies attack key can obtain with more accuracy than HWPM.

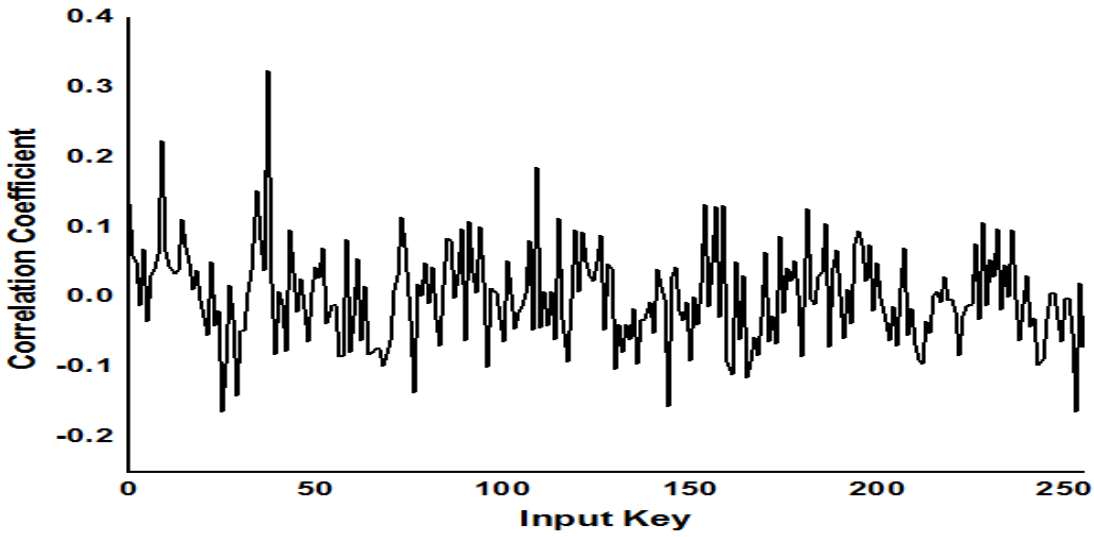

Figure 5.15 Correlation coefficient vs guess key of mask SBOX with HWPM to guessed correct key 25H

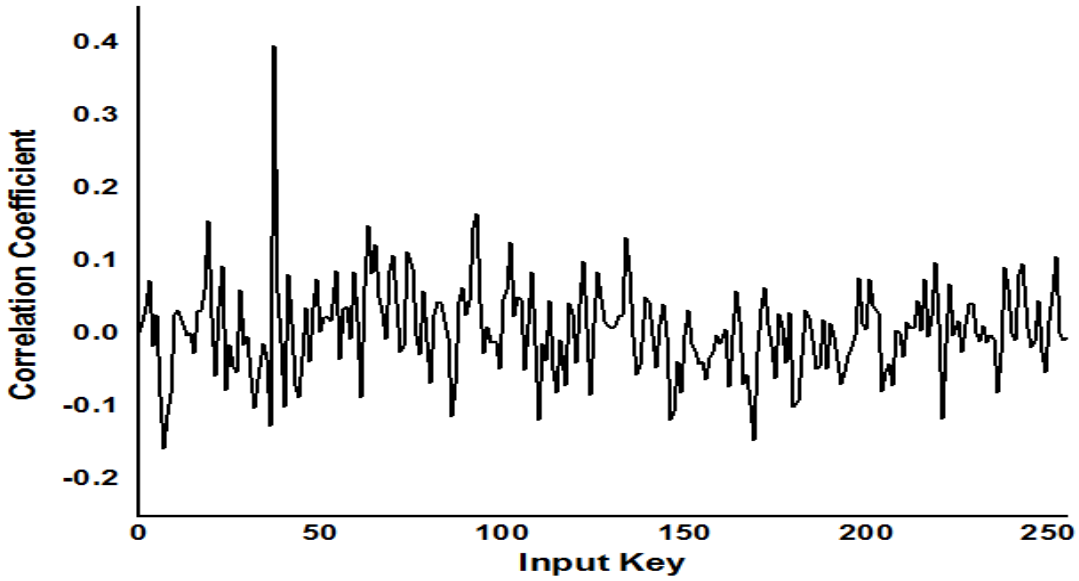

Figure 5.16 Correlation coefficient vs guess key of SBOX with HDPM to guess correct key 25H

Another experiment with CPA shows if incorrect key guessed no peak occurs on the correlation curve, Power trace obtained with secret key 25H and hypothesized key is 12H for HWPM and HDPM model. Collect the correlation coefficient value for all 256 combinations of plain text 00-FF, figure 5.17, and figure 5.18 present the correlation coefficient between actual power trace and HWPM and HDPM power value with wrong key 12H respectively. The correlation coefficient curve resembles a ghost peak attacker not able to decide the correct key. The attacker predicted the wrong key and repeated the experiment with another guessed key until the peak occurs in the curve. Thus, the CPA attack can guess the hidden key correctly. It put light to have CPA attack resistant SBOX, which will not reveal the key under power attack.

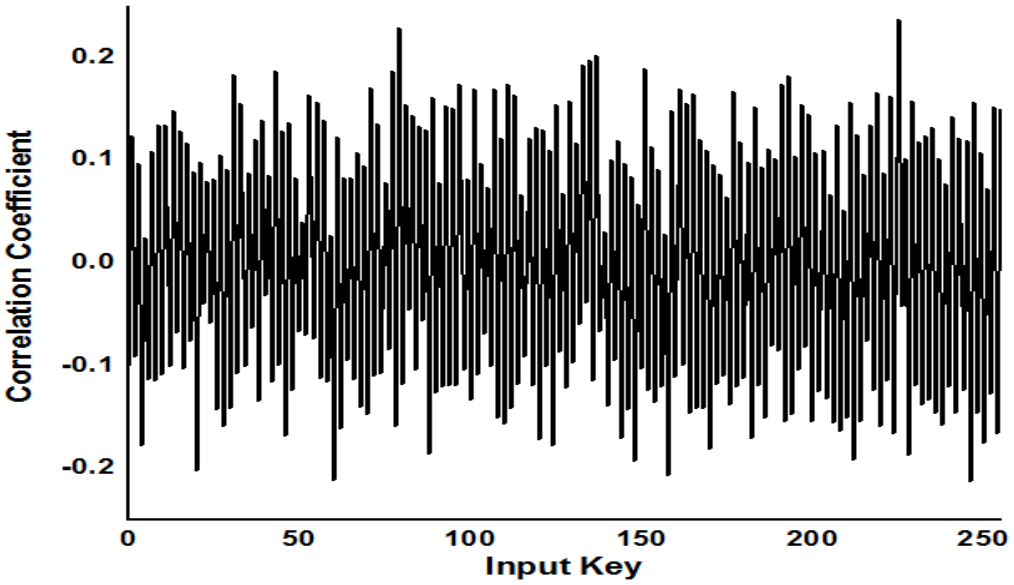

Figure 5.17 Correlation coefficient vs guess key of SBOX with HWPM to guessed incorrect key

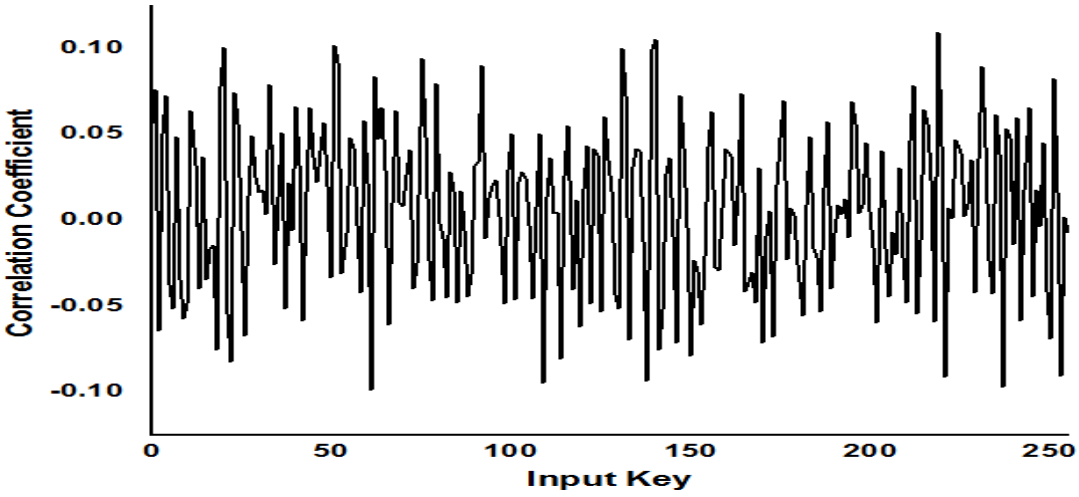

Figure 5.18 Correlation coefficient vs guess key of mask SBOX with HDPM to guessed incorrect key

Chapter 6

Power Attack Countermeasure

6.1 Introduction

Scaling offers an advantage in area reduction, but continuous scaling enhances power requirement [121, 144, 145]. The data-dependent part of power consumption leads to a novel class of security attacks. It is not enough to include security into the IC at the algorithmic level. Protection aligns with at various design levels during design and fabrication known as attack resistant IC. CMOS based cell library is the default standard for low power VLSI circuit design [146]. Existing two types of countermeasure techniques explained in [75] have been practiced (a) hiding [147]and (b) masking at the cell level. The criteria of hiding countermeasures are making the power requirement of the cryptographic module depend on both the intermediate values and then an operation that is to perform. Hiding technique sows balance or equal power for all hypothesis key, the correlation coefficient of close to zero, and not conclusive to make a correct decision. Hiding technique has implemented in sense-amplifier based logic (SABL), wave dynamic differential logic (WDDL) [148, 149], and three-phase dual-rail precharge logic (TDPL). The hiding technique modifies the circuit power consumption at various levels. The masking technique randomizes the input signals to bypass input-output data dependency.
In the power analysis attack, one uses the dependency between actual power consumption with hamming weighs or hamming distance of the input. The output of the cryptographic circuit relies on the hamming weight of the final output terminal. Power attack resistant mask circuit creates multiple internal nodes; power value depends on the all internal node instead of a single node. The ideology behind masking countermeasure is specific operation involved before cryptographic operations; input to SBOX is masked. The hamming weight of data is to be processed and looks random to outside. The output of SBOX doesn't depend on the original data to be processed (unmask); it depends on mask data. Input to SBOX is mask or scramble with a random bit than perform the substitution

shown in figure 6.1. At the output terminal, the mask bit is again unmasked or descrambled with a random bit generated internally or external to the circuit.

The goal of resistance to a power attack is completely removing or reduces the correlation. Practically it is not possible to remove the data dependency altogether. Adding noise or reducing the power value at an internal terminal correlation between actual and predicted power could reduce. Masking applied on an algorithmic level without modifying the power consumption properties without changing the power consumption characteristic of the cryptographic circuit and masking at the algorithmic level requires rewriting the algorithm, which puts an additional burden. Masking is a method to randomize the internal result that applied at the algorithmic or gate level.

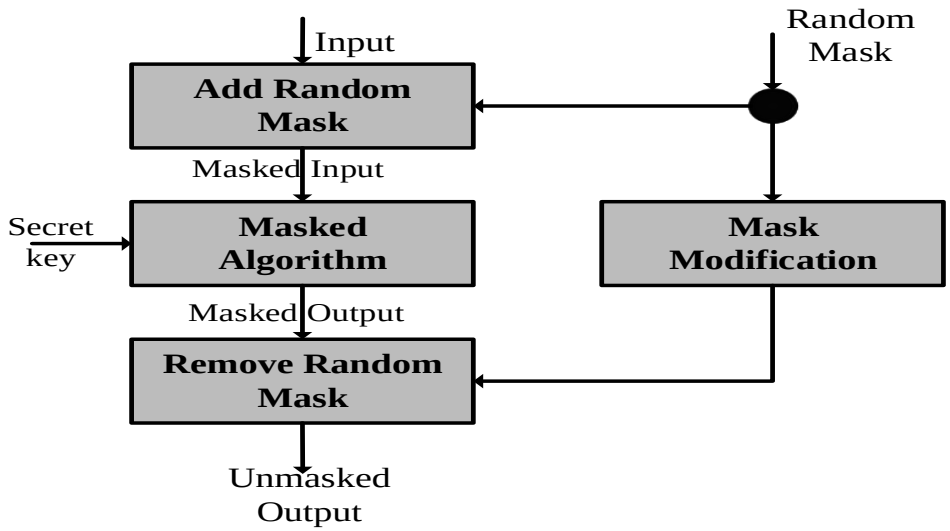

Figure 6.1 Boolean masking process

An alternative method of masking is the use of the mask logic gate to designing cryptographic circuits shown in [150,61]. From the schematic of SBOX, it observed that it requires an ASIC cell of XOR-AND. SBOX design with mask cell ensures the lower correlation between hypothesis and actual power. Mask cells randomize the intermediate value such that wire does not store the value correlated with the algorithmic amount. Security analysis of the mask cell is another aspect that validates to use it to attack resistant cells. Security analyzed with the assumption that input of the mask cell is not switched over once per clock cycle, which does not hold. Arrival time of input signal is

not the same, and output switches more than once per clock cycle, the transition of the gate's output occurs before output stable to final value is called glitches [58,151].

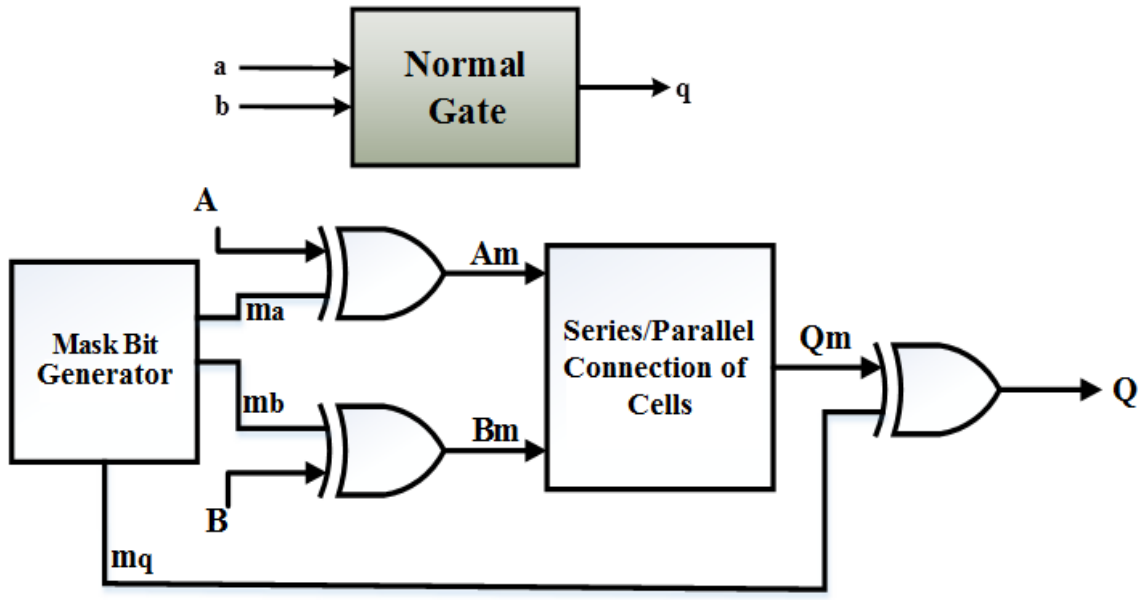

Figure 6.2 Input-output of normal(unmask) and mask gate

6.2 Mask cell

The architectural description of the normal and masks gate explained in figure 6.2 [50, 57, 152], where each gate of the normal gate modified by a random mask bit generated by a mask generator circuit. Mask generator generates a selective mask bit, which is XORed with the input signal. Similarly, the mask gate's output is unmasked with mask bit generated externally or internally to circuits. A normal gate output, q is a function of input a and b presented in [153] q=f(a,b). In mask gate input, a masked to am with mask bit ma and input b is masked to bm with mask bit mb similarly mask output qm is unmasked to q with mask bit mq. A mask gate has five input signals $q_m = f(a_m, m_a, b_m, m_b, m_q)$ [54, 59].

$$\begin{aligned} a_m &= (a \oplus m_a) \\ b_m &= (b \oplus m_b) \\ q_m &= (q \oplus m_q) \end{aligned} \tag{6.1}$$

In this work, mask cells analyzed with the assumption that each value does not change its worth more than one per clock cycle, and propagation delay of each gate is zero, i.e., glitches not considered into design. No energy requires while voltage maintains on the same level. In the digital circuit, the logic level is presented by voltage level at primary input/output or intermediate terminal. The previous section mentioned that power consumption of digital circuit data-dependent, Significant amount of energy is required to switch logic a level. In contrast, zero energy is drawn while the level remains the same discussed in [154]. Energy requires to switch the value from logic 0 to logic 1 is $E_{0\text{-}1}$ presents transition energy needed to perform (0-1) switching. Similarly, $E_{1\text{-}0}$ is transition energy to switch 1-0. $E_{0\text{-}1}$ calculates energy drawn from supply to charge the load capacitor, and $E_{1\text{-}0}$ is energy release while discharges the capacitor. Since $E_{0\text{-}1} \neq E_{1\text{-}0}$. Maintaining the same logic level does not require energy, thus $E_{0\text{-}0}=E_{1\text{-}1} \equiv 0$.

6.2.1 Security Analysis of Unmask cell

Here we have analyzed the power attack counter countermeasure of unmasking XOR and AND cell with the assumption that input to the gate is statistically independent uniformly distributed and arrive at the terminal at the same time. A 2-input gate has 16 possible transitions listed in Table 6.1. XOR and AND Gate Output for all input and transition energy needed to perform the transition. The 16 combinations split into two groups the first group contains transition leads to q=0, and the second group provides transition of q=1. The first group includes the transition combination of 0-0 and 1-0, while the second group includes 0-1 and 1-1 transitions.

In XOR gate 8 times q =1 and 8-time q=0.average energy require for q=0 $E_{(y=0)}$ and average energy requirement for q=0 $E_{(y=1)}$

$$E_{(y=0)} = \frac{4E_{(0-0)} + 4E_{(1-1)}}{8} \quad and \quad E_{(y=1)} = \frac{4E_{(0-1)} + 4E_{(1-0)}}{8}$$

Similarly, for AND gate q=1, 4 times and q=0, 12 times.

$$E_{(y=0)} = \frac{9E_{(0-0)} + 3E_{(1-1)}}{12} \quad and \quad E_{(y=1)} = \frac{3E_{(0-1)} + E_{(1-0)}}{4}$$

Average energy requires to compute XOR output high output $E_{(y=1)}$ is not equal to compute low output $E_{(y=0)}$ presented in equation (6.2).

For XOR gate

$$E_{(y=0)} = 4E_{(0-0)} + 4E_{(0-1)} = 13.18\,fJ$$
$$E_{(y=1)} = 4E_{(0-1)} + 4E_{(1-1)} = 11.79\,fJ$$
$$Thus \;\; E_{(y=0)} \neq E_{(y=1)} \qquad (6.2)$$
$$Average\, E_{(y=0)} = 1.6475\,fJ$$
$$Average\, E_{(y=1)} = 1.47\,fJ$$

Table 6.1Transition energy for unmask XOR-AND cell

Input Transition		AND			XOR		
A	B	Y	Energy (fW)	Energy to settle output Node Y	Y	Energy (fW)	Energy to settle output node Y
0-0	0-0	0-0	0.07	E_{0-0}	0-0	0.06	E_{0-0}
	0-1	0-0	0.08	E_{0-0}	0-1	1.93	E_{0-1}
	1-0	0-0	0.017	E_{0-0}	1-0	1.92	E_{1-0}
	1-1	0-0	0.004	E_{0-0}	1-1	0.07	E_{1-1}
0-1	0-0	0-0	0.09	E_{0-0}	0-1	1.5	E_{0-1}
	0-1	0-1	2.1	E_{0-1}	0-0	2.04	E_{0-0}
	1-0	0-0	2	E_{0-0}	1-1	2.52	E_{1-1}
	1-1	0-1	2.01	E_{0-1}	1-0	1.3	E_{1-0}
1-0	0-0	0-0	0.01	E_{0-0}	1-0	1.78	E_{1-0}
	0-1	0-0	0.11	E_{0-0}	1-1	2	E_{1-1}
	1-0	1-0	2.17	E_{1-0}	0-0	1.9	E_{0-0}
	1-1	1-0	1.92	E_{1-0}	0-1	1.8	E_{0-1}
1-1	0-0	0-0	0.01	E_{0-0}	1-1	0.13	E_{1-1}
	0-1	0-1	2.1	E_{0-1}	1-0	2.5	E_{1-0}
	1-0	1-0	2.07	E_{1-0}	0-1	2.1	E_{0-1}
	1-1	1-1	0.01	E_{1-1}	0-0	0.1	E_{0-0}

Average energy requires to compute AND output high output $E_{(y=1)}$ is not equal to compute low output $E_{(y=0)}$ presented in equation (6.3).

For AND gate

$$
\begin{aligned}
&E_{(y=0)} = 9E_{(0-0)} + 3E_{(1-0)} = 8.58\,fJ \\
&E_{(y=1)} = E_{(0-1)} + E_{(1-1)} = 6.22\,fJ \\
&Thus\, E_{(y=0)} \neq E_{(y=1)} \\
&Average\, E_{(y=0)} = 0.715\,fJ \\
&Average\, E_{(y=1)} = 1.555\,fJ
\end{aligned}
\tag{6.3}
$$

6.2.2 Power Consumption Pattern of Mask Cell

Average $E_{(y=0)} \neq$ Average $E_{(y=1)}$ indicates the attacker to measure leakage along with side-channel information. If Average $E_{(y=0)}$ – Average $E_{(y=1)}$ = 0 the cell is considered as resistant to attack. Objectives of a mask cell achieved if Average $E_{(y=0)}$ – Average $E_{(y=1)}$ = 0 for the gate used in the circuit [55]. It can achieve by having the same amount of energy required for all transition, i.e., $E_{00}=E_{01}=E_{10}=E_{11}$. Usually, first-order masking can extend to multiple order bit; it increases the requirement of multiple bits and enhances the rear and power requirement. In first-order masking, only one bit used to secure. Input a and b split into two values a_0= (a $\oplus$ m_0), a_1=a, b_0= (b $\oplus$ m_1), b_1=b with independent mask bit m0 and m1. Mask bit can be the same or different. It is not necessarily a particular mask gate that works well for all four combinations of m_0m_1, so the mask generator needs to generate a particular mask bit for the selected mask gate. Input is XORed with mask bit because of truth-table of XOR gate in uniform. Since the mask gate adds up the XOR gate at the input and output terminal, it enhances the gate's number at the architecture level. We have proposed two novel architecture of the XOR gate and one AND gate. The truth table of the mask gate for four possible combinations verify hamming weight are intermediate terminal for the output terminal is equal.

6.2.2.1 Proposed Mask XOR-1 cell

The proposed architecture of mask XOR cell implemented with 4-XOR and 1-AND gate. Intermediate signal presented by T0, T1, T2, and T3 AND gate at output computed unmask output.

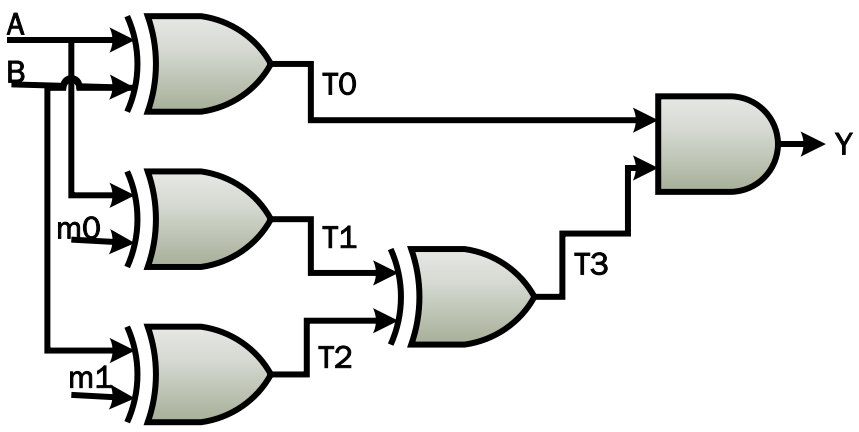

Figure 6.3 Proposed mask XOR-1 cell

$$Boolean\ \exp ression\ of\ masked\ XOR-1$$

$$T0 = A \oplus m0$$

$$T1 = B \oplus m1$$

$$T2 = A \oplus m1$$

$$T3 = T1 \oplus T2$$

$$Y = T0 \bullet T3$$

Table 6.2 Truth Table of Mask XOR-1 with Mask bit

Input		Internal node and output with mask bit 00					Internal node and output with mask bit 11				
A	B	T0	T1	T2	T3	Y	T0	T1	T2	T3	Y
0	0	0	0	0	0	0	0	1	1	0	0
0	1	1	0	1	1	1	1	1	0	1	1
1	0	1	1	0	1	1	1	0	1	1	1
1	1	0	1	1	0	0	0	0	0	0	0
HW		2	2	2	2	2	2	2	2	2	2

6.2.2.2 Proposed Mask XOR-2 Cell

Another architecture of mask XOR cell requires 3 XOR gate to input mask input, 2 XOR compute intermediate signals (T0-T4) and unmask output Y computed with XNOR gate.

$$Boolean \exp ression\ of\ mask\ XOR-2$$

$$T0 = A \oplus m0$$

$$T1 = m0 \oplus m1$$

$$T2 = B \oplus m1$$

$$T3 = T0 \oplus T1$$

$$T4 = T1 \oplus T2$$

$$Y = \sim (T3 \oplus T4)$$

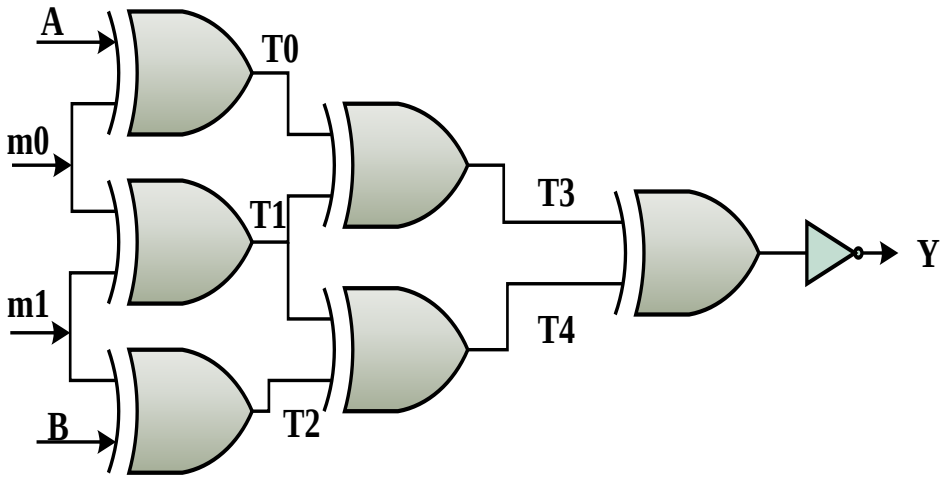

Figure 6.4 Proposed mask XOR-2 cell

Table 6.3 Truth Table of Mask XOR-2 with Mask bit

Input		Internal node and output with mask bit 01					Internal node and output with mask bit 10					
A	B	T0	T1	T2	T3	Y	T0	T1	T2	T3	T4	Y
0	0	0	0	0	0	0	1	1	0	1	1	0
0	1	1	0	1	1	1	1	1	1	0	0	1
1	0	1	1	0	1	1	0	1	0	1	1	1
1	1	0	1	1	0	0	0	1	1	0	0	0
HW		2	2	2	2	2	2	4	2	2	2	2

6.2.2.3 Proposed Mask AND Cell

Proposed mask AND cell require a 5-XOR and 3-AND gate. Mask bit m0m1 mask the current input A and B; the intermediate stage signal T0-T6 distributes the power in the cell's internal terminal. The output terminal at Y computes the output.

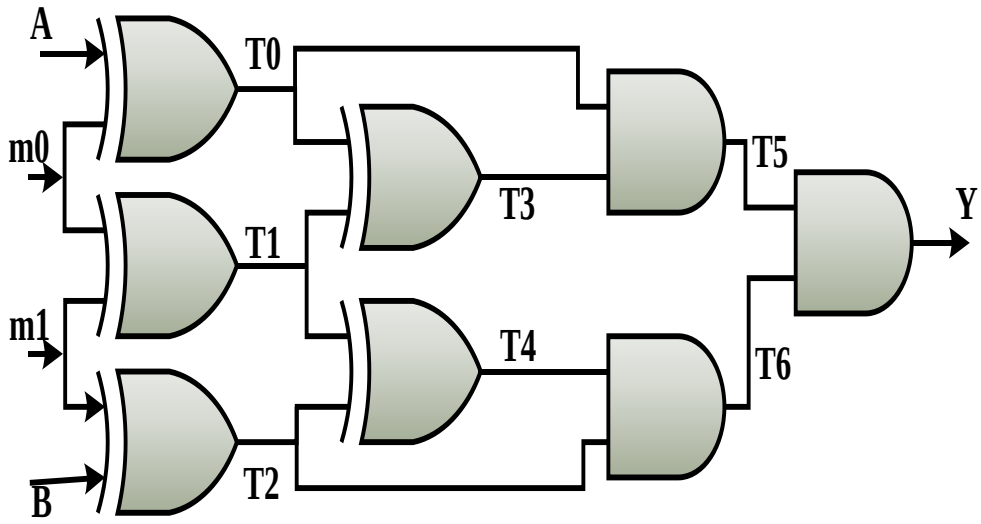

Figure 6.5 Proposed mask AND cell

Table 6.4Truth Table of Mask AND with Mask bit

Input		Internal node and output with mask bit m0m1= 00							
A	B	T0	T1	T2	T3	T4	T5	T6	Y
0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	1	0	1	0
1	0	1	0	0	1	0	1	0	0
1	1	1	0	1	1	1	1	1	1
HW		2	0	2	2	2	2	2	2

Boolean expression of mask AND cell

$$T0 = A \oplus m0$$
$$T1 = m0 \oplus m1$$
$$T2 = B \oplus m1$$
$$T3 = T0 \oplus T1$$
$$T4 = T1 \oplus T2$$
$$T5 = T0 \bullet T3$$
$$T6 = T2 \bullet T4$$
$$Y = T5 \bullet T6$$

Each input of the proposed gate is masked with an additional XOR gate. Boolean expression of mask XOR-1, mask XOR-2, and mask AND presented in figure 6.3, figure

6.4, and figure 6.5, respectively, at the intermediate terminal are given in the Boolean equation. Mask XOR-1 implements the XOR functionality for mask bit 00 and 11, while for 01 and 10, it implements the XNOR gate functionality. Mask XOR-2 works as XOR gate with mask bit 01 and 10, and masking of AND gate is achieved with mask bit 00. The output of the mask gate unmasked with an internally generated bit, mask XOR-1 unmask with AND gate, mask XOR-2 unmask with XNOR gate and mask, AND gate unmasked with AND gate respectively. The truth table of the mask gate at the internal terminal uniformly distributed "2" shown in table 6.2, table 6.3, and table 6.4 for mask XOR-1, mask XOR-2, and mask-AND cell, respectively. The equal value of the hamming weight present wire at the internal terminal does not store value; thus, the energy required to set the output node is distributed on an internal terminal and statically independent; to break the security of the mask gate attacked need to know the value to each intermediate terminal. At the same time, the unmask gate can break with the only output terminal.

6.2.3 Security Measures of Mask Cell

Mask gate enhances security level at the cost of increased gate count and power consumption. The requirement of attack resistant gate is intermediate result must independent of primary input and uniformly distributed. Uniform distribution of hamming weight makes output independent of primary input; an attacker would not be able to predict the sensitive information shown in [155] of the circuit by reverse engineering. Compare to masked XOR, masked AND gate hamming weight follows the uneven distribution. Those are sensitive attack points to reveals information. Hamming weight says the amount of energy requires switching from level. In regular gate low of high transition consumes more energy compared to high to low transition. The masked gate difference of average energy should be as low as possible. The difference of mean energy for masked XOR-1 to set output terminal high or low calculated as mentioned in section1; E(y=1) – E(y=0) = 0:15fJ, for mask XOR-2, 1.5625fJ and for masked AND cell 5.6fJ. Masked XOR-1 achieves an 82.9% reduction in switching energy compares to the

static unmask circuit. Thus, it deduces the proposed masked XOR gates are strong candidates for attack countermeasures. Normalized energy deviation (NED) and normalized standard deviation (NSD are two figures of merit to evaluate resistance to power attack shown in equation (10). NED presents the percentage difference between the minimum and maximum energy difference for all possible transitions. The smaller value of NED present low variation in energy and attacker requires more complex measurements.

NSD calculates as (σE / EAvg) where σE is a standard variation of energy. NSD measures how consumed energy distributed around the mean; the most significant value of NSD indicated energy widely spread across mean smaller value of NSD indicates they are close to the mean. NED-NSD analysis carried at frequency 1GHz, a lower value of NED and NSD shows a secure system, attacker requires more sophisticated measurement and resistance to power attack discussed in chapter4. Table 6.5 presents a comparative performance analysis of masked XOR-AND gate compare to static unmask CMOS architecture with their mask bit. It observed that the proposed masked XOR-1 for mask bit 00 and 11 shows 2.3% and 1.5% improvement in NED. Mask XOR-2 for mask bit 01 and 10 shows 1.7% and 1.89% improvement in NED. At the same time, NSD shows 6.95% and 3.29% improvement for masked XOR gate for mask bit 00 and mask bit 11.

Similarly proposed masked AND gate for mask bit 00 gate shows 0.03% and 0.05% improvement in NED and 25.75% and 29% reduction in NSD respectively at input frequency 1GHz. Masked XOR-AND gate does not show a much reduction in NED; this is because of the non-uniform distribution of hamming weight at the internal node. Compared to masked XOR masked AND gate shows more reduction in NSD because AND gate shows non-uniform hamming weight at the intermediate terminal. Proposed masked XOR-2 and AND cell achieves 20%, and 18.94% reduction in NSD compare to that in static unmask CMOS circuit at frequency 1GHz, verify complex measurement require for cryptanalysis. A similar complexity level requires recovering hidden secrets from the internal node of the circuit. Figure 6.6 presents the % reduction of Pearson correlation coefficient in masked XOR-AND at their mask bit gate compare to the static

unmask architecture of that one. By observing the result summarized in table 6.5, we concluded that there is a significant reduction in person coefficient in the masked gate, compared to a static unmask cell. Pearson coefficient of normal XOR is 0.134, whereas 0.0053 and 0.0239 for mask XOR-1 and, for mask XOR-2 0.0369 and 0.0416 for respective mask bit. Pearson coefficient of mask AND cell is 0.3 compared to normal cell 0.372. A lower value of mask cells exhibits power pattern independence with input data. Thus, cryptographic circuit implementation with masked gate improves the algorithm's security level at the increased hardware cost. A masking method of AND cell in [156] requires four AND & XOR cell the proposed mask AND cell need two XOR and one AND cell.

Support vector machine (SVM) supervised learning model based on the classification algorithm. Parameter of SVM to predict the performance of SBOX with the mask the bit is cost and epsilon. Cost measures how much SVM allows to ben with data. A lower value of cost exhibits larger, margin, and smoother decisions, while higher cost value exhibits a smaller margin, classifies more points. Epsilon denotes a margin to a tolerance no penalty to error, error into model presented with the higher value of epsilon. Implemented SBOX achieves C=8 and epsilon = 0.78 while mask C=4 and epsilon=0.97.

Table 6.5 Energy parameter and correlation of Mask gate

	Energy Parameter						Pearson Coefficient (ρ)
	E_{Min}	E_{Max}	NED	E_{Avg}	σ_E	NSD	
Static unmask XOR	0.06	2.52	0.9762	0.1538	1.5748	0.6402	0.134
Mask XOR-1 (00)	0.01	7.9	0.9987	0.4931	5.4026	0.6847	0.0053
Mask XOR-1 (11)	0.1	12.1	0.9917	0.75	7.9841	0.6653	0.0239
Mask XOR-2 (01)	0.1	15.4	0.9935	0.9563	7.8316	0.5119	0.0369
Mask XOR-2 (10)	0.09	16.9	0.9947	1.0506	8.7227	0.5189	0.0416
Static unmask AND	0.004	2.17	0.9982	0.1354	1.2715	0.587	0.372
Mask AND (00)	0.06	40	0.9985	2.4963	17.401	0.4357	0.3

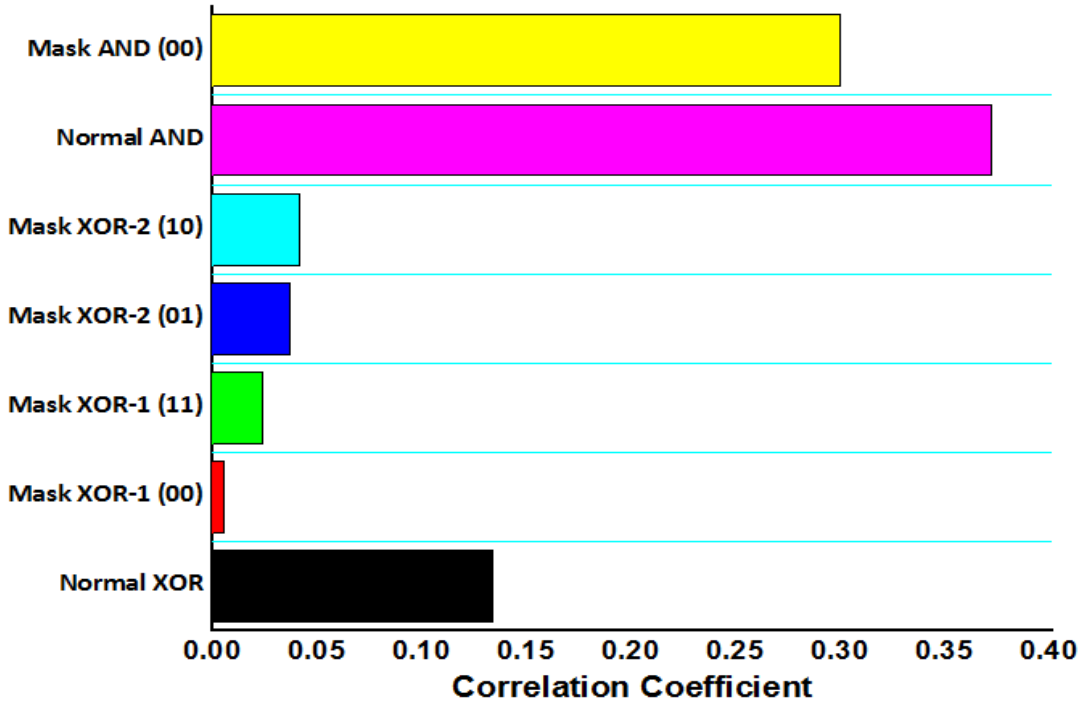

Figure 6.6 Pearson correlation coefficient of normal and mask cell

6.2.4 Mask cell with bias bit

Masking is known for secret sharing, a well-known approach to countermeasure the power attack. It splits the sensitive information into multiple shares into (d+1) share where d presents masking order. In present work, only 2-random bits used to mask the underlying cell of SBOX. The first-order mask splits into two parts with the help of the XOR gate. To mask a nonlinear block like SBOX, it necessary to apply an absolute random value for mask bit. For the perfect mask, have a probability value of 0.5, i.e., probability of occurrence '0' and '1' equal p0=p1. Sensitive information combined with a random mask by XOR gate, masked values sm = (s⊕m) becomes statically independent of primary input. The mutual information between s and sm is zero. While imperfect masking probability of occurrence takes value is (0.5 + ε) where ε≠0. Probability of occurrence of one value increase by ε and other value decreases by ε. When sensitive information is randomized with bias mask bit leaks, additional information used to

recoup the variable even with countermeasure [165]. The power distribution is no longer uniformly distributed. Information leakage analysis (ILA) and power trace and probability follow and likelihood are correspondence to one another as presented in equation (6.4).

If the bitmask is biased, information leakage quantity varies with biasness presented as the probability of occurrence for a bit. Figure 6.7 shows the relationship between ILA and p; the curve is symmetrical about the x-axis, which implies that the mask bit is 0 and 1. When p = 0 or p = 1 indicates, the circuit is without masking protection, leaks maximum information. For p = ½ the circuit is protected by perfect masking, ILA approaches to zero, which implies a large number of power traces are required to collect sufficient information. (Since the number of power trace needed for power attack is inversely proportional to attack (ILA), In case of imperfect or biased masking (p≠0), information leakage quantity increases as p approaches to axis terminal from the centre. The number of powers trace requirements decreases to implement the attack.

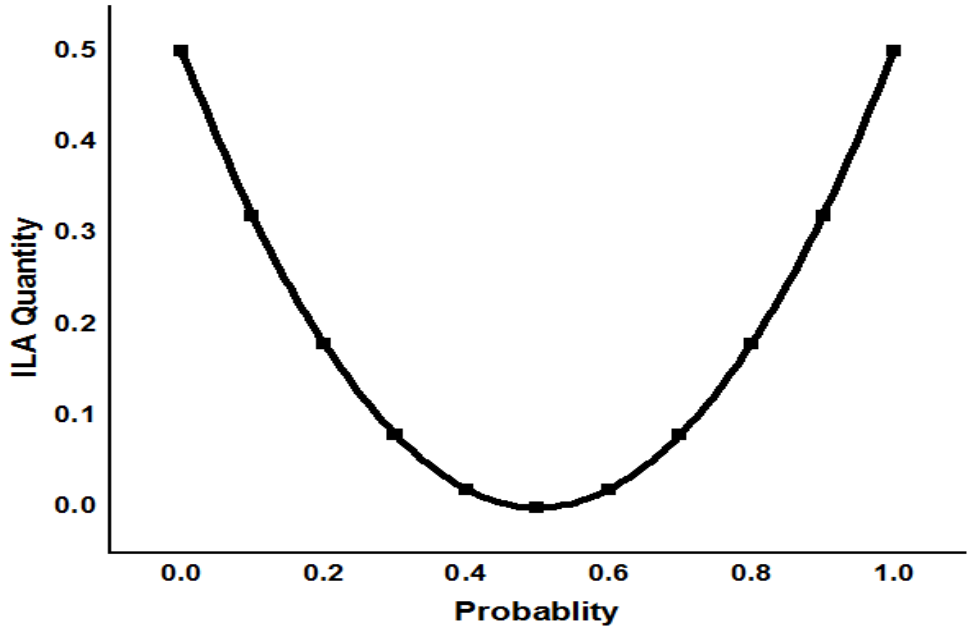

Figure 6.7 ILA vs probability of occurrence of mask bit

6.3 CPA Attack model on SBOX implemented with mask cell

In this, we have implemented SBOX with proposed mask XOR-1 and mask AND gate. The computing structure of SBOX presented in the chapter where simple XOR-AND replaced by mask XOR-1 mask AND. To analyzed the correlation power attack, simulated power traces of SBOX are obtained with guessed secret key 12H plain text varies in the range of 00-FFH. The correlation coefficient between hamming weight vs power trace and power trace vs hamming distances presented in figures 6.7 and 6.8. Unlike the SBOX mask, SBOX shows multiple peaks in the correlation coefficient that gives actual power consumption correlated with multiple plain texts. So, there is no exact guess on the correct key. The proposed mask SBOX requires 13260 BSIM3V3 MOS cells, which is 5.7 times greater than SBOX with static cells. Since several cells increases in Mask SBOX leakage current accumulated and result in high leakage power 55.96µW. The dynamic power consumption of mask SBOX is 964.3µW, and the total propagation delay is 1.567ns.

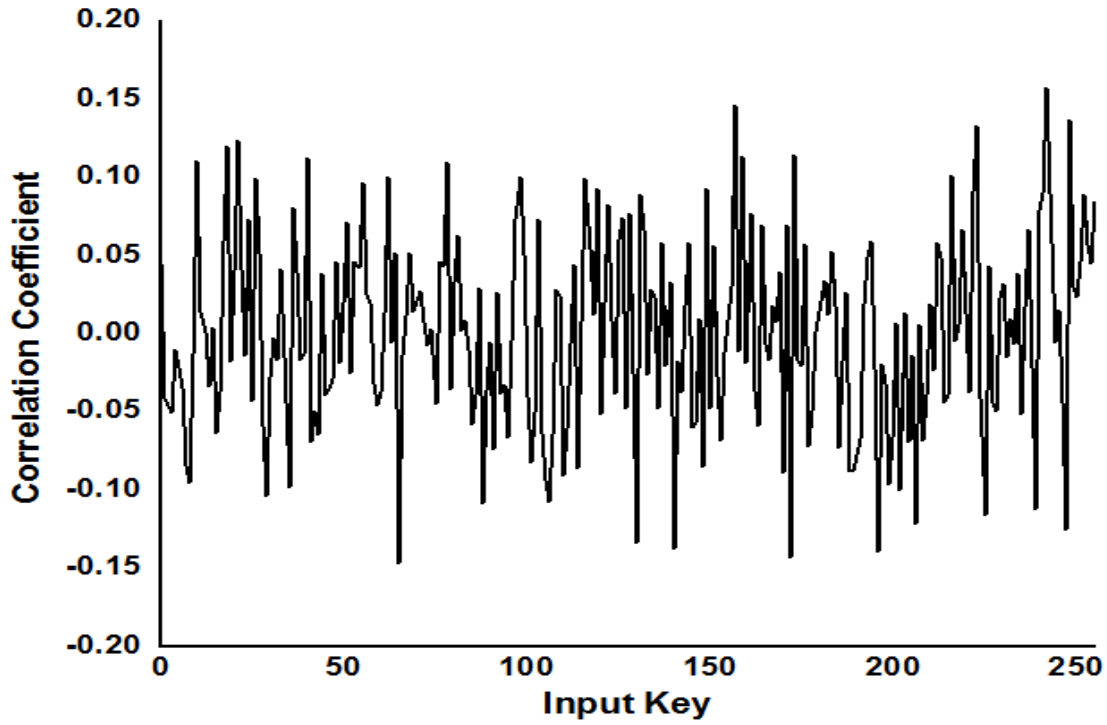

Figure 6.8 Correlation coefficient vs guess key of mask SBOX with HWPM to guessed correct key

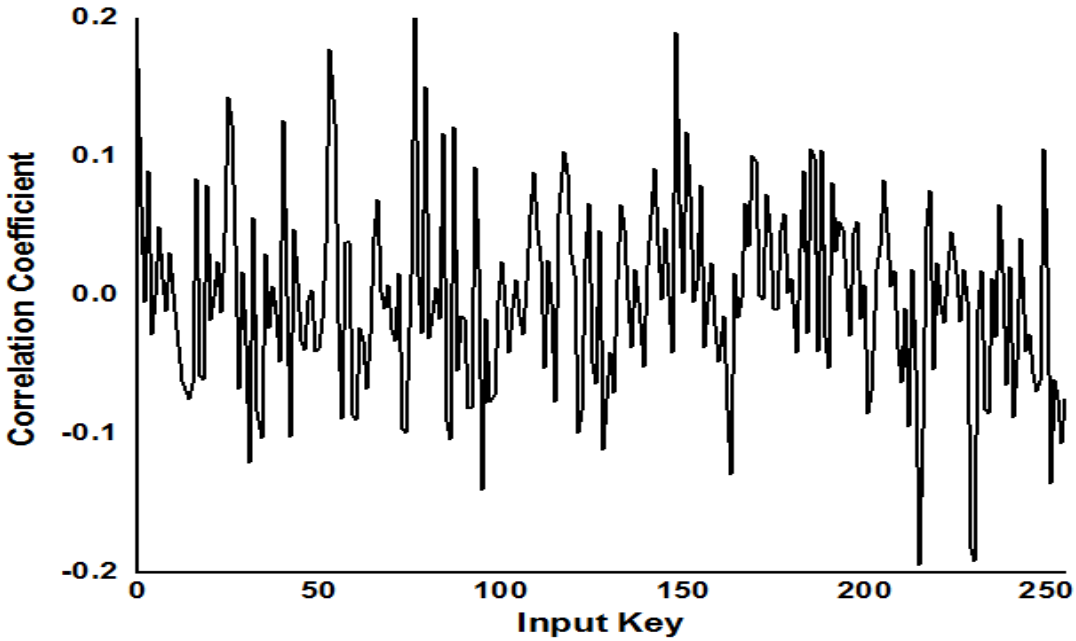

Figure 6.9 Correlation coefficient vs guess key of mask SBOX with HDPM to guessed correct key

Moreover, figure 6.10 and figure 6.11. presents the Pearson correlation coefficient (ρ) with different plaint text between simulated power consumption and hamming weight (HW) and hamming distance (HD) power model compare to unmask cell SBOX, respectively. Mask cell SBOX achieves a lower value of correlation. We can conclude that SBOX implementation with mask cell power consumption has a lower relation with input text than unmasked one. CPA attack with HWPM shown 57.86% and HDPM shows a 49.08% improvement in correlation co-efficient for plain text 12H. Table 6.6 and Table 6.7 presents the correlation coefficient of SBOX implemented with a mask cell with different guessed key for HWPM and HDPM. A significant reduction in correlation coefficient denotes mask cell power consumption is independent of input data to be processed.

Table 6.6 Correlation coefficient (ρ) between the actual power consumption of SBOX and hypothesized power with HWPM for the correct key

Unmasked Key12	Masked Key12	Masked Key 25	Masked Key 47	Masked Key 61	Masked Key 127	Masked Key 197
0.224545	0.094667	0.093314	0.09229	0.092292	0.092297	0.026184

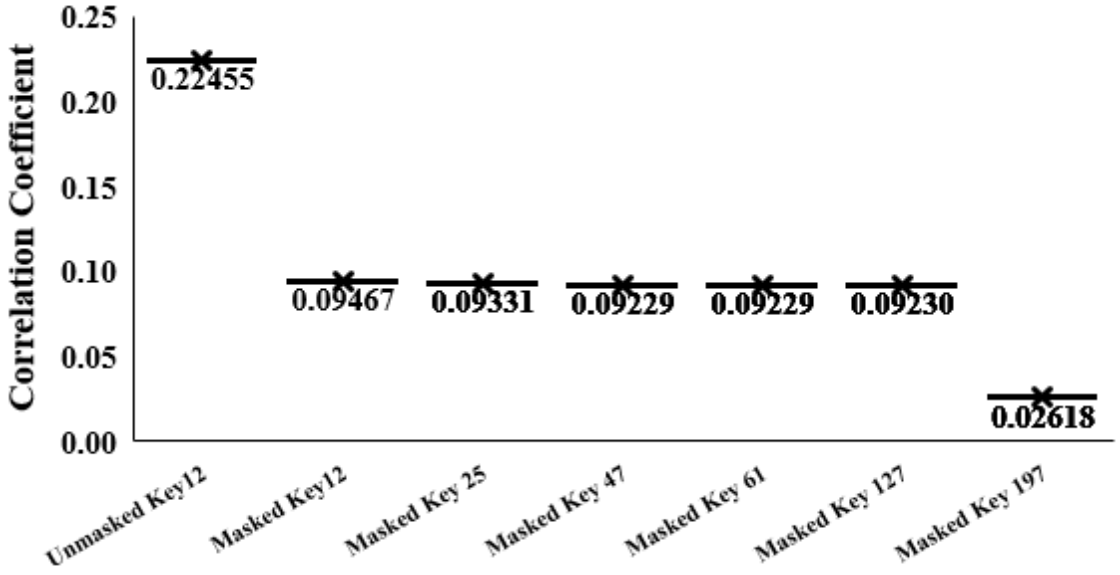

Figure 6.10 Comparison of Correlation Coefficient of SBOX under CPA attack with HWPM for the correct key

Table 6.7 Correlation coefficient(ρ) between the actual power consumption of SBOX and hypothesized power with HDPM for the correct key

Unmasked Key12	Masked Key12	Masked Key 25	Masked Key 47	Masked Key 61	Masked Key 127	Masked Key 197
0.28431	0.144759	0.17166	0.172455	0.172453	0.172449	0.166138

Figure 6.11 Comparison of Correlation Coefficient(ρ) of SBOX under CPA attack with HDPM for the correct key

6.4 Result and Discussion

In this work, we have implemented SBOX with CMOS 90nm technology node since power trace emits unavoidable known side-channel information. This information collected by inserting a low value of resistor a value along with power supply. The input of SBOX generates a unique response and consume different power for each input. Power consumption depends on the number of high bits and the input changes at the input of SBOX. Power consumption relies on the high bit's position at the input, i.e., a high bit on the most significant bit (MSB) consumes more power compared to the least significant bit (LSB).

Three types of power attack models analyzed in the previous section, SPA, DPA, and CPA. DPA and CPA attack based on statistical analyses between actual power and mathematical model to calculate power based on hamming weight (HW) and hamming distance (HD) power model. A power attack based on finding a relation between actual power and mathematically calculated power value. DPA attack is a brute force method search of the key, attacker assumes to the secret key and applies all possible input 00-FFH. The difference of men between power trace for the input of SBOX (LSB=1) and input of SBOX (LSB=0). For correct guess of key peak observe in the DOM waveform else waveform flats to the horizontal axis. DOM curve presents the peak value n trace-point 495. DPA attack shows the peak in the curve for the correct key, and the attacker obtains the guessed key as the correct key. Limitation of DPA attack arises due to noise present in the circuit, sometimes unexpected fluctuations of the curve are observed due to noise and peak found on the wrong key. Compare to DPA, CPA is considered a more efficient methodology that requires a lesser number of the trace to guess the key. Here Pearson correlation coefficient has been used to find a correlation between actual power value and power computed due to hamming weight and hamming distance power model. CPA attack starts with guessing the input text key and apply all possible key 00-FFH. Obtain correlation coefficient power consumption due to guessed key and user input with actual power consumption. Plot the correlation coefficient with respect guessed key. A

sharp peak observed if the guessed key is correct and ghost waveform for the incorrect key. In the CPA attack, it is possible to imagine the secret key with a high success rate.

Enhance the security level existing encryption algorithm; it required that attack countermeasure cells should be used to design the hardware structure. The hardware structure of SBOX implemented with normal XOR & AND gate show power consumption is highly correlated with input data show correlation coefficient 0.2905 and -0.273 for hamming weight and hamming distance, respectively. Masking is an efficient power attack countermeasure, demonstrated a significant reduction in correlation coefficient with the cost of inclusion of additional components in the design. Masking applied at algorithm level mask the input of SBOX, compute nonlinear mask byte, which needs to be unmasked by the new mask. Masking at the cell level reduces the hardware requirement. Here complete SBOX is designed to mask XOR and mask AND gate. Proposed mask XOR and mask AND cell show that power consumption is equally distributed at the internal terminal instead of lumped to a signal. Output model Truth table of mask XOR-AND cell shown hamming weight is two on each terminal. Mask gate requires an additional XOR gate to apply mask bit along with input, mask XOR cell uses AND/XNOR gate to unmask and mask AND uses additional AND gate to reveal the output terminal. Table 6.8 shows the hardware resource and attribute of unmasking and mask SBOX.

Compare to SBOX presented in [156], proposed to unmask SBOX reduce 41 number of cells. Since unmask SBOX is designed at CMOS 90nm to have a high value of static power to 8.729 μW, but the lower value of dynamic power consumption 216 μW compare to 4.187 mW. SBOX with mask cell requires 635 cells in contrast to 2061 cells present in [156]. Delay of mask SBOX is 1.567 ns, but significant increases the static power. Table 6.9 presents the comparison of the proposed SBOX parameter with existing results. The number of gate count required for mask SBOX reduces to 2173 while the [158] gate count is 5478 and 3628 for balance pipeline and multiplicative masking, respectively. Boolean masking employed in [160] requires lower gate count 2051, but the delay is higher. Masking at a particular segment, i.e., only AND cell in [160] and mask

only multiplicative unit in [158], requires higher gate counts and increases the delay proportionally. The unpredictability of SBOX with the masking feature shown in table 6.9.

Table 6.8 comparison of the different topology of SBOX

	Unmasked SBOX	[156]	Masked SBOX	[156]
Technology	CMOS 90nm	180nm CMOS	CMOS 90nm	180nm CMOS
Static Power	8.729 µW	22.973nW	55.96 µW	51.151nW
Dynamic Power	216µW	4.187mW	964.3 µW	16.2mW
Delay	0.005ns	7.02ns	1.567ns	9.44ns

Reduction in Pearson, correlation coefficient value, presents the power trace in loosely correlated with data processed. Which highlights attacker cannot predict sufficient hidden secret during CPA attack. The presented SBOX with Boolean masks at the cell level attains the correlation coefficient of 0.0946 and 0.1447 for HWPM and HDPM, respectively, much lower than [45 161-163]. The lower value of ρ in HWPM presents minimum information leaks compare to HDPM. SBOX with Boolean mask at cell reduces the correlation coefficient 0.0946(HWPM), and 0.1447(HDPM) compare to SBOX with a normal cell, which implies the improvement of unpredictability to 42.13% for HWPM and 50.89% for HDPM shown in figure 6.12.

Table 6.9 Comparison of correlation co-efficient with hamming weight/distance power model

	(ρ) HWPM	(ρ) HDPM
[161]	0.59	
[162]		-0.43
[45]	0.997	
[163]		0.1838
[164]		0.1352
This Work SBOX with unmask cell	0.2245	0.2843
This Work SBOX with mask cell	0.0946	0.1447

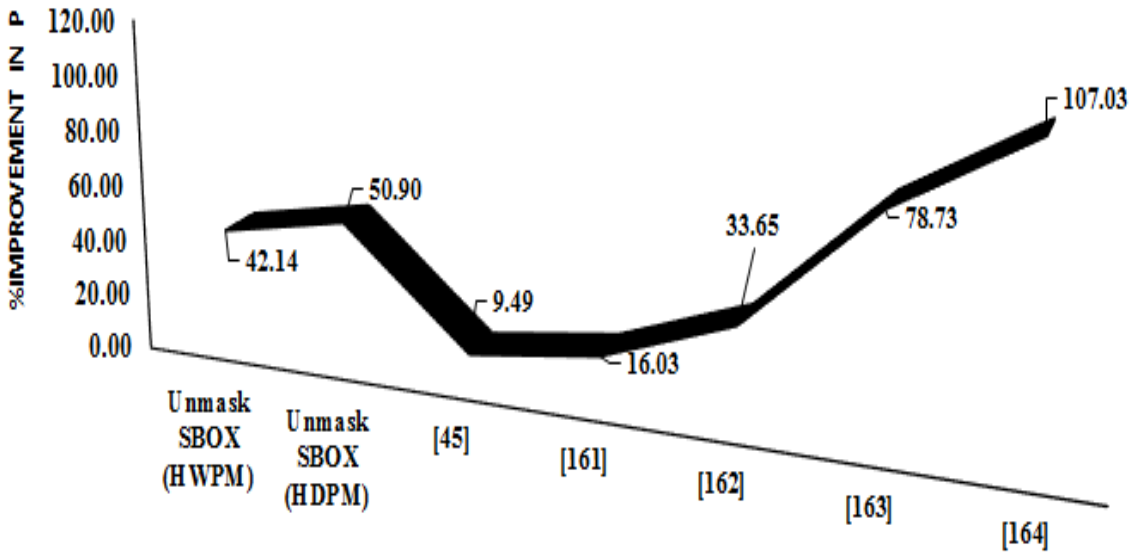

Figure 6.12 Improvement in ρ of SBOX with mask cell

Table 6.10 Comparison of SBOX parameter with making method

References	Area (Gate Counts)	Delay (ns)	Technology	Masking Method
[153]	13000		CMOS 180nm	Gate level masking of AND
[154]	5478	8.33	CMOS 90nm	Balanced Pipelining
[154]	3628	59.13	CMOS 90nm	Multiplicative Inverse
[156]	2061	14.299	CMOS 180nm	Boolean Masking
This Work	2173	1.567	CMOS 90nm	Gate level masking of XOR & AND

Table 6.10 presents a comparison of the performance metric of implemented SBOX with the proposed masked cell. Schematic of SBOX is implemented with masked XOR-AND gate attend the smaller value of delay compared to the actual result in [153,154 and 156]. The number of gate requirements is higher than [156] but lower than [153,154].

Chapter 7
Conclusions and Future Scope

7.1 Conclusion

The investing and comparison of secure 8-bit AES SBOX using CMOS 90nm technology and their countermeasure against power attack analysis carried out in this work. We have implemented the DPA and CPA at the output of the first round of SBOX. The underlying cell of SBOX is implemented with static, TG, and PTL CMOS logic to measure the dependency of power profile with processed data. Another limitation of static CMOS logic leaked information and power consumption patterns is the interest in power attack analysis. Hybrid CMOS logic saves the gate count at the cost of computation delay. NED and NSD measure the complexities of computation power attack, SBOX implemented with static CMOS logic score NED 0.855 and NSD 0.0266, which implies energy requirement maintain linear relation with input data. SBOX implementation with hybrid CMOS logic reduces the value of NED 0.4391, and NSD 0.0208 shows lesser information leaks and power traces.

In this work, we have implemented the power attack resistant substitution box with improvement in the correlation coefficient. The mask countermeasure presents information leaks along with power traces of power traces is low correlated with processed data but enhances the gate count and power measures. The proposed masked XOR and AND cell possess a nonlinear relationship between power consumption pattern and input masked value and mask input bit. Mask XOR-AND cell correlation coefficient is 0.0053 and 0.3, which is much lower than 0.134 and 0.372 for unmasked one. The power consumption pattern of SBOX with a mask cell shows an 8.49% improvement in the correlation coefficient, which measures the unpredictability of the independence of power consumption patterns from processed data. PUF is a preferred hardware-based security module for device authentication and key generation. It is necessary that the response of the PUF highly unique. PUF allows the designer to identify the circuit-

specific feature to mix into the computation. In this work, we have demonstrated the Schmitt trigger as a basic unit to design PUF, whose response is a function of input and two devices feature path delay and hysteresis width. The designed ST_PUF proves to unique 44.71 and 99.71% with operating conditions.

7.2 Future Scope

Any electronic feature with variable nature, a stable to a constant value, can be used to design a PUF. The selection of new device-specific features can turn a new category of PUF circuits. Identification of new electronics property to have a new robust PUF structure. The prime focus of PUF is to simplify the structure and response are unique such that cannot be predicted even in the learning environment. Analysis of CMOS logic to remove the dependency between primary output and secondary information would be another research area to have attack resistant cryptographic module. The attack resistance feature must include during different design phases of IC, so the external countermeasure technique not required to implement.

BIBLIOGRAPHY

[1] C. Helfmeier, C. Boit, D. Nedospasov, and J.-P. Seifert, "Cloning physically unclonable functions," IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Austin, TX, USA, pp. 1-6, IEEE, 2013.

[2] T. Xu and M. Potkonjak, "The digital bidirectional function as a hardware security primitive: Architecture and applications," IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED), Rome, Italy, pp. 335-340, IEEE, 2015.

[3] Z. Cherif, J.-L. Danger, F. Lozac'h, Y. Mathieu, and L. Bossuet, "Evaluation of Delay PUFs on CMOS 65 nm Technology: ASIC vs FPGA," proceeding of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy, Tel-Aviv, Israel, pp. 1-8, 2013.

[4] J. P. Garcia, "Electronic logic circuit with physically unclonable function characteristics," U.S. Patent No. 8,274,306. Washington, DC: U.S. Patent and Trademark Office, September 25, 2012.

[5] R. Maes, Physically unclonable functions: Constructions, properties, and applications, 1st ed. Netherlands: Springer Science & Business Media, pp. 11-168, 2013.

[6] R. Nithyanand and J. Solis, "A theoretical analysis: Physical unclonable functions and the software protection problem," IEEE Symposium on Security and Privacy Workshops, San Francisco, CA, USA, pp. 1-11, IEEE, 2012.

[7] M. Masoumi and S. S. Moghadam, "A simulation-based correlation power analysis attack to FPGA implementation of KASUMI block cipher," Int. J. Internet Technol. Secur. Trans, vol. 7, no. 2, pp. 175-191, 2017.

[8] A. Prathiba and V. Bhaaskaran, "Lightweight S-Box Architecture for Secure Internet of Things," Information, MDPI, vol. 9, no. 1, p. 13, 2018.

[9] W. Shan, X. Fu, and Z. Xu, "A secure reconfigurable crypto IC with countermeasures against SPA, DPA, and EMA," IEEE Transactions on Computer-Aided Design of Integrated Circuits Systems, vol. 34, no. 7, pp. 1201-1205, 2015.

[10] P. De, C. Mandal, and U. Prampalli, "Path-Balanced Logic Design to Realize Block Ciphers Resistant to Power and Timing Attacks," IEEE Transactions on Very Large Scale Integration Systems, vol. 27, no. 5, pp. 1080-1092, 2019.

[11] R. Ammupriya and P. Scolar, "Blur Gate Based Data Leakage Reduction Technique," ICON (Integrating Concepts) vol. 4, no. 3, p. 10, 2019.

[12] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," *DTIC Document, Tech. Rep*., 2001.

[13] G. Marsaglia., "Diehard battery of tests of randomness" [Online] Available: http://www.stat.fsu.edu/pub/diehard/

[14] Morioka, Sumio, and Akashi Satoh. "An optimized S-Box circuit architecture for low power AES design." In *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 172-186. Springer, Berlin, Heidelberg, 2002.

[15] Toumazou, Chris, George S. Moschytz, and Barrie Gilbert, eds. *Trade-offs in analog circuit design: the designer's companion*. Springer Science & Business Media, pp 75-114, 2004.

[16] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Silicon physical random functions," proceeding of 9th ACM conference on Computer and communications security, ACM, Washington DC, USA, pp. 148-160, 2002

[17] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," proceeding of 44th ACM/IEEE Design Automation Conference, San Diego, CA, USA, pp. 9-14, IEEE, 2007.

[18] H. Kang, Y. Hori, T. Katashita, A. Satoh, and K. Iwamura, "PUF Evaluation with Post-processing and Modified Modeling Attack, "International Journal of Security Its Applications, vol. 7, no. 4, pp. 231-242, 2013.

[19] A. Maiti and P. Schaumont, "Improved ring oscillator PUF: An FPGA-friendly secure primitive," Journal of cryptology, vol. 24, no. 2, pp. 375-397, 2011.

[20] S. Eiroa, M. I. Baturone Castillo, A. J. Acosta Jiménez, and J. Dávila, "Using physical unclonable functions for hardware authentication: A survey," proceeding of XXV Conference on Design of Circuits and Integrated Systems (DCIS), 2010.

[21] A. Kumar and R. S. Mishra, "Challenge-Response Pair (CRP) Generator Using Schmitt Trigger Physical Unclonable Function," proceeding of Advanced Computing and Communication Technologies, Panipat, India vol. 702, pp. 213-223, Springer, 2019.

[22] S. S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen, and P. Tuyls, "The butterfly PUF protecting IP on every FPGA," IEEE International Workshop on Hardware-Oriented Security and Trust, Anaheim, CA, USA, pp. 67-70, IEEE, 2008.

[23] S. Katzenbeisser, Ü. Kocabaş, V. Rožić, A.-R. Sadeghi, I. Verbauwhede, and C. Wachsmann, "PUFs: Myth, fact or busted? A security evaluation of physically unclonable functions (PUFs) cast in silicon," International Workshop on Cryptographic Hardware and Embedded Systems, Leuven, Belgium, vol. 7428, pp. 283-301, Springer, 2012.

[24] A. Kumar, J. Pathak, and S. L. Tripathi, "Frequency-Based RO-PUF," in AI Techniques for Reliability Prediction for Electronic Components: IGI Global, pp. 252-261, 2020.

[25] M. Majzoobi, M. Rostami, F. Koushanfar, D. S. Wallach, and S. Devadas, "Slender PUF protocol: A lightweight, robust, and secure authentication by substring matching, "IEEE Symposium on Security and Privacy Workshops, San Francisco, CA, USA, pp. 33-44, IEEE, 2012.

[26] H. Handschuh, G.-J. Schrijen, and P. Tuyls, "Hardware intrinsic security from physically unclonable functions," Towards Hardware-Intrinsic Security, pp. 39-53 Springer, 2010.

[27] B. Barak, R. Shaltiel, and E. Tromer, "True random number generators secure in a changing environment," International Workshop on Cryptographic Hardware and Embedded Systems, Cologne, Germany, 2003, vol. 2779, pp. 166-180: Springer, 2003.

[28] M. Bhargava, C. Cakir, and K. Mai, "Attack resistant sense amplifier based PUFs (SA-PUF) with deterministic and controllable reliability of PUF responses," IEEE international symposium on hardware-oriented security and trust (HOST), Anaheim, CA, USA, pp. 106-111, IEEE, 2010.

[29] NXP Semiconductors N.V, "PUF-physical unclonable functions-protecting next-generation smart card ics with sram based pufs," http: //www.nxp.com/documents/other/75017366.pdf, Feb. 2013.

[30] F. S. Hossain and M. L. Ali, "A Novel Byte-Substitution Architecture for the AES Cryptosystem," PloS one, vol. 10, no. 10, 2015.

[31] A. Maiti, R. Nagesh, A. Reddy, and P. Schaumont, "Physical unclonable function and true random number generator: a compact and scalable implementation," 19th ACM Great Lakes Symposium on VLSI, Boston Area, MA, USA, pp. 425-428: ACM, 2009.

[32] Kong, Joonho, and Farinaz Koushanfar. "Processor-based strong physical unclonable functions with aging-based response tuning." IEEE Transactions on Emerging Topics in Computing, vol. 2, no. 1,pp. 16-29, 2013.

[33] Maiti, Abhranil, Vikash Gunreddy, and Patrick Schaumont. "A systematic method to evaluate and compare the performance of physical unclonable functions." In Embedded systems design with FPGAs, pp. 245-267. Springer, New York, NY, 2013

[34] S. Zeitouni, Y. Oren, C. Wachsmann, P. Koeberl, and A.-R. Sadeghi, "Remanence decay side-channel: The PUF case," J. IEEE Transactions on Information Forensics Security vol. 11, no. 6, pp. 1106-1116, 2015.

[35] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," Proceedings of the IEEE, vol. 102, no. 8, pp. 1126-1141, 2014.

[36] S. Meguerdichian and M. Potkonjak, "Device aging-based physically unclonable functions," 48th ACM/EDAC/IEEE Design Automation Conference (DAC), New York, NY, USA, pp. 288-289, IEEE, 2011.

[37] Z. C. Jouini, J.-L. Danger, and L. Bossuet, "Characterization of Physically Unclonable Functions at Design Stage, "Colloque du GDR SoC-SiP, Paris, France< hal-00753222>, C. Universite de Lyon, Saint-Etienne, France, Ed., ed. France, 2012.

[38] R. Pegu and R. Mudoi, "Design and analysis of Mux based physical unclonable functions," IEEE Trans. Comput.-Aided Design Integr. Circ. Syst, vol. 33, no. 5, pp. 649-662, 2015.

[39] E. Ozturk, G. Hammouri, and B. Sunar, "Physical unclonable function with tristate buffers, "IEEE International Symposium on Circuits and Systems, Seattle, WA, USA, pp. 3194-3197, IEEE, 2008.

[40] S. U. Hussain, S. Yellapantula, M. Majzoobi, and F. Koushanfar, "BIST-PUF: Online, hardware-based evaluation of physically unclonable circuit identifiers," proceeding of IEEE/ACM International Conference on Computer-Aided Design (ICCAD), San Jose, CA, USA, pp. 162-169, IEEE, 2014.

[41] Y. Yao, M. Kim, J. Li, I. L. Markov, and F. Koushanfar, "ClockPUF: Physical Unclonable Functions based on clock networks," proceeding of Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, pp. 422-427, IEEE, 2013.

[42] H. Mestiri, N. Benhadjyoussef, M. Machhout, and R. Tourki, "A comparative study of power consumption models for CPA attack," International Journal of Computer Network and Information Security, vol. 5, no. 3, p. 25, 2013.

[43] T. Fujino, "Tamper-resistant memory integrated circuit and encryption circuit using same," U.S. Patent 8,861,720, Washington, DC: U.S. Patent and Trademark Office. October 14, 2014.

[44] M. Alioto, L. Giancane, G. Scotti, and A. Trifiletti, "Leakage power analysis attacks: A novel class of attacks to nanometer cryptographic circuits," IEEE Transactions on Circuits Systems I: Regular Papers, vol. 57, no. 2, pp. 355-367, 2009.

[45] J. Liu, D. Gu, and Z. Guo, "Correlation power analysis against stream cipher mickey v2, "International Conference on Computational Intelligence and Security, Nanning, China, pp. 320-324, IEEE, 2010.

[46] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model, "International workshop on cryptographic hardware and embedded systems, Cambridge, MA, USA, vol. 3156, pp. 16-29, Springer, 2004.

[47] A. Satoh, S. Morioka, K. Takano, and S. Munetoh, "A compact Rijndael hardware architecture with S-box optimization," proceeding of International Conference on the Theory and Application of Cryptology and Information Security, Gold Coast, QLD, Australia, 2001, vol. 2248, pp. 239-254, Berlin, Heidelberg: Springer, 2001.

[48] C. Canovas and J. Clédière, "What do S-boxes say in differential side-channel attacks?," IACR Cryptology ePrint Archive, p 311, 2005.

[49] D. Bellizia, S. Bongiovanni, P. Monsurro, G. Scotti, and A. Trifiletti, "Univariate power analysis attacks exploiting static dissipation of nanometer CMOS VLSI circuits for cryptographic applications," IEEE Transactions on Emerging Topics in Computing, vol. 5, no. 3, pp. 329-339, 2016.

[50] Y.-J. Baek and M.-J. Noh, "Differential power attack and masking method," Trends in Mathematics, vol. 8, no. 1, pp. 53-67, 2005.

[51] O. Lo, W. J. Buchanan, and D. Carson, "Power analysis attacks on the AES -128 S-box using differential power analysis (DPA) and correlation power analysis (CPA)," Journal of Cyber Security Technology, vol. 1, no. 2, pp. 88-107, 2017.

[52] P. Saravanan and P. Kalpana, "An energy-efficient XOR gate implementation resistant to power analysis attacks," J. Eng. Sci. Technol, vol. 10, pp. 1275-1292, 2015.

[53] L. Zhang, L. Vega, and M. Taylor, "Power side channels in security ICs: hardware countermeasures," arXiv preprint arXiv: 00681," University of California, San Diego2016.

[54] S. Mangard, T. Popp, and B. M. Gammel, "Side-Channel Leakage of Masked CMOS Gates," Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, vol. 3376, pp. 351-365: Springer Berlin Heidelberg, 2004

[55] M. Jayakumar, FPGA-Masked S-Box Implementation for AES Engine (VLSI Design: Circuits, Systems, and Applications, no. 469). pp. 223-233, Springer,2018.

[56] K. Tiri et al., "AES -based cryptographic and biometric security coprocessor IC in 0.18-/spl mu/m CMOS resistant to side-channel power analysis attacks," proceeding of 42nd Design Automation Conference, 2005., Anaheim, CA, USA, pp. 216-219, IEEE, 2005.

[57] M. Masoumi, "A highly efficient and secure hardware implementation of the advanced encryption standard," Journal of Information Security Applications, vol. 48, p. 102371, 2019.

[58] E. Tena-Sanchez, J. Castro, and A. J. Acosta, "A methodology for optimized design of secure differential logic gates for DPA resistant circuits," IEEE Journal on Emerging Selected Topics in Circuits Systems, vol. 4, no. 2, pp. 203-215, 2014.

[59] T. Popp and S. Mangard, "Masked dual-rail pre-charge logic: DPA-resistance without routing constraints, "International Workshop on Cryptographic Hardware and Embedded Systems, Edinburgh, United Kingdom, vol. 3659, pp. 172-186, Springer, 2005.

[60] Y.-h. ZENG, X.-c. ZOU, Z.-l. LIU, and J.-m. LEI, "Low-power clock-less hardware implementation of the Rijndael S-box for wireless sensor networks," The Journal of China Universities of Posts and Telecommunications, vol. 14, no. 4, pp. 104-109, 2007.

[61] S. Zhang and W. Zhong, "A New Type of Countermeasure against DPA in Multi-Sbox of Block Cipher," Wireless Communications Mobile Computing, vol. 2018, no. Article ID 5945312, p. 11 Pages, 2018.

[62] S. N. Dhanuskodi, S. Keshavarz, and D. Holcomb, "LLPA: logic state based leakage power analysis, "IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Pittsburgh, PA, USA, pp. 218-223, IEEE, 2016.

[63] C. Teegarden, M. Bhargava, and K. Mai, "Side-channel attack resistant ROM-based AES S-Box, "IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Anaheim, CA, USA, pp. 124-129, IEEE, 2010.

[64] B. Halak, J. Murphy, and A. Yakovlev, "Power balanced circuits for leakage-power-attacks resilient design, "Science and Information Conference (SAI), London, UK, pp. 1178-1183, IEEE, 2015.

[65] C. Monteiro, Y. Takahashi, and T. Sekine, "Resistance against power analysis attacks on adiabatic dynamic and adiabatic differential logics for smart card," International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS), Chiang Mai, Thailand, pp. 1-5, IEEE, 2015.

[66] K. Tiri and I. Verbauwhede, "Securing encryption algorithms against DPA at the logic level: Next-generation smart card technology, "International Workshop on Cryptographic Hardware and Embedded Systems, Cologne, Germany, 2003, vol. 2779, pp. 125-136: Springer, 2003.

[67] S. Mangard, E. Oswald, and F.-X. Standaert, "One for all–all for one: unifying standard differential power analysis attacks," IET Information Security, vol. 5, no. 2, pp. 100-110, 2011.

[68] J. J. Fournier, S. Moore, H. Li, R. Mullins, and G. Taylor, "Security evaluation of asynchronous circuits, "International Workshop on Cryptographic Hardware and Embedded Systems, Cologne, Germany, 2003, vol. 2779, pp. 137-151: Springer, 2003.

[69] N. Mentens, L. Batina, B. Preneel, and I. Verbauwhede, "A systematic evaluation of compact hardware implementations for the Rijndael S-box, "Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, vol. 3376, pp. 323-333, Springer, 2005.

[70] F. A. Khan, J. Ahmed, J. S. Khan, J. Ahmad, M. A. Khan, and S. O. Hwang, "A new technique for designing 8× 8 substitution box for image encryption applications, "9th Computer Science and Electronic Engineering (CEEC), Colchester, UK, pp. 7-12, IEEE, 2017.

[71] D. Kamel, F.-X. Standaert, and D. Flandre, "Scaling trends of the AES S-box low power consumption in 130 and 65 nm CMOS technology nodes, "IEEE International Symposium on Circuits and Systems, Taipei, Taiwan, pp. 1385-1388, IEEE, 2009.

[72] E. M. Mahmoud, A. Abd, T. A. E. El Hafez, and T. A. El Hafez, "Dynamic AES - 128 with key-dependent S-box," International Journal of Engineering Research and Applications, vol. 3, no. 1, pp. 1662-1670, 2013.

[73] A. Pammu, K.-S. Chong, and B.-H. Gwee, "Secured low power overhead compensator look-up-table (LUT) substitution box (S-Box) architecture," proceeding of IEEE International Conference on Networking, Architecture and Storage (NAS), Long Beach, CA, USA, pp. 1-7, IEEE, 2016.

[74] S. Dey and R. Ghosh, "A review of existing 4-bit crypto S-box cryptanalysis techniques and two new techniques with 4-bit Boolean functions for cryptanalysis of 4-bit crypto S-boxes," Advances in Pure Mathematics, vol. 8, no. 03, p. 272, 2018.

[75] K. Tanimura and N. Dutt, "A standard cell-based DPA attack countermeasure using homogeneous dual-rail logic (HDRL)," J University of California, pp. 12-01, 2012.

[76] Lao, Yingjie, and Keshab K. Parhi. "Statistical analysis of MUX-based physical unclonable functions." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 33, no. 5, pp. 649-662, 2014.

[77] S. Baur and H. Boche, "Robust secure storage of data sources with perfect secrecy," IEEE Workshop on Information Forensics and Security (WIFS), Rennes, France, pp. 1-6, IEEE, 2017.

[78] A. Maiti, I. Kim, and P. Schaumont, "A robust physical unclonable function with enhanced challenge-response set," IEEE Transactions on Information Forensics Security, vol. 7, no. 1, pp. 333-345, 2011.

[79] A. Mutlu, K. J. Le, M. Celik, D.-s. Tsien, G. Shyu, and L.-C. Yeh, "An exploratory study on statistical timing analysis and parametric yield optimization, "8th International Symposium on Quality Electronic Design (ISQED'07), San Jose, CA, USA, pp. 677-684, IEEE, 2007.

[80] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola, and V. Khandelwal, "Design and implementation of PUF-based "unclonable" RFID ICs for anti-counterfeiting and security applications," proceeding of IEEE international conference on RFID, Las Vegas, NV, USA, pp. 58-64, IEEE, 2008.

[81] Y. Wen and Y. Lao, "Efficient fuzzy extractor implementations for PUF based authentication," proceeding of the 12th International Conference on Malicious and Unwanted Software (MALWARE), Fajardo, Puerto Rico, 2017.

[82] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "Efficient and high-performance parallel hardware architectures for the AES -GCM," IEEE Transactions on Computers, vol. 61, no. 8, pp. 1165-1178, 2011.

[83] Z. Paral and S. Devadas, "Reliable and efficient PUF-based key generation using pattern matching, "IEEE International Symposium on Hardware-Oriented Security and Trust, San Diego CA, USA, pp. 128-133, IEEE, 2011.

[84] A. Kumar, R. S. Mishra, and K. Kashwan, "PUF Based challenge Response Pair For Secured Authentication," International Journal of Control Theory and Applications, vol. 9, no. 41, pp. 115-121, 2017.

[85] M. Rostami, F. Koushanfar, and R. Karri, "A primer on hardware security: Models, methods, and metrics," Proceedings of the IEEE, vol. 102, no. 8, pp. 1283-1295, 2014.

[86] A. Kumar, R. S. Mishra, and K. Kashwan, "Challenge-response generation using RO-PUF with reduced hardware," proceeding of International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, India, pp. 1305-1308, IEEE, 2016.

[87] G. T. Becker and Systems, "On the pitfalls of using arbiter-PUFs as building blocks," IEEE Transactions on Computer-Aided Design of Integrated Circuits, vol. 34, no. 8, pp. 1295-1307, 2015.

[88] X. Zhang and K. K. Parhi, "High-speed VLSI architectures for the AES algorithm," IEEE transactions on very large scale integration systems, vol. 12, no. 9, pp. 957-967, 2004.

[89] R. Kumar, V. C. Patil, and S. Kundu, "Design of unique and reliable physically unclonable functions based on current starved inverter chain," IEEE Computer Society Annual Symposium on VLSI, Chennai, India, pp. 224-229, IEEE, 2011.

[90] Z. Cherif, J.-L. Danger, S. Guilley, and L. Bossuet, "An easy-to-design PUF based on a single oscillator: the loop PUF," proceeding of 15th Euromicro Conference on Digital System Design, Izmir, Turkey, pp. 156-162, IEEE, 2012.

[91] H. Bin, S. Goto, and Y. Tsunoo, "A multiple bits output ring-oscillator physical unclonable function," proceeding of International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS), Chiang Mai, Thailand, pp. 1-5: IEEE, 2011.

[92] J.-L. Zhang, G. Qu, Y.-Q. Lv, and Q. Zhou, "A survey on silicon PUFs and recent advances in ring oscillator PUFs," Journal of computer science technology, vol. 29, no. 4, pp. 664-678, 2014.

[93] Y. Cao, L. Zhang, C.-H. Chang, and S. Chen, "A low-power hybrid RO PUF with improved thermal stability for lightweight applications," IEEE Transactions on computer-aided design of integrated circuits systems, vol. 34, no. 7, pp. 1143-1147, 2015.

[94] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," proceeding of International Conference on Field Programmable Logic and Applications, pp. 703-707, IEEE, 2009.

[95] M. Platonov, "SRAM-Based Physical Unclonable Function on an Atmel ATmega Microcontroller," Doctoral dissertation, Master's Thesis, Czech Technical University of Prague, Prague, 2013.

[96] V. Rožić, W. Dehaene, and I. Verbauwhede, "Design solutions for securing SRAM cell against power analysis," proceeding of IEEE International Symposium on Hardware-Oriented Security and Trust, San Francisco, CA, USA, pp. 122-127, IEEE, 2012.

[97] Y. Shifman, A. Miller, O. Keren, Y. Weizmann, and J. Shor, "A Method to Improve Reliability in a 65-nm SRAM PUF Array," IEEE Solid-State Circuits Letters, vol. 1, no. 6, pp. 138-141, 2018.

[98] X. Xu, A. Rahmati, D. E. Holcomb, K. Fu, and W. Burleson, "Reliable physical unclonable functions using data retention voltage of SRAM cells," IEEE Transactions on Computer-Aided Design of Integrated Circuits Systems, vol. 34, no. 6, pp. 903-914, 2015.

[99] F. Arith, M. I. Idris, M. N. S. Zainudin, M. Chachuli, and S. Amaniah, "Low voltage CMOS Schmitt trigger in 0.18 μm technology," IOSR Journal of Engineering, vol. 3, no. 2, pp. 8-15, 2013.

[100] L. A. P. Melek, A. L. da Silva, M. C. Schneider, and C. Galup-Montoro, "Analysis and design of the classical CMOS Schmitt trigger in subthreshold operation," IEEE Transactions on Circuits Systems I: Regular Papers, vol. 64, no. 4, pp. 869-878, 2016.

[101] T. Aoki, Y. Kurokawa, and M. Kozuma, "Semiconductor device having Schmitt trigger NAND circuit and Schmitt trigger inverter," U.S. Patent No. 9,245,589. Washington, DC: U.S. Patent and Trademark Office, January 26, 2016.

[102] S. B. Ramakrishna, S. Madhusudhan, B. Nikshep, B. Naveen, and H. Teerthaprasad, "Power and delay optimization of domino Schmitt trigger configurations

with enhanced hysteresis voltage," Analog Integrated Circuits Signal Processing, vol. 102, no. 1, pp. 1-9, 2019.

[103] J. Park, "Secure hardware design against side-channel attacks," Doctoral thesis, computer Engineering, Iowa state university, Ames, Iowa, 2016.

[104] J. B. Kuang, "SOI CMOS Schmitt trigger circuits with controllable hysteresis," U.S. Patent No. 6,441,663. Washington, DC: U.S. Patent and Trademark Office, August 27, 2002.

[105] D. S. Barlow, "Self-adjusting Schmitt trigger," U.S. Patent No. 7,167,032. Washington, DC: U.S. Patent and Trademark Office, January 23, 2007.

[106] W. Lew and R. Cadotte Jr, "Gate pulsing gate ladder," U.S. Patent No. 9,577,628. Washington, DC: U.S. Patent and Trademark Office, February 21, 2017.

[107] V. Chauhan and P. Garg, "Compensated Schmitt trigger circuit for providing monotonic hysteresis response," U.S. Patent Application No. 11/148,947, January 26, 2006.

[108] A. Kumar, S. L. Tripathi, and R. S. Mishra, "METAPUF: A challenge response pair generator," Periodicals of Engineering Natural Sciences, vol. 6, no. 2, pp. 58-63, 2018.

[109] K. Yelamarthi, "Timing-driven variation-aware partitioning and optimization of mixed static-dynamic CMOS circuits," Circuits and Systems, vol. 4, no. 2, p. s7 pages, Art. no. 29874, 2013.

[110] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection, "International workshop on cryptographic hardware and embedded systems, Vienna, Austria, vol. 4727, pp. 63-80, Springer, 2007.

[111] Standard, N. F. "Announcing the advanced encryption standard (AES)." Federal Information Processing Standards Publication, 197(1-51), pp 3-3, 2001

[112] D. Van Buer, "Method and apparatus for high speed implementation of data encryption and decryption utilizing, eg Rijndael or its subset AES, or other encryption/decryption algorithms having similar key expansion data flow," US, Patent Application 10/040,087, 2003.

[113] A. Kumar and S. Tejani, "S-BOX Architecture," proceeding of International Conference on Futuristic Trends in Network and Communication Technologies, Shimla, India, vol. 958, pp. 17-27, Springer, 2018.

[114] J. Wu, Y.-B. Kim, and M. Choi, "Low-power side-channel attack -resistant asynchronous S-box design for AES cryptosystems, "Proceedings of the 20th symposium on Great lakes symposium on VLSI, Rhode Island, USA, pp. 459-464, ACM, 2010.

[115] J. Wu, Y. Shi, and M. Choi, "Measurement and evaluation of power analysis attacks on asynchronous S-box," IEEE Transactions on Instrumentation Measurement, vol. 61, no. 10, pp. 2765-2775, 2012.

[116] D. Canright, "A very compact S-box for AES, "International Workshop on Cryptographic Hardware and Embedded Systems, Edinburgh, United Kingdom, vol. 3659, pp. 441-455, Springer, 2005.

[117] J. H. Kong, L.-M. Ang, and K. P. Seng, "A very compact AES -SPIHT selective encryption computer architecture design with improved S-box," Journal of Engineering doi.org/10.1155/2013/785126, 2013

[118] H. Kim and S. Hong, "AES Sbox GF (2^4) inversion functions based PUFs," proceeding of international SoC Design Conference (ISOCC), Jeju, South Korea, pp. 15-16, IEEE, 2014.

[119] S. Morioka and A. Satoh, "A 10-Gbps full-AES crypto design with a twisted BDD S-Box architecture," IEEE Transactions on Very Large Scale Integration Systems, vol. 12, no. 7, pp. 686-691, 2004.

[120] E. N. Mui, R. Custom, and D. Engineer, "Practical implementation of Rijndael S-box using Combinational logic," D Engineer Texco Enterprise Pvt. Ltd 2007.

[121] F. Regazzoni et al., "A simulation-based methodology for evaluating the DPA-resistance of cryptographic functional units with application to CMOS and MCML technologies," proceeding of International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation, Samos, Greece, pp. 209-214, IEEE, 2007.

[122] Y. Leblebici and S. Kang, CMOS digital integrated circuits: analysis and design, 4th ed. McGraw-Hill, pp. 378-392, 1996.

[123] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolić, Digital integrated circuits: a design perspective. Pearson Education Upper Saddle River, NJ, 2003.

[124] L. Angrisani, M. D'Apuzzo, and M. Vadursi, "Power measurement in digital wireless communication systems through the parametric spectral estimation," IEEE transactions on instrumentation measurement, vol. 55, no. 4, pp. 1051-1058, 2006.

[125] P. Zode and R. Deshmukh, "MOS switch-based DPA-resistant GF (28) modulo multiplier," International Journal of Electronics Letters, vol. 6, no. 2, pp. 231-241, 2018.

[126] S. Yang, W. Wolf, N. Vijaykrishnan, D. N. Serpanos, and Y. Xie, "Power attack resistant cryptosystem design: A dynamic voltage and frequency switching approach," proceeding of Design, Automation, and Test in Europe, Munich, Germany, pp. 64-69, IEEE, 2005.

[127] K. Tiri and I. Verbauwhede, "A VLSI design flow for secure side-channel attack resistant ICs," in Design, Automation, and Test in Europe, Munich, Germany, Germany, pp. 58-63, IEEE, 2005.

[128] V. Konstantakos, K. Kosmatopoulos, S. Nikolaidis, and T. Laopoulos, "Measurement of power consumption in digital systems," IEEE Transactions on Instrumentation measurement, vol. 55, no. 5, pp. 1662-1670, 2006.

[129] T. Lopez and R. Elferich, "Measurement technique for the static output characterization of high-current power MOSFETs," IEEE Transactions on Instrumentation Measurement, vol. 56, no. 4, pp. 1347-1354, 2007.

[130] S. B. Ors, F. Gurkaynak, E. Oswald, and B. Preneel, "Power-Analysis Attack on an ASIC AES implementation," proceeding of International Conference on Information Technology: Coding and Computing,(ITCC 2004)., Las Vegas, NV, USA, USA, vol. 2, pp. 546-552, IEEE, 2004.

[131] J. Ma, X. Li, and M. Wang, "Power-aware hiding method for S-box protection," Electronics Letters, vol. 50, no. 22, pp. 1604-1606, 2014.

[132] A. J. L. Martin, "Cadence design environment, Tutorial paper," New Mexico State University 2002.

[133] A. Singh, P. Agarwal, and M. Chand, "Analysis of Development of Dynamic S-Box Generation," Comput. Sci. Inf. Technol, vol. 5, no. 5, pp. 154-163, 2017.

[134] Mangard, Stefan, Thomas Popp, and Berndt M. Gammel. “Side-Channel Leakage of Masked CMOS Gates.” proceedings of Topics in Cryptology CT-RSA, Lecture Notes in Computer Science, vol 3376, Berlin, Heidelberg:springer, 2005.

[135] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," proceeding of Annual International Cryptology Conference, Santa Barbara, CA, USA, vol. 1666, pp. 388-397, Springer, Berlin, Heidelberg: Springer, 1999.

[136] J. J. Pettengill and W. R. Hnath, "Differential Power Analysis Side-Channel Attacks in Cryptography," Journal of Cryptographic Engineering, vol. 1, no. 1, pp. 5-27, 2010.

[137] F.-X. Standaert, “Introduction to side-channel attacks, Secure Integrated Circuits, and Systems (Integrated Circuits and Systems)”. US: Springer, 2009.

[138] M. Alioto, S. Bongiovanni, M. Djukanovic, G. Scotti, and A. Trifiletti, "Effectiveness of leakage power analysis attacks on DPA-resistant logic styles under process variations," IEEE Transactions on Circuits Systems I: Regular Papers, vol. 61, no. 2, pp. 429-442, 2013.

[139] W. Yu and S. Köse, "A voltage regulator-assisted lightweight AES implementation against DPA attacks," IEEE Transactions on Circuits Systems I: Regular Papers, vol. 63, no. 8, pp. 1152-1163, 2016.

[140] W. Yu and S. Köse, "A lightweight masked AES implementation for securing IoT against CPA attacks," IEEE Transactions on Circuits Systems I: Regular Papers, vol. 64, no. 11, pp. 2934-2944, 2017.

[141] Y. HAN, X.-c. ZOU, Z.-l. LIU, and Y.-c. CHEN, "The research of DPA attacks against AES implementations," The Journal of China Universities of Posts and Telecommunications, vol. 15, no. 4, pp. 101-106, 2008.

[142] W. Schindler, K. Lemke, and C. Paar, "A stochastic model for differential side-channel cryptanalysis," proceeding if international Workshop on Cryptographic Hardware and Embedded Systems, Edinburgh, United Kingdom, vol. 3659, pp. 30-46, Springer, 2005.

[143] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," National Taiwan University, Taiwan 2003.

[144] Z. Toprak, A. Verma, Y. Leblebici, P. Ienne, and C. Paar, "Design of low-power DPA-resistant cryptographic functional units," proceeding of the Cryptographic Advances in Secure Hardware, 2005.

[145] D. Yamamoto, G. Hospodar, R. Maes, and I. Verbauwhede, "Performance and security evaluation of AES s-box-based glitch PUFs on FPGAs," proceeding of

International Conference on Security, Privacy, and Applied Cryptography Engineering, Chennai, India, vol. 7644, pp. 45-62, Springer, 2012.

[146] P. Hamalainen, T. Alho, M. Hannikainen, and T. D. Hamalainen, "Design and implementation of low-area and low-power AES encryption hardware core," proceeding of 9th EUROMICRO conference on digital system design (DSD'06), Dubrovnik, Croatia, pp. 577-583, IEEE, 2006.

[147] M. Nassar, Y. Souissi, S. Guilley, and J.-L. Danger, "RSM: a small and fast countermeasure for AES, secure against 1st and 2nd-order zero-offset SCAs," proceeding of Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, pp. 1173-1178, IEEE, 2012.

[148] R. P. McEvoy, C. C. Murphy, W. P. Marnane, M. Tunstall, and Systems, "Isolated WDDL: a hiding countermeasure for differential power analysis on FPGAs," ACM Transactions on Reconfigurable Technology, vol. 2, no. 1, pp. 1-23, 2009.

[149] Y. Li, K. Ohta, and K. Sakiyama, "Revisit fault sensitivity analysis on WDDL-AES," proceeding of IEEE International Symposium on Hardware-Oriented Security and Trust, San Diego CA, USA, pp. 148-153, IEEE, 2011.

[150] H. Kim, S. Hong, and J. Lim, "A fast and provably secure higher-order masking of AES S-box," proceeding of International Workshop on Cryptographic Hardware and Embedded Systems, Nara, Japan vol. 6917, pp. 95-107, Springer, 2011.

[151] T. S. Messerges, "Power analysis attacks and countermeasures for cryptographic algorithms," Doctoral Thesis, Elect. Eng./Computer Science, the University of Illinois at Chicago, the University of Illinois at Chicago, United States, 2000.

[152] H. Gross, K. Stoffelen, L. De Meyer, M. Krenn, and S. Mangard, "First-Order Masking with Only Two Random Bits," proceeding of the Proceeding of ACM Workshop on Theory of Implementation Security Workshop, London, United Kingdom, 2019.

[153] J. Boyar and R. Peralta, "A new combinational logic minimization technique with applications to cryptology," proceedings of Experimental Algorithms SEA. Lecture Notes in Computer Science, vol 6049, pp. 178-189, Naples, Italy, Springer, 2010.

[154] J. D. Golic and R. Menicocci, "Universal masking on logic gate level," Electronics Letters, vol. 40, no. 9, pp. 526-528, 2004.

[155] J.-F. Lin, Y.-T. Hwang, M.-H. Sheu, and C.-C. Ho, "A novel high-speed and energy-efficient 10-transistor full adder design," IEEE Transactions on Circuits Systems I: Regular Paper, vol. 54, no. 5, pp. 1050-1059, 2007.

[156] J. Zeng and C. Z. Xu, "An improved masked S-box for AES and hardware implementation," Journal of Convergence Information Technology, vol. 7, no. 10, pp. 338-344, 2012.

[157] E. Trichina, "Combinational Logic Design for AES SubByte Transformation on Masked Data," IACR Eprint archive, vol. 2003, p. 236, IACR, 2003.

[158] M. Alam, S. Ghosh, M. Mohan, D. Mukhopadhyay, D. R. Chowdhury, and I. Gupta, "Effect of glitches against masked AES S-box implementation and countermeasure," IET Information Security, vol. 3, no. 1, pp. 34-44, 2009.

[159] K. Kumar, D. Mukhopadhyay, and D. Roy Chowdhury, "Design of a differential power analysis resistant masked AES S-Box," proceeding of International Conference on Cryptology in India, Chennai, India, vol. 4859, pp. 373-383, Springer, 2007.

[160] Y. Zhou, G. Qian, Y. Xing, H. Liu, S. Goto, and Y. Tsunoo, "An approach of using different positions of double registers to protect AES hardware structure from DPA," proceeding of third International Symposium on Electronic Commerce and Security, Guangzhou, China, pp. 223-227, IEEE, 2010.

[161] O. Adegbite and S. R. Hasan, "A novel correlation power analysis attack on PIC based AES -128 without access to crypto device," proceeding of 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, USA, pp. 1320-1323, IEEE, 2017.

[162] J. M. Padilla, U. Meyer-Baese, and S. Foo, "Security evaluation of Tree Parity Re-keying Machine implementations utilizing side-channel emissions," EURASIP Journal on Information Security, vol. 2018, no. 1, pp. 1-16, 2018.

[163] H. Mestiri, F. Kahri, B. Bouallegue, and M. Machhout, "A CPA attack against cryptographic hardware implementation on SASEBO-GII," proceeding of International Conference on Green Energy Conversion Systems (GECS), Hammamet, Tunisia, pp. 1-5, IEEE, 2017.

[164] M. Jahanbani, Z. Norouzi, and N. Bagheri, "CPA on hardware implementation of COLM authenticated cipher and protect it with DOM masking scheme," Cryptology ePrint Archive, Tech. Rep. 2019/954, 2019.

[165] Zhang, Liwei, A. Adam Ding, Yunsi Fei, and Pei Luo. "A unified metric for quantifying information leakage of cryptographic devices under power analysis attacks." proceeding of International Conference on the Theory and Application of Cryptology and Information Security, pp. 338-360. Springer, Berlin, Heidelberg, 2015.

INDEX

List of Publication

In this thesis, we have designed a new architecture for PUF based on Schmitt trigger circuit and power attack resistant cryptographic module using a mask cell. The obtained results are published in the following journal/conference/patent/copyright and book chapter

Objective-1

[1] Kumar, Abhishek, Suman Lata Tripathi, and Ravi Shankar Mishra. "METAPUF: A challenge response pair generator." *Periodicals of Engineering and Natural Sciences* 6, no. 2 (2018): 58-63, **SJR 0.2**

[2] Kumar, Abhishek, Ravi Shankar Mishra, and K. R. Kashwan. "Challenge-response generation using RO-PUF with reduced hardware." *International Conference on Advances in Computing, Communications, and Informatics (ICACCI)*, pp. 1305-1308. IEEE, 2016.

[3] Kumar, Abhishek, Jyotirmoy Pathak, and Suman Lata Tripathi. "Frequency-Based RO-PUF." *AI Techniques for Reliability Prediction for Electronic Components*, pp. 252-261. IGI Global, 2020.

Objective-2

[1] Kumar, Abhishek, Ravi Shankar Mishra, and K. R. Kashwan. "PUF Based challenge Response Pair for Secured Authentication." *International Journal of Control Theory and Applications*, vol 9, issue 41 (2016): 115-121, **SJR 0.11**

[2] Kumar, Abhishek, and Ravi Shankar Mishra. "Challenge-Response Pair (CRP) Generator Using Schmitt Trigger Physical Unclonable Function." *Advanced Computing and Communication Technologies*, pp. 213-223. Springer, Singapore, 2019.

[3] Kumar, Abhishek, Jyotirmoy Pathak, and Suman Lata Tripathi. "A system to generate key for hardware authentication." *The Patent Office,* Journal No. 51/2019, Application No.201911049739 A

Objective-3

[1] Kumar, Abhishek, and Sokat Tejani. "S-BOX Architecture." *International Conference on Futuristic Trends in Network and Communication Technologies*, pp. 17-27. Springer, Singapore, 2018.

[2] Kumar Abhishek, Suman Lata Tripathi, Ravi Shankar Mishra, and K. R. Kashwan "Power Calculation Method of CMOS Circuit." *Copyright office Government of India*, Registration No. L-89581/2020

[3] Kumar, A., Tripathi, S.L. SBOX under PVT variation. *Analog Integr Circ Sig Process* (2020). https://doi.org/10.1007/s10470-020-01691-0, **SCI IF 0.925**

Objective-4

[1] Kumar Abhishek, and Suman Lata Tripathi. "Power attack resistant masked logic gate." *The Patent Office*, Journal No. 52/2019, Application No.201911053046 A

[2] Kumar, Abhishek, Jyotirmoy Pathak, and Suman Lata Tripathi. "Power analysis to ensure CMOS architecture." *Recent Advancement in Electronic Devices, Circuit, and Materials*, pp. 71-86. Nova Science, 2020